This document describes a summary of the *scu* commands implemented for the different device types and the actual SCSI command issued for each . There are many other *scu* commands and options described in the help file, which is accessed via "*scu help*". Each *scu* command is described in fuller detail in the help file, and is usually accompanied by an example of each command.

Several SCSI commands are listed with a *scu* command of NONE. These are commands we may wish to consider implementing if we want these tested.

I did not create a separate table for optical devices, since we don't usually qualify these, but they are very similar to direct access devices. However, *scu* does have an "*erase media*" command in support of optical devices.

Note: Most of the SCSI-2 commands are listed, as are those SCSI-3 commands now used by Tru64 UNIX/Clusters V5.0. Please let me know if there are additional SCSI commands which ought to be added.

I think this is a good document to send other groups who would like to know what test coverage is provided by *scu*. I'll (try) to keep this updated as new commands are implemented. There's a fair amount of *scu* work necessary for future DVD support.

As usual, your feedback and comments are welcome.

Enjoy,
Robin

Monday, January 1st, 2001

SCSI - All Devices	Opcode	Scu Command
COMPARE	39h	*NONE*
COPY	18h	*NONE*
COPY AND VERIFY	3Ah	*NONE*
INQUIRY	12h	*show inquiry [pages [inquiry-page] [supported]]*
LOG SELECT	4Ch	*zero log pages pcf { code hex-value | current | default | threshold }*
LOG SENSE	4Dh	*{ [code hex-code } [full | summary]* *show log pages { [log-page] } [pcf page-control]* *{ [supported] }*
MODE SELECT(6)	15h	*{ [code hex-code] }* *set pages { [mode-page] } [field-name] [pcf page-control]* **or** *{ [code hex-code] }* *change pages { [mode-page] } [pcf page-control*] Note: Use “*show layout*” to display field names for “*set pages*”.

MODE SENSE(6)
MODE SENSE(10)
	1Ah
	5Ah
		{ [code hex-code] }
		show pages { [mode-page] } [pcf page-control]
		{ [supported] }
		Note: The 10 byte Mode Sense is used if supported by the device.

PERSISTENT RESERVE IN PERSISTENT RESERVE OUT	5Eh 5Fh	*preserve { action } [aptpl] [key value] [type ptype] \* *[scope stype] [skey value]* Where persistent reservation 'action' is one of: *{ clear | register | release | reserve | preempt | preclear }* *show reservations*
READ BUFFER WRITE BUFFER	3Ch 3Bh	*test memory [test-parameters] [parameters]* *download filename [save] [parameters]* Note: “*test memory*” uses both Read/Write Buffer commands, where “*download*” only uses the Write Buffer command.
RECEIVE DIAGNOSTIC RESULTS **SEND DIAGNOSTIC**	1Ch 1Dh	*show diagnostic pages* *test self*
REPORT LUNS	A0h	*show luns*
REQUEST SENSE	03h	*show sense* Note: Autosense is supported by most operating systems, so the sense data is usually returned on any command failed with a Check Condition.
TEST UNIT READY	00h	*tur*

SCSI – Direct Access	Opcode	*Scu* Command
FORMAT UNIT	04h	*format [defects defect-list] [interleave value]* \ *[density density-type] [flags value]* \ *[dlist defect-format] [options ...]* \ *[pattern value] [working time-value]*
LOCK-UNLOCK CACHE	36h	*NONE*

MODE SELECT(6)
MODE SENSE(6)
	15h
	1Ah
		{ block-length value }
		set device { capacity value }
		{ density-code value }
		{ specific value }

PRE-FETCH	34h	*NONE*
PREVENT-ALLOW MEDIUM REMOVAL	1Eh	*allow | prevent*

READ(6)
READ(10)
	08h
	28h
		read media [test-parameters]
		Note: A 6-byte or 10-byte CDB is chosen automatically depending on the *lba* range and/or transfer size selected.

READ CAPACITY	25h	*show capacity*
READ DEFECT DATA (10 & 12)	37h	*show defects [[defect-list [defect-format]]]*
READ LONG WRITE LONG	3Eh 3Fh	*mbad lba n [{ hard | soft }] [flip bits]* \ *[length bytes] [nowrite]*
REASSIGN BLOCKS	07h	*reassign lba n*
RECEIVE DIAGNOSTIC RESULTS SEND DIAGNOSTIC	1Ch 1Dh	*translate address supplied-format to translate-format*
RELEASE	17h	*release device [rid reservation-identification]*
RESERVE	16h	*reserve device [rid id [rtype type] [lba n] [blocks n]]* Note: I have **not** encountered any drives supporting extents.
REZERO UNIT	01h	*rezero*
SEEK(10)	2Bh	*seek lba value*
SET LIMITS	33h	*NONE*
START STOP UNIT	1Bh	*start [immed] | stop [immed]*
SYNCHRONIZE CACHE	35h	*NONE*
VERIFY	2Fh	*verify media [test-parameters]*

WRITE(6)
WRITE(10)
	0Ah
	2Ah
		write media [test-parameters]
		Note: A 6-byte or 10-byte CDB is chosen automatically depending on the *lba* range and/or transfer size selected.

