NAME

camcontrol - CAM control program

SYNOPSIS

camcontrol <*command*> [device id] [generic args] [command args]

camcontrol devlist [**-b**] [**-v**]

camcontrol periphlist [device id] [**-n** *dev_name*] [**-u** *unit_number*]

camcontrol tur [device id] [generic args]

camcontrol inquiry [device id] [generic args] [**-D**] [**-S**] [**-R**]

camcontrol identify [device id] [generic args] [**-v**]

camcontrol reportluns [device id] [generic args] [**-c**] [**-l**] [**-r** *reporttype*]

camcontrol readcap [device id] [generic args] [**-b**] [**-h**] [**-H**] [**-l**] [**-N**] [**-q**] [**-s**]

camcontrol start [device id] [generic args]

camcontrol stop [device id] [generic args]

camcontrol load [device id] [generic args]

camcontrol eject [device id] [generic args]

camcontrol reprobe [device id]

camcontrol rescan <all | device id | bus[:target:lun]>

camcontrol reset <all | device id | bus[:target:lun]>

camcontrol defects [device id] [generic args] <**-f** *format*> [**-P**] [**-G**] [**-q**] [**-s**] [**-S** *offset*] [**-X**]

camcontrol modepage [device id] [generic args] [**-6**] <**-m** *page[,subpage]* | **-l**> [**-P** *pgctl*] [**-D**] [**-L**] [**-b** | **-e**] [**-d**]

camcontrol cmd [device id] [generic args] <**-a** *cmd* [args]> <**-c** *cmd* [args]> [**-d**] [**-f**] [**-i** *len fmt*] [**-o** *len fmt* [args]] [**-r** *fmt*]

camcontrol smpcmd [device id] [generic args] <**-r** *len fmt* [args]> <**-R** *len fmt* [args]>

camcontrol smprg [device id] [generic args] [**-l**]

camcontrol smppc [device id] [generic args] <**-p** *phy*> [**-l**] [**-o** *operation*] [**-d** *name*] [**-m** *rate*] [**-M** *rate*] [**-T** *pp_timeout*] [**-a** *enable*|*disable*] [**-A** *enable*|*disable*] [**-s** *enable*|*disable*] [**-S** *enable*|*disable*]

camcontrol smpphylist [device id] [generic args] [**-l**] [**-q**]

camcontrol smpmaninfo [device id] [generic args] [**-l**]

camcontrol debug [**-I**] [**-P**] [**-T**] [**-S**] [**-X**] [**-c**] [**-p**] <all | off | device id | bus[:target[:lun]]>

camcontrol tags [device id] [generic args] [**-N** *tags*] [**-q**] [**-v**]

camcontrol negotiate [device id] [generic args] [**-c**] [**-D** *enable*|*disable*] [**-M** *mode*] [**-O** *offset*] [**-q**] [**-R** *syncrate*] [**-T** *enable*|*disable*] [**-U**] [**-W** *bus_width*] [**-v**]

camcontrol format [device id] [generic args] [**-q**] [**-r**] [**-w**] [**-y**]

camcontrol sanitize [device id] [generic args] <**-a** *overwrite* | *block* | *crypto* | *exitfailure*> [**-c** *passes*] [**-I**] [**-P** *pattern*] [**-q**] [**-U**] [**-r**] [**-w**] [**-y**]

camcontrol idle [device id] [generic args] [**-t** *time*]

camcontrol standby [device id] [generic args] [**-t** *time*]

camcontrol sleep [device id] [generic args]

camcontrol powermode [device id] [generic args]
camcontrol apm [device id] [generic args] [**-l** *level*]
camcontrol aam [device id] [generic args] [**-l** *level*]
camcontrol fwdownload [device id] [generic args] <**-f** *fw_image*> [**-q**] [**-s**] [**-y**]
camcontrol security [device id] [generic args] [**-d** *pwd*] [**-e** *pwd*] [**-f**] [**-h** *pwd*] [**-k** *pwd*]
[**-l** *high|maximum*] [**-q**] [**-s** *pwd*] [**-T** *timeout*] [**-U** *user|master*] [**-y**]
camcontrol hpa [device id] [generic args] [**-f**] [**-l**] [**-P**] [**-p** *pwd*] [**-q**] [**-s** *max_sectors*] [**-U** *pwd*] [**-y**]
camcontrol ama [device id] [generic args] [**-f**] [**-q**] [**-s** *max_sectors*]
camcontrol persist [device id] [generic args] <**-i** *action* | **-o** *action*> [**-a**] [**-I** *trans_id*] [**-k** *key*] [**-K** *sa_key*]
[**-p**] [**-R** *rel_tgt_port*] [**-s** *scope*] [**-S**] [**-T** *res_type*] [**-U**]
camcontrol attrib [device id] [generic args] <**-r** *action* | **-w** *attrib*> [**-a** *attr_num*] [**-c**] [**-e** *elem_addr*]
[**-F** *form1,form2*] [**-p** *part*] [**-s** *start_addr*] [**-T** *elem_type*] [**-V** *lv_num*]
camcontrol opcodes [device id] [generic args] [**-o** *opcode*] [**-s** *service_action*] [**-N**] [**-T**]
camcontrol zone <**-c** *cmd*> [**-a**] [**-l** *lba*] [**-o** *rep_opts*] [**-P** *print_opts*]
camcontrol epc <**-c** *cmd*> [**-d**] [**-D**] [**-e**] [**-H**] [**-p** *power_cond*] [**-P**] [**-r** *restore_src*] [**-s**] [**-S** *power_src*]
[**-T** *timer*]
camcontrol timestamp [device id] [generic args] <**-r** [**-f** *format* | **-m** | **-U**] | **-s** <**-f** *format* **-T** *time* | **-U**>>
camcontrol devtype [device id]
camcontrol depop [device id] [generic args] <**-l** | **-d** | **-r**> [**-e** *elem*] [**-c** *capacity*]
camcontrol help

DESCRIPTION

The **camcontrol** utility allows users to access and control the FreeBSD CAM subsystem described in cam(4).

The **camcontrol** utility can cause a loss of data and/or system crashes if used improperly. Even expert users are encouraged to exercise caution when using this command. Novice users should stay away from this utility.

The **camcontrol** utility has a number of primary functions, many of which support an optional device identifier. A device identifier can take one of three forms:

deviceUNIT Specify a device name and unit number combination, like "da5" or "cd3".

bus:target Specify a bus number and target id. The bus number can be determined from the output of "camcontrol devlist". The lun defaults to 0.

bus:target:lun Specify the bus, target and lun for a device. (e.g. 1:2:0)

The device identifier, if it is specified, *must* come immediately after the function name, and before any generic or function-specific arguments. Note that the **-n** and **-u** arguments described below will override any device name or unit number specified beforehand. The **-n** and **-u** arguments will *not* override a specified bus:target or bus:target:lun, however.

Most of the **camcontrol** primary functions support these generic arguments:

-C *count* SCSI command retry count. In order for this to work, error recovery (**-E**) must be turned on.

-E Instruct the kernel to perform generic SCSI error recovery for the given command. This is needed in order for the retry count (**-C**) to be honored. Other than retrying commands, the generic error recovery in the code will generally attempt to spin up drives that are not spinning. It may take some other actions, depending upon the sense code returned from the command.

-n *dev_name* Specify the device type to operate on, e.g. "da", "cd".

-Q *task_attr* SCSI task attribute for the command, if it is a SCSI command. This may be ordered, simple, head, or aca. In most cases this is not needed. The default is simple, which works with all SCSI devices. The task attribute may also be specified numerically.

-t *timeout* SCSI command timeout in seconds. This overrides the default timeout for any given command.

-u *unit_number* Specify the device unit number, e.g. "1", "5".

-v Be verbose, print out sense information for failed SCSI commands.

Primary command functions:

devlist List all physical devices (logical units) attached to the CAM subsystem. This also includes a list of peripheral drivers attached to each device. With the **-v** argument, SCSI bus number, adapter name and unit numbers are printed as well. On the other hand, with the **-b** argument, only the bus adapter, and unit information will be printed, and device information will be omitted.

periphlist List all peripheral drivers attached to a given physical device (logical unit).

tur Send the SCSI test unit ready (0x00) command to the given device. The **camcontrol** utility will report whether the device is ready or not.

inquiry Send a SCSI inquiry command (0x12) to a device. By default, **camcontrol** will print out the standard inquiry data, device serial number, and transfer rate information. The user can specify that only certain types of inquiry data be printed:

-D Get the standard inquiry data.

