91-00718-01

SYSGEN INTELLIGENT DUAL FUNCTION DISK/TAPE CONTROLLER

SC2111EX PRODUCT SPECIFICATION

(8K)

LETTER	DESCRIPTION	DATE	CHNG	APPRVD
	RELEASE	3/1/82	NL	
A	SEL TP TK, TP DR TYPE, HEADER cmd's	3/15/82	NL	
B	MERGE TWO TP DR TYPES, SW SETTING	4/13/82	NL	
C	ADD TP FLG IN POSEOF, MOD 4A ERR	4/28/82	NL	
D	REMOVE W5 SHUNT, SHUNT EXPLANATION	5/06/82	NL	
E	UPGRADE TO EXTEND SERIES AND MISC	6/06/82	NL	
F	PARA. 12.5 TO ADD BOOT AREA DESC.	8/20/82	NL	
G	UPGRADE TO 8K AND ADD 9045B	1/06/83	NL	

DOCUMENT NO. 2111E-500-00

----- TABLE OF CONTENTS -----

1.0 SCOPE

This document describes the features, functionality and specification of SC2111E, the DISK/TAPE controller.

2.0 GENERAL DESCRIPTION

SC2111E is a high performance, microprocessor-based, single board controller capable of controlling a maximum of four 5 1/4 " hard disk drives and an ARCHIVE'S SIDEWINDER cartridge tape drive, either 9020B or 9045B.

The controller board is designed to be easily integrated into the system. It also significantly reduces the traditional interaction with the system CPU for controller operations and automatically handles the disk and tape media defects.

SC2111E is the Extended Series, which is equipped with define disk parameters command to allow the controller interfacing to all ST506 interface compatible disk drives.

2.1 PRODUCT FAMILY

SC2111E controller can interface to a variety of disk and tape drives.

PRODUCT	Up to 4 disk drives from the group	Archive's Sidewinder 9020B or 9045B
SC2111E	X	X

Device	Capacity		
Disk drive group:			
Seagate ST506	6.38 Mbytes	153 cyl.	4 tracks/cyl.
Seagate ST412	12.76 Mbytes	306 cyl.	4 tracks/cyl.
All drives of ST506			
Tape drive :			
9020B or Cipher Quarterback	20 Mbytes	450 Ft.	4 tracks
9045B	45 Mbytes	450 Ft.	9 tracks

3.12 TAPE READ-WHILE-WRITE ERROR CORRECTION
The controller performs a read-while-write on tape during backup operation and will rewrite any blocks in error.

3.13 TAPE READ RETRY CAPABILITY ON RESTORE
The controller automatically performs read retries on any data block in error during restore operation.

3.14 MAXIMUM UTILIZATION OF TAPE CAPACITY
It allows users to append more files on the remaining portion of the cartridge after power-down, removal of cartridge, or other interruptions causing misalignment of the read head in the tape drive. Therefore utilization of a whole cartridge is warranted.

3.15 RETRY CAPABILITY ON DISK OPERATIONS
Automatic retries are built in the controller in the event of seek, read or write errors encountered.

3.16 INTEGRAL TAPE DATA SEPARATOR
On board data separator circuitry for the streaming tape data.

3.17 STATISTICAL COUNTER FOR TAPE ERROR
A two-byte statistical counter for tape rewrite or re-read errors is maintained in the controller. These information allows the system to monitor any degradation in tape quality of the cartridge and the tape drive.

3.18 DIRECT TAPE READ ACCESS
The controller provides direct read access to the tape by the host without going through the disk drive.

3.19 NON-STOP STREAMING
The controller is designed to handle non-stop tape streaming operation at 90 IPS under nominal conditions.

3.20 DATA ERROR DETECTED AND CORRECTED
During disk data transfer, the controller with on board ECC circuitry can detect 11 bit burst error and correct 4 bit error burst.

3.21 BAD TRACK HANDLING
Bad tracks on disk may be assigned to have alternate tracks so that hard defects are masked off from the system. The controller will perform the automatic seek to alternate tracks on encountering any defective tracks.

3.22 LOGICAL ADDRESSING TO DISK DATA
The sectors on the disk are addressed logically using logical address bytes instead of cylinder, head and sector physical locations.

3.23 LOGICAL UNIT NUMBER
All disk drives are accessed by logical unit number independent of the physical port connections.

3.24 OVERLAPPED SEEK

In multiple drives environment, the host can issue seek to next drive without waiting for the current drive to complete its seek operation.