WRITE AND VERIFY	2Eh	*write media [test-parameters] verify*
WRITE SAME	41h	*write same [test-parameters]*

SCSI – Sequential Access	Opcode	*Scu* Command
ERASE	19h	*mt erase*
LOAD UNLOAD	1Bh	*mt load | mt unload* **or** *mt online | mt offline*
LOCATE	2Bh	*mt locate lba value*

MODE SELECT(6)
MODE SENSE(6)
	15h
	1Ah
		{ blocking { fixed [size] | variable } }
		set tape { buffered mode }
		{ density { code value | type } }

PREVENT ALLOW MEDIUM REMOVAL	1Eh	*allow | prevent*
READ	08h	*read media [test-parameters]*
READ BLOCK LIMITS	05h	*mt limits*
READ POSITION	34h	*mt rdpos [short | full] | mt tell*
READ REVERSE	0Fh	*NONE*
RECOVER BUFFERED DATA	14h	*NONE*
RELEASE UNIT	17h	*release device*
REPORT DENSITY	44h	*mt density*
RESERVE UNIT	16h	*release device [rid reservation-identification]*
REWIND	01h	*mt rewind*
SPACE	11h	*mt (bsf | bsr | fsf | fsr } [count]* *mt retention* *mt { seod | seorm }* Note: Spacing Setmarks is not currently implemented.
VERIFY	13h	*verify media [test-parameters]*
WRITE	0Ah	*write media [test-parameters]*
WRITE FILEMARKS	10h	*mt { eof | weof } [count]*

SCSI – Read-Only Direct Aceess	Opcode	*Scu* Command

MODE SELECT(6)
MODE SENSE(6)
	15h
	1Ah
		{ address { lba | msf } }
		set audio { density { code hex-code | density-code } }
		{ length block-length }
		{ volume [channel-{ 0 | 1 }] level n }
		show audio volume

PAUSE/RESUME	4Bh	*pause | resume*
PLAY AUDIO(10)	45h	*play audio lba n length n*
PLAY AUDIO(12)	A5h	*NONE*
PLAY AUDIO MSF	47h	*play msf starting minute-units second-units frame-units \* *ending minute-units second-units frame-units*
PLAY AUDIO TRACK/INDEX	48h	*play track n* ***or*** *play [[starting_track [n]] [ending_track [n]]]*
PLAY TRACK RELATIVE(10)	49h	*NONE*
PLAY TRACK RELATIVE(12)	A9h	*NONE*
PREVENT/ALLOW MEDIUM REMOVAL	1Eh	*allow | prevent*

READ(6)
READ(10)
	08h
	28h
		read media [test-parameters]
		Note: A 6-byte or 10-byte CDB is chosen automatically depending on the *lba* range and/and transfer size selected.

READ CD	BEh	*NONE*
READ CD-ROM CAPACITY	25h	*show capacity*
READ HEADER	44h	*show audio header lba n [address-format]*
READ LONG	3Eh	*NONE*

READ SUB-CHANNEL	42h	*{ channel }*
		{ catalog }
		show audio { isrc } [address-format]
		{ position }
		{ status }

READ TOC	43h	*show audio toc [{ entry n | header }] \* *[address-format] [report-format]*
RELEASE	17h	*release device*
RESERVE	16h	*reserve device [rid id [rtype type] [lba n] [blocks n]]*
REZERO UNIT	01h	*rezero*
SEEK(10)	2Bh	*seek lba n*
SET CD SPEED	BBh	*NONE*
START STOP UNIT	1Bh	*start [immed] | stop [immed]*
VERIFY(10)	2Fh	*verify media [test-parameters]*

SCSI – Medium Changers	Opcode	scu Command
EXCHANGE MEDIUM	A6h	*exchange medium source e-address destination1 e-address* \\ *destination2 e-address [medium-transport e-address]* \\ *[invert1] [invert2]* \\ *[drive value] [port value]* \\ *[slot value] [transport value]* \\ *[tags string] [atags string] [ptags string]*
MOVE MEDIUM	A5h	*move medium source e-address destination e-address* \\ *[medium-transport e-address] [invert]* \\ *[drive value] [port value]* \\ *[slot value] [transport value]* \\ *[tags string] [atags string] [ptags string]*
POSITION TO ELEMENT	2Bh	*position element destination e-address* \\ *[medium-transport e-address]* \\ *[drive value] [port value]* \\ *[slot value] [transport value]* \\ *[tags string] [atags string] [ptags string]*
PREVENT ALLOW MEDIUM REMOVAL	1Eh	*allow | prevent*
READ ELEMENT STATUS	B8h	*show elements [starting e-address] [ending e-address]* \\ *[number value] [type element-type]* \\ *[drive value] [port value]* \\ *[slot value] [transport value]* \\ *[{ full | summary }]*
RELEASE	17h	*release device* **TODO:** *release element* is **not** implemented (my oversight).
REQUEST VOLUME ELEMENT ADDRESS SEND VOLUME TAG	B5h B6h	*show elements [tags string] [atags string] [ptags string]* \\ *[notags] [voltags]* \\ *[{ full | summary }]*
RESERVE	16h	*reserve elements [starting e-address] [ending e-address]* \\ *[number value] [type element-type]* \\ *[drive value] [port value]* \\ *[slot value] [transport value]* \\ *[tags string] [atags string] [ptags string]* \\ *[rid id]*
REZERO UNIT	01h	*rezero*
SEND VOLUME TAG	B6h	*set element [action { assert | replace | undefine }]* \\ *[drive value] [port value]* \\ *[slot value] [transport value]* \\ *[tag string] [atag string] [ptag string]*