-S Print out the serial number. If this flag is the only one specified, **camcontrol** will not print out "Serial Number" before the value returned by the drive. This is to aid in script writing.

-R Print out transfer rate information.

identify Send a ATA identify command (0xec) to a device.

reportluns

Send the SCSI REPORT LUNS (0xA0) command to the given device. By default, **camcontrol** will print out the list of logical units (LUNs) supported by the target device. There are a couple of options to modify the output:

-c Just print out a count of LUNs, not the actual LUN numbers.

-l Just print out the LUNs, and do not print out the count.

-r *reporttype* Specify the type of report to request from the target:

default Return the default report. This is the **camcontrol** default. Most targets will support this report if they support the REPORT LUNS command.

wellknown Return only well known LUNs.

all Return all available LUNs.

camcontrol will try to print out LUN numbers in a reasonable format. It can understand the peripheral, flat, LUN and extended LUN formats.

readcap Send the SCSI READ CAPACITY command to the given device and display the results. If the device is larger than 2TB, the SCSI READ CAPACITY (16) service action will be sent to obtain the full size of the device. By default, **camcontrol** will print out the last logical block of the device, and the blocksize of the device in bytes. To modify the output format, use the following options:

-b Just print out the blocksize, not the last block or device size. This cannot be used with **-N** or **-s**.

-h Print out the device size in human readable (base 2, 1K == 1024) format. This implies **-N** and cannot be used with **-q** or **-b**.

-H Print out the device size in human readable (base 10, 1K == 1000) format.

-l Skip sending the SCSI READ CAPACITY (10) command. Send only the SCSI READ CAPACITY (16) service action and report its results. When the two do not match, a quirk is needed to resolve the ambiguity.

-N Print out the number of blocks in the device instead of the last logical block.

-q Quiet, print out the numbers only (separated by a comma if **-b** or **-s** are not specified).

-s Print out the last logical block or the size of the device only, and omit the blocksize.

Note that this command only displays the information, it does not update the kernel data structures. Use the **camcontrol** reprobe subcommand to do that.

start Send the SCSI Start/Stop Unit (0x1B) command to the given device with the start bit set.

stop Send the SCSI Start/Stop Unit (0x1B) command to the given device with the start bit cleared.

load Send the SCSI Start/Stop Unit (0x1B) command to the given device with the start bit set and the load/eject bit set.

eject Send the SCSI Start/Stop Unit (0x1B) command to the given device with the start bit cleared and the load/eject bit set.

rescan Tell the kernel to scan all buses in the system (with the *all* argument), the given bus (XPT_SCAN_BUS), bus:target:lun or device (XPT_SCAN_LUN) for new devices or devices that have gone away. The user may specify a scan of all buses, a single bus, or a lun. Scanning all luns on a target is not supported.

If a device is specified by peripheral name and unit number, for instance da4, it may only be rescanned if that device currently exists in the CAM EDT (Existing Device Table). If the device is no longer there (see **camcontrol** devlist), you must use the bus:target:lun form to rescan it.

reprobe Tell the kernel to refresh the information about the device and notify the upper layer, GEOM(4). This includes sending the SCSI READ CAPACITY command and updating the disk size visible to the rest of the system.

reset Tell the kernel to reset all buses in the system (with the *all* argument), the given bus (XPT_RESET_BUS) by issuing a SCSI bus reset for that bus, or to reset the given bus:target:lun or device (XPT_RESET_DEV), typically by issuing a BUS DEVICE RESET message after connecting to that device. Note that this can have a destructive impact on the system.

defects Send the SCSI READ DEFECT DATA (10) command (0x37) or the SCSI READ DEFECT DATA (12) command (0xB7) to the given device, and print out any combination of: the total number of defects, the primary defect list (PLIST), and the grown defect list (GLIST).

-f *format* Specify the requested format of the defect list. The format argument is required. Most drives support the physical sector format. Some drives support the logical block format. Many drives, if they do not support the requested format, return the data in an alternate format, along with sense information indicating that the requested data format is not supported. The **camcontrol** utility attempts to detect this, and print out whatever format the drive returns. If the drive uses a non-standard sense code to report that it does not support the requested format, **camcontrol** will probably see the error as a failure to complete the request.

The format options are:

block Print out the list as logical blocks. This is limited to 32-bit block sizes, and isn't supported by many modern drives.

longblock Print out the list as logical blocks. This option uses a 64-bit block size.

bfi Print out the list in bytes from index format.

extbfi Print out the list in extended bytes from index format. The extended format allows for ranges of blocks to be printed.

phys Print out the list in physical sector format. Most drives support this format.

extphys Print out the list in extended physical sector format. The extended format allows for ranges of blocks to be printed.

-G Print out the grown defect list. This is a list of bad blocks that have been remapped since the disk left the factory.

-P Print out the primary defect list. This is the list of defects that were present in the factory.

-q When printing status information with **-s**, only print the number of defects.

-s Just print the number of defects, not the list of defects.

-S *offset* Specify the starting offset into the defect list. This implies using the SCSI READ DEFECT DATA (12) command, as the 10 byte version of the command doesn't support the address descriptor index field. Not all drives support the 12 byte command, and some drives that support the 12 byte command don't support the address descriptor index field.

-X Print out defects in hexadecimal (base 16) form instead of base 10 form.

If neither **-P** nor **-G** is specified, **camcontrol** will print out the number of defects given in the READ DEFECT DATA header returned from the drive. Some drives will report 0 defects if neither the primary or grown defect lists are requested.

modepage

Allows the user to display and optionally edit a SCSI mode page. The mode page formats are located in */usr/share/misc/scsi_modes*. This can be overridden by specifying a different file in the SCSI_MODES environment variable. The **modepage** command takes several arguments:

-6 Use 6 byte MODE commands instead of default 10 byte. Old devices may not support 10 byte MODE commands, while new devices may not be able to report all mode pages with 6 byte commands. If not specified, **camcontrol** starts with 10 byte commands and falls back to 6 byte on error.

-d Disable block descriptors for mode sense.

-D Display/edit block descriptors instead of mode page.

-L Use long LBA block descriptors. Allows number of LBAs bigger then 2^^32.

-b Displays mode page data in binary format.

-e This flag allows the user to edit values in the mode page. The user may either edit mode page values with the text editor pointed to by his EDITOR environment variable, or supply mode page values via standard input, using the same format that **camcontrol** uses to display mode page values. The editor will be invoked if **camcontrol** detects that standard input is terminal.

-l Lists all available mode pages. If specified more then once, also lists subpages.

-m *page[,subpage]*
This specifies the number of the mode page and optionally subpage the user would like to view and/or edit. This argument is mandatory unless **-l** is specified.

-P *pgctl* This allows the user to specify the page control field. Possible values are:

0	Current values
1	Changeable values
2	Default values
3	Saved values

cmd Allows the user to send an arbitrary ATA or SCSI CDB to any device. The **cmd** function requires the **-c** argument to specify SCSI CDB or the **-a** argument to specify ATA Command Block registers values. Other arguments are optional, depending on the command type. The command and data specification syntax is documented in cam_cdbparse(3). NOTE: If the CDB specified causes data to be transferred to or from the SCSI device in question, you MUST specify either **-i** or **-o**.

-a *cmd* [args] This specifies the content of 12 ATA Command Block registers (command, features, lba_low, lba_mid, lba_high, device, lba_low_exp, lba_mid_exp. lba_high_exp, features_exp, sector_count, sector_count_exp).

-c *cmd* [args] This specifies the SCSI CDB. SCSI CDBs may be 6, 10, 12 or 16 bytes.

-d Specifies DMA protocol to be used for ATA command.

-f Specifies FPDMA (NCQ) protocol to be used for ATA command.

-i *len fmt* This specifies the amount of data to read, and how it should be displayed. If the format is '-', *len* bytes of data will be read from the device and written to standard output.

-o *len fmt* [args] This specifies the amount of data to be written to a device, and the data that is to be written. If the format is '-', *len* bytes of data will be read from standard input and written to the device.

-r *fmt* This specifies that 11 result ATA Command Block registers should be displayed (status, error, lba_low, lba_mid, lba_high, device, lba_low_exp, lba_mid_exp, lba_high_exp, sector_count, sector_count_exp), and how. If the format is '-', 11 result registers will be written to standard output in hex.

smpcmd Allows the user to send an arbitrary Serial Management Protocol (SMP) command to a device. The **smpcmd** function requires the **-r** argument to specify the SMP request to be sent, and the **-R** argument to specify the format of the SMP response. The syntax for the SMP request and response arguments is documented in cam_cdbparse(3).