3.25 IMPLIED DISK POSITIONING DURING DATA TRANSFER COMMAND

For data transfer commands requiring new positioning on the drive, seek, head select and track verification are performed automatically by the controller.

3.26 AUTOMATIC HEAD AND CYLINDER SWITCHING

For data transfer commands involving multiple sectors across track or cylinder boundaries, the controller automatically switches to the next head or seeks to the next cylinder respectively. The new track is also verified to have the correct cylinder and head bytes in the ID field.

3.27 MULTIPLE SECTORS BUFFER

On board sector buffer is used to prevent data overrun during disk/host, disk/tape or host/tape data transfers.

3.28 FAST COPY COMMAND

Fast disk copy is implemented by concurrent multiple sector transfer between source and destination devices.

3.29 DEFINE DISK PARAMETERS

Definable disk parameter allows the controller to interface with all present and future interface compatible disk drives.

3.30 DISK CAPACITY INQUIRY

The host can send disk capacity inquiry command to find out the maximum logical address of the specified disk drive.

3.31 SEEK PROGRESS INQUIRY

The host can inquire about the completion of the previous seek command. This is specially useful in overlapped seek operation.

4.0 COMMAND DESCRIPTION

The host initiates the commands by passing a command descriptor block (CDB) to the controller. The first byte of the block contains the class and op code of the command. Depending on this byte, the controller will further interpret the following bytes as logical unit number, logical address, number of blocks to be transferred, interleave constant, control flag and etc.

The following is the command summary:

CLASS	OPCODE		COMMAND
0	00		Test drive ready
0	01		Recalibrate
0	02		Request syndrome
0	03		Request sense
0	04		Format the drive
0	05		Check track format
0	06		Format a track
0	07		Reserved
0	08		Read blocks
0	09		Reserved
0	0A		Write blocks
0	0B		Seek
0	0C		Select tape track
0	0D		Reserved
0	0E		Format bad track with alternate address
0	0F		Erase the cartridge
0	10		Request statistic
0	11		Position to end of file
0	12		Disk capacity inquiry
0	13		Controller type inquiry
0	14		Seek progress inquiry
0	15		Maximum tape track inquiry
1	00	$20	Copy from disk to disk
1	01	$21	Restore
1	02	$22	Backup
1	03	$23	Read file from tape to host
1	04	$24	Read tape file header
6	00	$C0	Reserved
6	01	$C1	Reserved
6	02	$C2	Define tape drive type
6	03	$C3	Define disk parameters

4.1 COMMAND DESCRIPTOR BLOCK DEFINITION:

The host passes commands to the controller through command descriptor block (CDB), which consists of either six bytes or twelve bytes depending on the command class.
The first byte of the block always denotes a class code and an operation code.

Because of variation of formats in commands, refer to Section 4.2 for each command's CDB format.

4.1.1 LOGICAL ADDRESS COMPUTATION

The physical address to logical address conversion is as following:

LOGICAL ADR = (CYL * HDCYL + HEAD) * SECTRK + SECT

where: CYL = cylinder address
HEAD = head address
SECT = sector address

HDCYL = number of heads per cylinder
SECTRK = number of sectors per track

Bit 0 of logical adr byte 0 = the least significant bit
Bit 4 of logical adr byte 2 = the most significant bit

4.1.2 BLOCK SIZE

Data transferred by the controller are in blocks.

1 block = 256 bytes
1 sector = 1 block
1 disk track = 32 sectors

4.2 COMMAND DESCRIPTOR BLOCK FORMAT

Class 0

opcode 00 Test drive ready - Selects drive and tests if it is ready.

CDB

	7	6	5	4	3	2	1	0
byte 0	0	0	0	0	0	0	0	0
1	⟨	LUN	⟩	0	0	0	0	0
2	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0

opcode 01 Recalibrate - Positions the head to track 0 and resets error status on the drive.

CDB

	7	6	5	4	3	2	1	0
byte 0	0	0	0	0	0	0	0	1
1	⟨	LUN	⟩	0	0	0	0	0
2	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0

opcode 02 Request syndrome sense bytes - The controller sends the bit offset position and syndrome response upon system request for data field error correction.

CDB

	7	6	5	4	3	2	1	0
byte 0	0	0	0	0	0	0	1	0
1	‹ LUN ›			0	0	0	0	0
2	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0

2 Syndrome Sense bytes definition

	7	6	5	4	3	2	1	0
byte 0	M.S. bit offset							
1	! L.S. bit offset !			!	!	syndrome bits		

Note: If the 4 Syndrome bits returned are all zeros, it means the error occurred in ECC bytes of the data field and no correction is needed.

opcode 03 Request sense bytes - This command must be issued immediately after error detected (The host detects error by checking the composite error bit of completion status byte).
The 4 bytes response further explains the type of error.

