allocateBufferOfLength:actualStart:actualLength:
getDMAAlignment:

Requesting I/O executeRequest:buffer:client:
maxTransfer

Reserving SCSI targets reserveTarget:lun:forOwner:
releaseTarget:lun:forOwner:

Resetting the SCSI bus resetSCSIBus

Getting the IOReturn equivalent of a sc_status_t value
returnFromScStatus:

(void *)allocateBufferOfLength:(unsigned)length
actualStart:(void **)actualStart
actualLength:(unsigned *)actualLength

Allocates and returns a pointer to some well-aligned memory. Well-aligned memory is necessary for calls to executeRequest:buffer:client:. You should use actualStart and actualLength when freeing the memory, as follows (italicized text delineated in angle brackets, that is << >>, is to be filled in with device-specific code):

Executes the specified request. Indirect devices invoke this method whenever they need the IOSCS
I/O.

Subclasses of IOSCSIController must implement this method. A typical implementation of this me
following:

·Using IOScheduleFunc() to schedule a timeout function to be called after scsiRequest->timeoutLe
without I/O completion

·Sending the command descriptor block (CDB) specified in scsiRequest to the controller

·When the I/O has completed, unscheduling the timeout function

This method should return scsiRequest->driverStatus, which should be set by the part of the driver
completion or timeout.

Indirect devices use this method as shown below (italicized text delineated in angle brackets, that i
in with device-specific code):

(void)getDMAAlignment:(IODMAAlignment *)alignment

Returns the DMA alignment requirements for the current architecture. IOSCSIController subclasse
method to specify any device-specific alignment requirements. See the description of allocateBuffe
actualStart:actualLength: for an example of using this method.

allocateBufferOfLength:actualStart:actualLength:

(unsigned)maxTransfer

Releases the specified target/lun pair. If owner hasn't reserved the pair, this method uses IOLog to

reserveTarget:lun:forOwner:

(int)reserveTarget:(unsigned char)target
lun:(unsigned char)lun
forOwner:owner

Reserves the specified target/lun pair, if it isn't already reserved. This method is invoked by a clien
SCSIDisk instance) to mark a particular target/lun as being in use by that client. Usually, this happ
however, the SCSIGeneric driver uses this method at other times.

This method returns a nonzero value if the target/lun pair is already reserved. Otherwise, it returns

releaseTarget:lun:forOwner:

(sc_status_t)resetSCSIBus

Resets the SCSI bus. Subclasses of IOSCSIController must implement this method so that it resets
sc_status_t enumerated type is defined and described in the header file bsd/dev/scsireg.h.

(IOReturn)returnFromScStatus:(sc_status_t)sc_status

Returns the IOReturn value corresponding to the specified sc_status_t value. The sc_status_t enum
and described in the header file bsd/dev/scsireg.h.