Note that SAS adapters that support SMP passthrough (at least the currently known adapters) do not accept CRC bytes from the user in the request and do not pass CRC bytes back to the user in the response. Therefore users should not include the CRC bytes in the length of the request and not expect CRC bytes to be returned in the response.

-r *len fmt* [args] This specifies the size of the SMP request, without the CRC bytes, and the SMP request format. If the format is '-', *len* bytes of data will be read from standard input and written as the SMP request.

-R *len fmt* [args] This specifies the size of the buffer allocated for the SMP response, and the SMP response format. If the format is '-', *len* bytes of data will be allocated for the response and the response will be written to standard output.

smprg Allows the user to send the Serial Management Protocol (SMP) Report General command to a device. **camcontrol** will display the data returned by the Report General command. If the SMP target supports the long response format, the additional data will be requested and displayed automatically.

-l Request the long response format only. Not all SMP targets support the long response format. This option causes **camcontrol** to skip sending the initial report general request without the long bit set and only issue a report general request with the long bit set.

smppc Allows the user to issue the Serial Management Protocol (SMP) PHY Control command to a device. This function should be used with some caution, as it can render devices inaccessible, and could potentially cause data corruption as well. The **-p** argument is required to specify the PHY to operate on.

-p *phy* Specify the PHY to operate on. This argument is required.

-l Request the long request/response format. Not all SMP targets support the long response format. For the PHY Control command, this currently only affects whether the request length is set to a value other than 0.

-o *operation* Specify a PHY control operation. Only one **-o** operation may be specified. The operation may be specified numerically (in decimal, hexadecimal, or octal) or one of the following operation names may be specified:

nop No operation. It is not necessary to specify this argument.

linkreset Send the LINK RESET command to the phy.

hardreset Send the HARD RESET command to the phy.

disable Send the DISABLE command to the phy. Note that the LINK RESET or HARD RESET commands should re-enable the phy.

clearerrlog Send the CLEAR ERROR LOG command. This clears the error log counters for the specified phy.

clearaffiliation — Send the CLEAR AFFILIATION command. This clears the affiliation from the STP initiator port with the same SAS address as the SMP initiator that requests the clear operation.

sataportsel — Send the TRANSMIT SATA PORT SELECTION SIGNAL command to the phy. This will cause a SATA port selector to use the given phy as its active phy and make the other phy inactive.

clearitnl — Send the CLEAR STP I_T NEXUS LOSS command to the PHY.

setdevname — Send the SET ATTACHED DEVICE NAME command to the PHY. This requires the **-d** argument to specify the device name.

-d *name* — Specify the attached device name. This option is needed with the **-o** *setdevname* phy operation. The name is a 64-bit number, and can be specified in decimal, hexadecimal or octal format.

-m *rate* — Set the minimum physical link rate for the phy. This is a numeric argument. Currently known link rates are:

0x0	Do not change current value.
0x8	1.5 Gbps
0x9	3 Gbps
0xa	6 Gbps

Other values may be specified for newer physical link rates.

-M *rate* — Set the maximum physical link rate for the phy. This is a numeric argument. See the **-m** argument description for known link rate arguments.

-T *pp_timeout* — Set the partial pathway timeout value, in microseconds. See the ANSI SAS Protocol Layer (SPL) specification for more information on this field.

-a *enable|disable* Enable or disable SATA slumber phy power conditions.

-A *enable|disable* Enable or disable SATA partial power conditions.

-s *enable|disable* Enable or disable SAS slumber phy power conditions.

-S *enable|disable* Enable or disable SAS partial phy power conditions.

smpphylist

List phys attached to a SAS expander, the address of the end device attached to the phy, and the inquiry data for that device and peripheral devices attached to that device. The inquiry data and peripheral devices are displayed if available.

-l Turn on the long response format for the underlying SMP commands used for this command.

-q Only print out phys that are attached to a device in the CAM EDT (Existing Device Table).

smpmaninfo

Send the SMP Report Manufacturer Information command to the device and display the response.

-l Turn on the long response format for the underlying SMP commands used for this command.

debug Turn on CAM debugging printfs in the kernel. This requires options CAMDEBUG in your kernel config file. WARNING: enabling debugging printfs currently causes an EXTREME number of kernel printfs. You may have difficulty turning off the debugging printfs once they start, since the kernel will be busy printing messages and unable to service other requests quickly. The **debug** function takes a number of arguments:

-I Enable CAM_DEBUG_INFO printfs.

-P Enable CAM_DEBUG_PERIPH printfs.

-T Enable CAM_DEBUG_TRACE printfs.

-S Enable CAM_DEBUG_SUBTRACE printfs.

-X Enable CAM_DEBUG_XPT printfs.

-c Enable CAM_DEBUG_CDB printfs. This will cause the kernel to print out the SCSI CDBs sent to the specified device(s).

-p Enable CAM_DEBUG_PROBE printfs.

all Enable debugging for all devices.

off Turn off debugging for all devices

bus[:target[:lun]] Turn on debugging for the given bus, target or lun. If the lun or target and lun are not specified, they are wildcarded. (i.e., just specifying a bus turns on debugging printfs for all devices on that bus.)

tags Show or set the number of "tagged openings" or simultaneous transactions we attempt to queue to a particular device. By default, the **tags** command, with no command-specific arguments (i.e., only generic arguments) prints out the "soft" maximum number of transactions that can be queued to the device in question. For more detailed information, use the **-v** argument described below.

-N *tags* Set the number of tags for the given device. This must be between the minimum and maximum number set in the kernel quirk table. The default for most devices that support tagged queueing is a minimum of 2 and a maximum of 255. The minimum and maximum values for a given device may be determined by using the **-v** switch. The meaning of the **-v** switch for this **camcontrol** subcommand is described below.

-q Be quiet, and do not report the number of tags. This is generally used when setting the number of tags.

-v The verbose flag has special functionality for the *tags* argument. It causes **camcontrol** to print out the tagged queueing related fields of the XPT_GDEV_TYPE CCB:

dev_openings This is the amount of capacity for transactions queued to a given device.

dev_active This is the number of transactions currently queued to a device.

allocated This is the number of CCBs allocated for the device.

held The held count is the number of CCBs held by peripheral drivers that have either just been completed or are about to be released to the transport layer for service by a device. Held CCBs reserve capacity on a given device.

mintags This is the current "hard" minimum number of transactions that can be queued to a device at once. The *dev_openings* value above cannot go below this number. The default value for *mintags* is 2, although it may be set higher or lower for various devices.

maxtags This is the "hard" maximum number of transactions that can be queued to a device at one time. The *dev_openings* value cannot go above this number. The default value for *maxtags* is 255, although it may be set higher or lower for various devices.

negotiate Show or negotiate various communication parameters. Some controllers may not support setting or changing some of these values. For instance, the Adaptec 174x controllers do not support changing a device’s sync rate or offset. The **camcontrol** utility will not attempt to set the parameter if the controller indicates that it does not support setting the parameter. To find out what the controller supports, use the **-v** flag. The meaning of the **-v** flag for the **negotiate** command is described below. Also, some controller drivers do not support setting negotiation parameters, even if the underlying controller supports negotiation changes. Some controllers, such as the Advansys wide controllers, support enabling and disabling synchronous negotiation for a device, but do not support setting the synchronous negotiation rate.

-a Attempt to make the negotiation settings take effect immediately by sending a Test Unit Ready command to the device.

-c Show or set current negotiation settings. This is the default.

-D *enable|disable* Enable or disable disconnection.

-M *mode* Set ATA mode.

-O *offset* Set the command delay offset.

-q Be quiet, do not print anything. This is generally useful when you want to set a parameter, but do not want any status information.

-R *syncrate* Change the synchronization rate for a device. The sync rate is a floating point value specified in MHz. So, for instance, '20.000' is a legal value, as is '20'.

-T *enable*|*disable* Enable or disable tagged queueing for a device.

-U Show or set user negotiation settings. The default is to show or set current negotiation settings.

-v The verbose switch has special meaning for the **negotiate** subcommand. It causes **camcontrol** to print out the contents of a Path Inquiry (XPT_PATH_INQ) CCB sent to the controller driver.

-W *bus_width* Specify the bus width to negotiate with a device. The bus width is specified in bits. The only useful values to specify are 8, 16, and 32 bits. The controller must support the bus width in question in order for the setting to take effect.

In general, sync rate and offset settings will not take effect for a device until a command has been sent to the device. The **-a** switch above will automatically send a Test Unit Ready to the device so negotiation parameters will take effect.

format Issue the SCSI FORMAT UNIT command to the named device.

WARNING! WARNING! WARNING!

Low level formatting a disk will destroy ALL data on the disk. Use extreme caution when issuing this command. Many users low-level format disks that do not really need to be low-level formatted. There are relatively few scenarios that call for low-level formatting a disk. One reason for low-level formatting a disk is to initialize the disk after changing its physical sector size. Another reason for low-level formatting a disk is to revive the disk if you are getting "medium format corrupted" errors from the disk in response to read and write requests.

Some disks take longer than others to format. Users should specify a timeout long enough to allow the format to complete. The default format timeout is 3 hours, which should be long enough for most disks. Some hard disks will complete a format operation in a very short period of time (on the order of 5 minutes or less). This is often because the drive does not really support the FORMAT UNIT command **--** it just accepts the command, waits a few minutes and then returns it.

The ‘format’ subcommand takes several arguments that modify its default behavior. The **-q** and **-y** arguments can be useful for scripts.

-q Be quiet, do not print any status messages. This option will not disable the questions, however. To disable questions, use the **-y** argument, below.

-r Run in "report only" mode. This will report status on a format that is already running on the drive.

-w Issue a non-immediate format command. By default, **camcontrol** issues the FORMAT UNIT command with the immediate bit set. This tells the device to immediately return the format command, before the format has actually completed. Then, **camcontrol** gathers SCSI sense information from the device every second to determine how far along in the format process it is. If the **-w** argument is specified, **camcontrol** will issue a non-immediate format command, and will be unable to print any information to let the user know what percentage of the disk has been formatted.

-y Do not ask any questions. By default, **camcontrol** will ask the user if he/she really wants to format the disk in question, and also if the default format command timeout is acceptable. The user will not be asked about the timeout if a timeout is specified on the command line.

sanitize Issue the SANITIZE command to the named device.

WARNING! WARNING! WARNING!

ALL data on the disk will be destroyed or made inaccessible. Recovery of the data is not possible. Use extreme caution when issuing this command.

The ‘sanitize’ subcommand takes several arguments that modify its default behavior. The **-q** and **-y** arguments can be useful for scripts.

-a *operation*
Specify the sanitize operation to perform.

overwrite Perform an overwrite operation by writing a user supplied data pattern to the device one or more times. The pattern is given by the **-P** argument. The number of times is given by the **-c** argument.

block — Perform a block erase operation. All the device's blocks are set to a vendor defined value, typically zero.

crypto — Perform a cryptographic erase operation. The encryption keys are changed to prevent the decryption of the data.

exitfailure — Exits a previously failed sanitize operation. A failed sanitize operation can only be exited if it was run in the unrestricted completion mode, as provided by the **-U** argument.

-c *passes*
The number of passes when performing an 'overwrite' operation. Valid values are between 1 and 31. The default is 1.

-I When performing an 'overwrite' operation, the pattern is inverted between consecutive passes.

-P *pattern*
Path to the file containing the pattern to use when performing an 'overwrite' operation. The pattern is repeated as needed to fill each block.

-q Be quiet, do not print any status messages. This option will not disable the questions, however. To disable questions, use the **-y** argument, below.

-U Perform the sanitize in the unrestricted completion mode. If the operation fails, it can later be exited with the 'exitfailure' operation.

-r Run in "report only" mode. This will report status on a sanitize that is already running on the drive.

-w Issue a non-immediate sanitize command. By default, **camcontrol** issues the SANITIZE command with the immediate bit set. This tells the device to immediately return the sanitize command, before the sanitize has actually completed. Then, **camcontrol** gathers SCSI sense information from the device every second to determine how far along in the sanitize process it is. If the **-w** argument is specified, **camcontrol** will issue a non-immediate sanitize command, and will be unable to print any information to let the user know what percentage of the disk has been sanitized.

-y Do not ask any questions. By default, **camcontrol** will ask the user if he/she really wants to sanitize the disk in question, and also if the default sanitize command timeout is acceptable. The user will not be asked about the timeout if a timeout is specified on the command line.

idle Put ATA device into IDLE state. Optional parameter (**-t**) specifies automatic standby timer value in seconds. Value 0 disables timer.

standby Put ATA device into STANDBY state. Optional parameter (**-t**) specifies automatic standby timer value in seconds. Value 0 disables timer.

sleep Put ATA device into SLEEP state. Note that the only way get device out of this state may be reset.

powermode
Report ATA device power mode.

apm It optional parameter (**-l**) specified, enables and sets advanced power management level, where 1 -- minimum power, 127 -- maximum performance with standby, 128 -- minimum power without standby, 254 -- maximum performance. If not specified -- APM is disabled.

aam It optional parameter (**-l**) specified, enables and sets automatic acoustic management level, where 1 -- minimum noise, 254 -- maximum performance. If not specified -- AAM is disabled.

security Update or report security settings, using an ATA identify command (0xec). By default, **camcontrol** will print out the security support and associated settings of the device. The **security** command takes several arguments:

-d *pwd*

Disable device security using the given password for the selected user according to the devices configured security level.

-e *pwd*

Erase the device using the given password for the selected user.

WARNING! WARNING! WARNING!

Issuing a secure erase will *ERASE ALL* user data on the device and may take several hours to complete.

When this command is used against an SSD drive all its cells will be marked as empty, restoring it to factory default write performance. For SSD's this action usually takes just a few seconds.

-f

Freeze the security configuration of the specified device.

After command completion any other commands that update the device lock mode shall be command aborted. Frozen mode is disabled by power-off or hardware reset.

-h *pwd*

Enhanced erase the device using the given password for the selected user.

WARNING! WARNING! WARNING!

Issuing an enhanced secure erase will *ERASE ALL* user data on the device and may take several hours to complete.

An enhanced erase writes predetermined data patterns to all user data areas, all previously written user data shall be overwritten, including sectors that are no longer in use due to reallocation.

-k *pwd*

Unlock the device using the given password for the selected user according to the devices configured security level.

-l *high|maximum*

Specifies which security level to set when issuing a **-s** *pwd* command. The security level determines device behavior when the master password is used to unlock the device. When the security level is set to high the device requires the unlock command and the master password to unlock. When the security level is set to maximum the device requires a secure erase with the master password to unlock.

This option must be used in conjunction with one of the security action commands.

Defaults to *high*

-q

Be quiet, do not print any status messages. This option will not disable the questions, however. To disable questions, use the **-y** argument, below.

-s *pwd*

Password the device (enable security) using the given password for the selected user. This option can be combined with other options such as **-e** *pwd*

A master password may be set in a addition to the user password. The purpose of the master password is to allow an administrator to establish a password that is kept secret from the user, and which may be used to unlock the device if the user password is lost.

Note: Setting the master password does not enable device security.

If the master password is set and the drive supports a Master Revision Code feature the Master Password Revision Code will be decremented.

-T *timeout*

Overrides the default timeout, specified in seconds, used for both **-e** and **-h** this is useful if your system has problems processing long timeouts correctly.

Usually the timeout is calculated from the information stored on the drive if present, otherwise it defaults to 2 hours.

-U *user|master*

Specifies which user to set / use for the running action command, valid values are user or master and defaults to master if not set.

This option must be used in conjunction with one of the security action commands.

Defaults to *master*

-y

Confirm yes to dangerous options such as **-e** without prompting for confirmation.

If the password specified for any action commands does not match the configured password for the specified user the command will fail.

The password in all cases is limited to 32 characters, longer passwords will fail.

hpa Update or report Host Protected Area details. By default **camcontrol** will print out the HPA support and associated settings of the device. The **hpa** command takes several optional arguments:

-f

Freeze the HPA configuration of the specified device.

After command completion any other commands that update the HPA configuration shall be command aborted. Frozen mode is disabled by power-off or hardware reset.

-l

Lock the HPA configuration of the device until a successful call to unlock or the next power-on reset occurs.

-P

Make the HPA max sectors persist across power-on reset or a hardware reset. This must be used in combination with **-s** *max_sectors*

-p *pwd*

Set the HPA configuration password required for unlock calls.

-q

Be quiet, do not print any status messages. This option will not disable the questions. To disable questions, use the **-y** argument, below.

-s *max_sectors*

Configures the maximum user accessible sectors of the device. This will change the number of sectors the device reports.

WARNING! WARNING! WARNING!

Changing the max sectors of a device using this option will make the data on the device beyond the specified value inaccessible.

Only one successful **-s** *max_sectors* call can be made without a power-on reset or a hardware reset of the device.

-U *pwd*

Unlock the HPA configuration of the specified device using the given password. If the password specified does not match the password configured via **-p** *pwd* the command will fail.

After 5 failed unlock calls, due to password miss-match, the device will refuse additional unlock calls until after a power-on reset.

-y

Confirm yes to dangerous options such as **-e** without prompting for confirmation

The password for all HPA commands is limited to 32 characters, longer passwords will fail.

ama Update or report Accessible Max Address Configuration. By default **camcontrol** will print out the Accessible Max Address Configuration support and associated settings of the device. The **ama** command takes several optional arguments:

-f

Freeze the Accessible Max Address Configuration of the specified device.

After command completion any other commands that update the configuration shall be command aborted. Frozen mode is disabled by power-off.

-q

Be quiet, do not print any status messages.

-s *max_sectors*

Configures the maximum user accessible sectors of the device. This will change the number of sectors the device reports.

WARNING! WARNING! WARNING!

Changing the max sectors of a device using this option will make the data on the device beyond the specified value indeterminate.

Only one successful **-s** *max_sectors* call can be made without a power-on reset of the device.

fwdownload

Program firmware of the named SCSI or ATA device using the image file provided.

If the device is a SCSI device and it provides a recommended timeout for the WRITE BUFFER command (see the **camcontrol** opcodes subcommand), that timeout will be used for the firmware download. The drive-recommended timeout value may be overridden on the command line with the **-t** option.

Current list of supported vendors for SCSI/SAS drives:

HGST Tested with 4TB SAS drives, model number HUS724040ALS640.

HITACHI

HP

IBM Tested with LTO-5 (ULTRIUM-HH5) and LTO-6 (ULTRIUM-HH6) tape drives. There is a separate table entry for hard drives, because the update method for hard drives is different than the method for tape drives.

PLEXTOR

QUALSTAR

QUANTUM

SAMSUNG Tested with SM1625 SSDs.

SEAGATE Tested with Constellation ES (ST32000444SS), ES.2 (ST33000651SS) and ES.3 (ST1000NM0023) drives.

SmrtStor Tested with 400GB Optimus SSDs (TXA2D20400GA6001).

WARNING! WARNING! WARNING!

Little testing has been done to make sure that different device models from each vendor work correctly with the fwdownload command. A vendor name appearing in the supported list means only that firmware of at least one device type from that vendor has successfully been programmed with the fwdownload command. Extra caution should be taken when using this command since there is no guarantee it will not break a device from the listed vendors. Ensure that you have a recent backup of the data on the device before performing a firmware update.

Note that unknown SCSI protocol devices will not be programmed, since there is little chance of the firmware download succeeding.

camcontrol will currently attempt a firmware download to any ATA or SATA device, since the standard ATA DOWNLOAD MICROCODE command may work. Firmware downloads to ATA and SATA devices are supported for devices connected to standard ATA and SATA controllers, and devices connected to SAS controllers with SCSI to ATA translation capability. In the latter case, **camcontrol** uses the SCSI ATA PASS-THROUGH command to send the ATA DOWNLOAD MICROCODE command to the drive. Some SCSI to ATA translation implementations don't work fully when translating SCSI WRITE BUFFER commands to ATA DOWNLOAD MICROCODE commands, but do support ATA passthrough well enough to do a firmware download.

-f *fw_image* Path to the firmware image file to be downloaded to the specified device.

-q Do not print informational messages, only print errors. This option should be used with the **-y** option to suppress all output.

-s Run in simulation mode. Device checks are run and the confirmation dialog is shown, but no firmware download will occur.

-v Show SCSI or ATA errors in the event of a failure.

In simulation mode, print out the SCSI CDB or ATA register values that would be used for the firmware download command.

-y Do not ask for confirmation.

persist Persistent reservation support. Persistent reservations are a way to reserve a particular SCSI LUN for use by one or more SCSI initiators. If the **-i** option is specified, **camcontrol** will issue the SCSI PERSISTENT RESERVE IN command using the requested service action. If the **-o** option is specified, **camcontrol** will issue the SCSI PERSISTENT RESERVE OUT command using the requested service action. One of those two options is required.

Persistent reservations are complex, and fully explaining them is outside the scope of this manual. Please visit https://www.t10.org and download the latest SPC spec for a full explanation of persistent reservations.

-i *mode* Specify the service action for the PERSISTENT RESERVE IN command. Supported service actions:

read_keys Report the current persistent reservation generation (PRgeneration) and any registered keys.

read_reservation Report the persistent reservation, if any.

report_capabilities Report the persistent reservation capabilities of the LUN.

read_full_status Report the full status of persistent reservations on the LUN.

-o *mode* Specify the service action for the PERSISTENT RESERVE OUT command. For service actions like register that are components of other service action names, the entire name must be specified. Otherwise, enough of the service action name must be specified to distinguish it from other possible service actions. Supported service actions:

register Register a reservation key with the LUN or unregister a reservation key. To register a key, specify the requested key as the Service Action Reservation Key. To unregister a key, specify the previously registered key as the Reservation Key. To change a key, specify the old key as the Reservation Key and the new key as the Service Action Reservation Key.

register_ignore
: This is similar to the register subcommand, except that the Reservation Key is ignored. The Service Action Reservation Key will overwrite any previous key registered for the initiator.

reserve
: Create a reservation. A key must be registered with the LUN before the LUN can be reserved, and it must be specified as the Reservation Key. The type of reservation must also be specified. The scope defaults to LUN scope (LU_SCOPE), but may be changed.

release
: Release a reservation. The Reservation Key must be specified.

clear
: Release a reservation and remove all keys from the device. The Reservation Key must be specified.

preempt
: Remove a reservation belonging to another initiator. The Reservation Key must be specified. The Service Action Reservation Key may be specified, depending on the operation being performed.

preempt_abort
: Remove a reservation belonging to another initiator and abort all outstanding commands from that initiator. The Reservation Key must be specified. The Service Action Reservation Key may be specified, depending on the operation being performed.

register_move
: Register another initiator with the LUN, and establish a reservation on the LUN for that initiator. The Reservation Key and Service Action Reservation Key must be specified.

replace_lost
: Replace Lost Reservation information.

-a
: Set the All Target Ports (ALL_TG_PT) bit. This requests that the key registration be applied to all target ports and not just the particular target port that receives the command. This only applies to the register and register_ignore actions.

-I *tid*
: Specify a Transport ID. This only applies to the Register and Register and Move service actions for Persistent Reserve Out. Multiple Transport IDs may be specified with multiple **-I** arguments. With the Register service action, specifying

one or more Transport IDs implicitly enables the **-S** option which turns on the SPEC_I_PT bit. Transport IDs generally have the format protocol,id.

SAS A SAS Transport ID consists of "sas," followed by a 64-bit SAS address. For example:

```
sas,0x1234567812345678
```

FC A Fibre Channel Transport ID consists of "fcp," followed by a 64-bit Fibre Channel World Wide Name. For example:

```
fcp,0x1234567812345678
```

SPI A Parallel SCSI address consists of "spi," followed by a SCSI target ID and a relative target port identifier. For example:

```
spi,4,1
```

1394 An IEEE 1394 (Firewire) Transport ID consists of "sbp," followed by a 64-bit EUI-64 IEEE 1394 node unique identifier. For example:

```
sbp,0x1234567812345678
```

RDMA

A SCSI over RDMA Transport ID consists of "srp," followed by a 128-bit RDMA initiator port identifier. The port identifier must be exactly 32 or 34 (if the leading 0x is included) hexadecimal digits. Only hexadecimal (base 16) numbers are supported. For example:

```
srp,0x12345678123456781234567812345678
```

iSCSI An iSCSI Transport ID consists an iSCSI name and optionally a separator and iSCSI session ID. For example, if only the iSCSI name is specified:

```
iqn.2012-06.com.example:target0
```

If the iSCSI separator and initiator session ID are specified:

```
iqn.2012-06.com.example:target0,i,0x123
```

PCIe A SCSI over PCIe Transport ID consists of "sop," followed by a PCIe Routing ID. The Routing ID consists of a bus, device and function or in the alternate form, a bus and function. The bus must be in the range of 0 to 255 inclusive and the device must be in the range of 0 to 31 inclusive. The function must be in the range of 0 to 7 inclusive if the standard form is used, and in the range of 0 to 255 inclusive if the alternate form is used. For example, if a bus, device and function are specified for the standard Routing ID form:

```
sop,4,5,1
```

If the alternate Routing ID form is used:

```
sop,4,1
```

-k *key* Specify the Reservation Key. This may be in decimal, octal or hexadecimal format. The value is zero by default if not otherwise specified. The value must be between 0 and 2^64 - 1, inclusive.

-K *key* Specify the Service Action Reservation Key. This may be in decimal, octal or hexadecimal format. The value is zero by default if not otherwise specified. The value must be between 0 and 2^64 - 1, inclusive.

-p Enable the Activate Persist Through Power Loss bit. This is only used for the register and register_ignore actions. This requests that the reservation persist across power loss events.

-s *scope* Specify the scope of the reservation. The scope may be specified by name or by number. The scope is ignored for register, register_ignore and clear. If the desired scope isn't available by name, you may specify the number.

lun LUN scope (0x00). This encompasses the entire LUN.

extent Extent scope (0x01).

element Element scope (0x02).

-R *rtp* Specify the Relative Target Port. This only applies to the Register and Move service action of the Persistent Reserve Out command.

-S Enable the SPEC_I_PT bit. This only applies to the Register service action of Persistent Reserve Out. You must also specify at least one Transport ID with **-I** if this option is set. If you specify a Transport ID, this option is automatically set. It is an error to specify this option for any service action other than Register.

-T *type* Specify the reservation type. The reservation type may be specified by name or by number. If the desired reservation type isn't available by name, you may specify the number. Supported reservation type names:

read_shared Read Shared mode.

wr_ex Write Exclusive mode. May also be specified as "write_exclusive".

rd_ex Read Exclusive mode. May also be specified as "read_exclusive".

ex_ac Exclusive access mode. May also be specified as "exclusive_access".

wr_ex_ro Write Exclusive Registrants Only mode. May also be specified as "write_exclusive_reg_only".

ex_ac_ro Exclusive Access Registrants Only mode. May also be specified as "exclusive_access_reg_only".

wr_ex_ar Write Exclusive All Registrants mode. May also be specified as "write_exclusive_all_regs".

ex_ac_ar Exclusive Access All Registrants mode. May also be specified as "exclusive_access_all_regs".

-U Specify that the target should unregister the initiator that sent the Register and Move request. By default, the target will not unregister the initiator that sends the Register and Move request. This option only applies to the Register and Move service action of the Persistent Reserve Out command.

attrib Issue the SCSI READ or WRITE ATTRIBUTE commands. These commands are used to read and write attributes in Medium Auxiliary Memory (MAM). The most common place Medium Auxiliary Memory is found is small flash chips included tape cartriges. For instance, LTO tapes have MAM. Either the **-r** option or the **-w** option must be specified.

-r *action* Specify the READ ATTRIBUTE service action.

attr_values Issue the ATTRIBUTE VALUES service action. Read and decode the available attributes and their values.

attr_list Issue the ATTRIBUTE LIST service action. List the attributes that are available to read and write.

lv_list Issue the LOGICAL VOLUME LIST service action. List the available logical volumes in the MAM.

part_list Issue the PARTITION LIST service action. List the available partitions in the MAM.

supp_attr Issue the SUPPORTED ATTRIBUTES service action. List attributes that are supported for reading or writing. These attributes may or may not be currently present in the MAM.

-w *attr* Specify an attribute to write to the MAM. This option is not yet implemented.

-a *num* Specify the attribute number to display. This option only works with the attr_values, attr_list and supp_attr arguments to **-r**.

-c Display cached attributes. If the device supports this flag, it allows displaying attributes for the last piece of media loaded in the drive.

-e *num* Specify the element address. This is used for specifying which element number in a medium changer to access when reading attributes. The element number could be for a picker, portal, slot or drive.

-F *form1,form2* Specify the output format for the attribute values (attr_val) display as a comma separated list of options. The default output is currently set to field_all,nonascii_trim,text_raw. Once this code is ported to FreeBSD 10, any text fields will be converted from their codeset to the user's native codeset with iconv(3).

The text options are mutually exclusive; if you specify more than one, you will get unpredictable results. The nonascii options are also mutually exclusive. Most of the field options may be logically ORed together.

text_esc Print text fields with non-ASCII characters escaped.

text_raw Print text fields natively, with no codeset conversion.

nonascii_esc If any non-ASCII characters occur in fields that are supposed to be ASCII, escape the non-ASCII characters.

nonascii_trim If any non-ASCII characters occur in fields that are supposed to be ASCII, omit the non-ASCII characters.

nonascii_raw If any non-ASCII characters occur in fields that are supposed to be ASCII, print them as they are.

field_all Print all of the prefix fields: description, attribute number, attribute size, and the attribute’s readonly status. If field_all is specified, specifying any other field options will not have an effect.

field_none Print none of the prefix fields, and only print out the attribute value. If field_none is specified, specifying any other field options will result in those fields being printed.

field_desc Print out the attribute description.

field_num Print out the attribute number.

field_size Print out the attribute size.

field_rw Print out the attribute’s readonly status.

-p *part* Specify the partition. When the media has multiple partitions, specifying different partition numbers allows seeing the values for each individual partition.

-s *start_num* Specify the starting attribute number. This requests that the target device return attribute information starting at the given number.

-T *elem_type* Specify the element type. For medium changer devices, this allows specifying the type the element referenced in the element address (**-e**). Valid types are: "all", "picker", "slot", "portal", and "drive".

-V *vol_num* Specify the number of the logical volume to operate on. If the media has multiple logical volumes, this will allow displaying or writing attributes on the given logical volume.

opcodes Issue the REPORT SUPPORTED OPCODES service action of the SCSI MAINTENANCE IN command. Without arguments, this command will return a list of all SCSI commands supported by the device, including service actions of commands that support service actions. It will also include the SCSI CDB (Command Data Block) length for each command, and the description of each command if it is known.

-o *opcode* Request information on a specific opcode instead of the list of supported commands. If supported, the target will return a CDB-like structure that indicates the opcode, service action (if any), and a mask of bits that are supported in that CDB.

-s *service_action* For commands that support a service action, specify the service action to query.

-N If a service action is specified for a given opcode, and the device does not support the given service action, the device should not return a SCSI error, but rather indicate in the returned parameter data that the command is not supported. By default, if a service action is specified for an opcode, and service actions are not supported for the opcode in question, the device will return an error.

-T Include timeout values. This option works with the default display, which includes all commands supported by the device, and with the **-o** and **-s** options, which request information on a specific command and service action. This requests that the device report Nominal and Recommended timeout values for the given command or commands. The timeout values are in seconds. The timeout descriptor also includes a command-specific

zone Manage SCSI and ATA Zoned Block devices. This allows managing devices that conform to the SCSI Zoned Block Commands (ZBC) and ATA Zoned ATA Command Set (ZAC) specifications. Devices using these command sets are usually hard drives using Shingled Magnetic Recording (SMR). There are three types of SMR drives:

Drive Managed Drive Managed drives look and act just like a standard random access block device, but underneath, the drive reads and writes the bulk of its capacity using SMR zones. Sequential writes will yield better performance, but writing sequentially is not required.

Host Aware Host Aware drives expose the underlying zone layout via SCSI or ATA commands and allow the host to manage the zone conditions. The host is not required to manage the zones on the drive, though. Sequential writes will yield better performance in Sequential Write Preferred zones, but the host can write randomly in those zones.

Host Managed Host Managed drives expose the underlying zone layout via SCSI or ATA commands. The host is required to access the zones according to the rules described by the zone layout. Any commands that violate the rules will be returned with an error.

SMR drives are divided into zones (typically in the range of 256MB each) that fall into three general categories:

Conventional These are also known as Non Write Pointer zones. These zones can be randomly written without an unexpected performance penalty.

Sequential Preferred These zones should be written sequentially starting at the write pointer for the zone. They may be written randomly. Writes that do not conform to the zone layout may be significantly slower than expected.

Sequential Required These zones must be written sequentially. If they are not written sequentially, starting at the write pointer, the command will fail.

-c *cmd* Specify the zone subcommand:

rz Issue the Report Zones command. All zones are returned by default. Specify report options with **-o** and printing options with **-P**. Specify the starting LBA with **-l**. Note that "reportzones" is also accepted as a command argument.

open Explicitly open the zone specified by the starting LBA.

close Close the zone specified by starting LBA.

finish Finish the zone specified by the starting LBA.

rwp Reset the write pointer for the zone specified by the starting LBA.

-a For the Open, Close, Finish, and Reset Write Pointer operations, apply the operation to all zones on the drive.

-l *lba* Specify the starting LBA. For the Report Zones command, this tells the drive to report starting with the zone that starts at the given LBA. For the other commands, this allows the user to identify the zone requested by its starting LBA. The LBA may be specified in decimal, hexadecimal or octal notation.

-o *rep_opt* For the Report Zones command, specify a subset of zones to report.

all Report all zones. This is the default.

emtpy Report only empty zones.

imp_open Report zones that are implicitly open. This means that the host has sent a write to the zone without explicitly opening the zone.

exp_open Report zones that are explicitly open.

closed Report zones that have been closed by the host.

full Report zones that are full.

ro Report zones that are in the read only state. Note that "readonly" is also accepted as an argument.

offline Report zones that are in the offline state.

reset Report zones where the device recommends resetting write pointers.

nonseq Report zones that have the Non Sequential Resources Active flag set. These are zones that are Sequential Write Preferred, but have been written non-sequentially.

nonwp Report Non Write Pointer zones, also known as Conventional zones.

-P *print_opt* Specify a printing option for Report Zones:

normal Normal Report Zones output. This is the default. The summary and column headings are printed, fields are separated by spaces and the fields themselves may contain spaces.

summary

Just print the summary: the number of zones, the maximum LBA (LBA of the last logical block on the drive), and the value of the "same" field. The "same" field describes whether the zones on the drive are all identical, all different, or whether they are the same except for the last zone, etc.

script Print the zones in a script friendly format. The summary and column headings are omitted, the fields are separated by commas, and the fields do not contain spaces. The fields contain underscores where spaces would normally be used.

epc Issue ATA Extended Power Conditions (EPC) feature set commands. This only works on ATA protocol drives, and will not work on SCSI protocol drives. It will work on SATA drives behind a SCSI to ATA translation layer (SAT). It may be helpful to read the ATA Command Set - 4 (ACS-4) description of the Extended Power Conditions feature set, available at t13.org, to understand the details of this particular **camcontrol** subcommand.

-c *cmd* Specify the epc subcommand

restore Restore drive power condition settings.

-r *src* Specify the source for the restored power settings, either "default" or "saved". This argument is required.

-s Save the settings. This only makes sense to specify when restoring from defaults.

goto Go to the specified power condition.

-p *cond* Specify the power condition: Idle_a, Idle_b, Idle_c, Standby_y, Standby_z. This argument is required.

-D Specify delayed entry to the power condition. The drive, if it supports this, can enter the power condition after the command completes.

-H Hold the power condition. If the drive supports this option, it will hold the power condition and reject all commands that would normally cause it to exit that power condition.

timer Set the timer value for a power condition and enable or disable the condition. See the "list" display described below to see what the current timer settings are for each Idle and Standby mode supported by the drive.

-e Enable the power condition. One of **-e** or **-d** is required.

-d Disable the power condition. One of **-d** or **-e** is required.

-T *timer* Specify the timer in seconds. The user may specify a timer as a floating point number with a maximum supported resolution of tenths of a second. Drives may or may not support sub-second timer values.

-p *cond* Specify the power condition: Idle_a, Idle_b, Idle_c, Standby_y, Standby_z. This argument is required.

-s Save the timer and power condition enable/disable state. By default, if this option is not specified, only the current values for this power condition will be affected.

state Enable or disable a particular power condition.

-e Enable the power condition. One of **-e** or **-d** is required.

-d Disable the power condition. One of **-d** or **-e** is required.

-p *cond* Specify the power condition: Idle_a, Idle_b, Idle_c, Standby_y, Standby_z. This argument is required.

-s Save the power condition enable/disable state. By default, if this option is not specified, only the current values for this power condition will be affected.

enable Enable the Extended Power Condition (EPC) feature set.

disable Disable the Extended Power Condition (EPC) feature set.

source Specify the EPC power source.

-S *src* Specify the power source, either "battery" or "nonbattery".

status Get the current status of several parameters related to the Extended Power Condition (EPC) feature set, including whether APM and EPC are supported and enabled, whether Low Power Standby is supported, whether setting the EPC power source is supported, whether Low Power Standby is supported and the current power condition.

-P Only report the current power condition. Some drives will exit their current power condition if a command other than the ATA CHECK POWER MODE command is received. If this flag is specified, **camcontrol** will only issue the ATA CHECK POWER MODE command to the drive.

list Display the ATA Power Conditions log (Log Address 0x08). This shows the list of Idle and Standby power conditions the drive supports, and a number of parameters about each condition, including whether it is enabled and what the timer value is.

timestamp

Issue REPORT TIMESTAMP or SET TIMESTAMP SCSI commands. Either the **-r** option or the **-s** option must be specified.

-r Report the device's timestamp. If no more arguments are specified, the timestamp will be reported using the national representation of the date and time, followed by the time zone.

-f *format* Specify the strftime format string, as documented in strftime(3), to be used to format the reported timestamp.

-m Report the timestamp as milliseconds since the epoch.

-U Report the timestamp using the national representation of the date and time, but override the system time zone and use UTC instead.

-s Set the device’s timestamp. Either the **-f** and **-T** options or the **-U** option must be specified.

-f *format* Specify the strptime format string, as documented in strptime(3). The time must also be specified with the **-T** option.

-T *time* Provide the time in the format specified with the **-f** option.

-U Set the timestamp to the host system’s time in UTC.

devtype Print out the device type for specified device.

ata	An ATA device attached directly to an ATA controller
satl	An SATA device attached behind a SAS controller via SCSI-ATA Translation Layer (SATL)
scsi	A SCSI device
nvme	An directly attached NVMe device
mmcsd	An MMC or SD device attached via a mmcsd bus
none	No device type reported
unknown	Device type is unknown
illegal	A programming error occurred

depop Commands necessary to support the depopulation (depop) of defective elements of a device (typically heads for hard drives) or setting capacity point (typically used on flash drives). Issues either GET PHYSICAL ELEMENT STATUS, REMOVE ELEMENT AND TRUNCATE, or RESTORE ELEMENT AND REBUILD command to manage storage elements of a drive. Removal or restoration of elements may take up to a day to complete. One of the **-d**, **-l**, or **-r** options must be specified. These options are mutually exclusive. Only SCSI drives are supported. Changing the storage elements of a storage drive may result in the loss of all data on that storage drive. The drive may need to reinitialize after **-d** or **-r** commands. The data on the drive is inaccessible until one of these commands complete. Once one of these commands start, the drive is format corrupt until the operation successfully completes. While format corrupt, no read or write I/O is possible to the drive. If the drive power cycles, it will remain format corrupt and the operation must be restarted. TEST UNIT READY or "camcontrol tur" can monitor an in-progress depop operation.

-c *capacity*
Specify the desired capacity point for the drive. Valid only for the **-d** flag.

-d Remove the physical element from service or set the capacity point specified by the **-e** or **-c** flags. The drive’s capacity may be reduced by this operation.

-e *element*
Specify the physical element to remove from service. Valid only for the **-d** flag.

-l Report the current status of the physical elements of a drive.

-r Restore all the eligible physical elements to service.

help Print out verbose usage information.

ENVIRONMENT

The SCSI_MODES variable allows the user to specify an alternate mode page format file.

The EDITOR variable determines which text editor **camcontrol** starts when editing mode pages.

FILES

/usr/share/misc/scsi_modes is the SCSI mode format database.
/dev/xpt0 is the transport layer device.
*/dev/pass** are the CAM application passthrough devices.

EXAMPLES

```
camcontrol eject -n cd -u 1 -v
```

Eject the CD from cd1, and print SCSI sense information if the command fails.

```
camcontrol tur da0
```

Send the SCSI test unit ready command to da0. The **camcontrol** utility will report whether the disk is ready, but will not display sense information if the command fails since the **-v** switch was not specified.

```
camcontrol tur da1 -E -C 4 -t 50 -Q head -v
```

Send a test unit ready command to da1. Enable kernel error recovery. Specify a retry count of 4, and a timeout of 50 seconds. Enable sense printing (with the **-v** flag) if the command fails. Since error recovery is turned on, the disk will be spun up if it is not currently spinning. The SCSI task attribute for the command will be set to Head of Queue. The **camcontrol** utility will report whether the disk is ready.

```
camcontrol cmd -n cd -u 1 -v -c "3C 00 00 00 00 00 00 00 0e 00" \
          -i 0xe "s1 i3 i1 i1 i1 i1 i1 i1 i1 i1 i1 i1"
```

Issue a READ BUFFER command (0x3C) to cd1. Display the buffer size of cd1, and display the first 10 bytes from the cache on cd1. Display SCSI sense information if the command fails.

```
camcontrol cmd -n cd -u 1 -v -c "3B 00 00 00 00 00 00 00 0e 00" \
          -o 14 "00 00 00 00 1 2 3 4 5 6 v v v v" 7 8 9 8
```

Issue a WRITE BUFFER (0x3B) command to cd1. Write out 10 bytes of data, not including the (reserved) 4 byte header. Print out sense information if the command fails. Be very careful with this command, improper use may cause data corruption.

```
camcontrol modepage da3 -m 1 -e -P 3
```

Edit mode page 1 (the Read-Write Error Recover page) for da3, and save the settings on the drive. Mode page 1 contains a disk drive’s auto read and write reallocation settings, among other things.

```
camcontrol rescan all
```

Rescan all SCSI buses in the system for devices that have been added, removed or changed.

```
camcontrol rescan 0
```

Rescan SCSI bus 0 for devices that have been added, removed or changed.

```
camcontrol rescan 0:1:0
```

Rescan SCSI bus 0, target 1, lun 0 to see if it has been added, removed, or changed.

```
camcontrol tags da5 -N 24
```

Set the number of concurrent transactions for da5 to 24.

```
camcontrol negotiate -n da -u 4 -T disable
```

Disable tagged queueing for da4.

```
camcontrol negotiate -n da -u 3 -R 20.000 -O 15 -a
```

Negotiate a sync rate of 20MHz and an offset of 15 with da3. Then send a Test Unit Ready command to make the settings take effect.

```
camcontrol smpcmd ses0 -v -r 4 "40 0 00 0" -R 1020 "s9 i1"
```

Send the SMP REPORT GENERAL command to ses0, and display the number of PHYs it contains. Display SMP errors if the command fails.

```
camcontrol security ada0
```

Report security support and settings for ada0

```
camcontrol security ada0 -U user -s MyPass
```

Enable security on device ada0 with the password MyPass

```
camcontrol security ada0 -U user -e MyPass
```

Secure erase ada0 which has had security enabled with user password MyPass

WARNING! WARNING! WARNING!

This will *ERASE ALL* data from the device, so backup your data before using!

This command can be used against an SSD drive to restoring it to factory default write performance.

```
camcontrol hpa ada0
```

Report HPA support and settings for ada0 (also reported via identify).

```
camcontrol hpa ada0 -s 10240
```

Enables HPA on ada0 setting the maximum reported sectors to 10240.

WARNING! WARNING! WARNING!

This will *PREVENT ACCESS* to all data on the device beyond this limit until HPA is disabled by setting HPA to native max sectors of the device, which can only be done after a power-on or hardware reset!

DO NOT use this on a device which has an active filesystem!

```
      camcontrol persist da0 -v -i read_keys
```

This will read any persistent reservation keys registered with da0, and display any errors encountered when sending the PERSISTENT RESERVE IN SCSI command.

```
      camcontrol persist da0 -v -o register -a -K 0x12345678
```

This will register the persistent reservation key 0x12345678 with da0, apply that registration to all ports on da0, and display any errors that occur when sending the PERSISTENT RESERVE OUT command.

```
      camcontrol persist da0 -v -o reserve -s lun -k 0x12345678 -T ex_ac
```

This will reserve da0 for the exlusive use of the initiator issuing the command. The scope of the reservation is the entire LUN. Any errors sending the PERSISTENT RESERVE OUT command will be displayed.

```
      camcontrol persist da0 -v -i read_full
```

This will display the full status of all reservations on da0 and print out status if there are any errors.

```
      camcontrol persist da0 -v -o release -k 0x12345678 -T ex_ac
```

This will release a reservation on da0 of the type ex_ac (Exclusive Access). The Reservation Key for this registration is 0x12345678. Any errors that occur will be displayed.

```
      camcontrol persist da0 -v -o register -K 0x12345678 -S \
              -I sas,0x1234567812345678 -I sas,0x8765432187654321
```

This will register the key 0x12345678 with da0, specifying that it applies to the SAS initiators with SAS addresses 0x1234567812345678 and 0x8765432187654321.

```
      camcontrol persist da0 -v -o register_move -k 0x87654321 \
              -K 0x12345678 -U -p -R 2 -I fcp,0x1234567812345678
```

This will move the registration from the current initiator, whose Registration Key is 0x87654321, to the Fibre Channel initiator with the Fiber Channel World Wide Node Name 0x1234567812345678. A new registration key, 0x12345678, will be registered for the initiator with the Fibre Channel World Wide Node Name 0x1234567812345678, and the current initiator will be unregistered from the target. The reservation will be moved to relative target port 2 on the target device. The registration will persist across power losses.

```
camcontrol attrib sa0 -v -i attr_values -p 1
```

This will read and decode the attribute values from partition 1 on the tape in tape drive sa0, and will display any SCSI errors that result.

```
camcontrol zone da0 -v -c rz -P summary
```

This will request the SMR zone list from disk da0, and print out a summary of the zone parameters, and display any SCSI or ATA errors that result.

```
camcontrol zone da0 -v -c rz -o reset
```

This will request the list of SMR zones that should have their write pointer reset from the disk da0, and display any SCSI or ATA errors that result.

```
camcontrol zone da0 -v -c rwp -l 0x2c80000
```

This will issue the Reset Write Pointer command to disk da0 for the zone that starts at LBA 0x2c80000 and display any SCSI or ATA errors that result.

```
camcontrol epc ada0 -c timer -T 60.1 -p Idle_a -e -s
```

Set the timer for the Idle_a power condition on drive *ada0* to 60.1 seconds, enable that particular power condition, and save the timer value and the enabled state of the power condition.

```
camcontrol epc da4 -c goto -p Standby_z -H
```

Tell drive *da4* to go to the Standby_z power state (which is the drive's lowest power state) and hold in that state until it is explicitly released by another **goto** command.

```
camcontrol epc da2 -c status -P
```

Report only the power state of drive *da2*. Some drives will power up in response to the commands sent by the *status* subcommand, and the **-P** option causes **camcontrol** to only send the ATA CHECK POWER MODE command, which should not trigger a change in the drive's power state.

```
camcontrol epc ada0 -c list
```

Display the ATA Power Conditions log (Log Address 0x08) for drive *ada0*.

```
camcontrol timestamp sa0 -s -f "%a, %d %b %Y %T %z" \
        -T "Wed, 26 Oct 2016 21:43:57 -0600"
```

Set the timestamp of drive *sa0* using a strptime(3) format string followed by a time string that was created using this format string.

SEE ALSO

cam(3), cam_cdbparse(3), cam(4), pass(4), xpt(4), diskinfo(8), trim(8), zonectl(8)

HISTORY

The **camcontrol** utility first appeared in FreeBSD 3.0.

The mode page editing code and arbitrary SCSI command code are based upon code in the old scsi(8) utility and scsi(3) library, written by Julian Elischer and Peter Dufault. The scsi(8) program first appeared in 386BSD-0.1.2.4, and first appeared in FreeBSD in FreeBSD 2.0.5.

AUTHORS

Kenneth Merry <*ken@FreeBSD.org*>

BUGS

The code that parses the generic command line arguments does not know that some of the subcommands take multiple arguments. So if, for instance, you tried something like this:

```
camcontrol cmd -n da -u 1 -c "00 00 00 00 00 v" 0x00 -v
```

The sense information from the test unit ready command would not get printed out, since the first getopt(3) call in **camcontrol** bails out when it sees the second argument to **-c** (0x00), above. Fixing this behavior would take some gross code, or changes to the getopt(3) interface. The best way to circumvent this problem is to always make sure to specify generic **camcontrol** arguments before any command-specific arguments.