CDB

	7	6	5	4	3	2	1	0
byte 0	0	0	0	0	0	0	1	1
1	< LUN >			0	0	0	0	0
2	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0

4 Sense bytes definition after disk error - logical adr 2,1,0 only valid when bit 7 of byte 0 is set.

	7	6	5	4	3	2	1	0
byte 0	block addr valid		error type		error code			
1	< LUN >			logical adr 2				
2	logical adr 1							
3	logical adr 0							

4 Sense bytes definition after tape related error
(i.e. tape full and missing block errors)

	7	6	5	4	3	2	1	0
byte 0	0	!	error type	!	error code			
1	(TP LUN)			*	Byte 2			
2	*	Byte 1						
3	*	Byte 0						

* = this 21-bit counter contains the number of blocks that have been sucessfully saved or restored when tape full or missing block error was detected

opcode 04 Format drive - Formats the whole disk. Refer to Track Format section in this document. To obtain the best performance of tape data transfer, the user should use interleave constant of 0B (11 in decimal).

CDB

	7	6	5	4	3	2	1	0
byte 0	0	0	0	0	0	1	0	0
1	(LUN)			0	0	0	0	0
2	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0
4	interleave constant							
5	0	0	0	0	0	0	0	0

opcode 05 Check track format - Checks the format on the track for correct ID field and interleave constant.

CDB

	7	6	5	4	3	2	1	0
byte 0	0	0	0	0	0	1	0	1
1	(LUN)				logical adr 2			
2	logical adr 1							
3	logical adr 0							
4	interleave constant							
5	0	0	0	0	0	0	0	0

opcode 06 Format a track - Formats a single track.

CDB

	7	6	5	4	3	2	1	0
byte 0	0	0	0	0	0	1	1	0
1	(LUN)				logical adr 2			
2	logical adr 1							
3	logical adr 0							
4	interleave constant							
5	0	0	0	0	0	0	0	0

opcode 07 Reserved

opcode 08 Read blocks - Reads the specified number of blocks starting from the given logical address.

CDB

	7	6	5	4	3	2	1	0
byte 0	0	0	0	0	1	0	0	0
1	(LUN)			logical adr 2				
2	logical adr 1							
3	logical adr 0							
4	number of blocks							
5	*	**	0	0	0	0	0	0

* = Disable retry flag
** = Disable data correction flag

opcode 09 Reserved

opcode 0A Write blocks - Writes the specified number of blocks starting from the given logical address.

CDB

	7	6	5	4	3	2	1	0
byte 0	0	0	0	0	1	0	1	0
1	(LUN)			logical adr 2				
2	logical adr 1							
3	logical adr 0							
4	number of blocks							
5	*	0	0	0	0	0	0	0

* = Disable retry flag

Note: 00 in byte 4 (number of blocks byte in class 0) is equivalent to FF + 1 = 256

opcode 0B Seek - Seeks to the specified cylinder and immediately returns completion status before seek complete. This allows overlap seeks in a multiple drives environment.

CDB

	7	6	5	4	3	2	1	0
byte 0	0	0	0	0	1	0	1	1
1	(LUN)				logical adr 2			
2	logical adr 1							
3	logical adr 0							
4	0	0	0	0	0	0	0	0
5	*	0	0	0	0	0	0	0

* = Disable retry flag

opcode 0C Select tape track - Returns the tape to Beginning of Tape position and selects the starting track #. For split tape drive mode, the starting track # is either 0, 2, 4, or 8. For the whole tape drive mode, the starting track # should be always 0.

CDB	7	6	5	4	3	2	1	0
byte 0	0	0	0	0	1	1	0	0
1	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0
4	(TP LUN)		0	0	0	starting track #		
5	0	0	0	0	0	0	0	0

Note: Since the controller can only interface to one Archive tape drive, the two-bit TP LUN should always be 00 in all CDB's.

Note = Select Track positions to BOT!

opcode 0D Reserved

opcode 0E Format bad track with alternate address -
This command sets the M.S. bit of head byte in I.D. field to indicate it is a bad track and writes the given alternate address into the first two bytes of the data field.
This operation is done on all sectors of the bad track and it is assumed at least one sector is good enough to hold the alternate address. Refer to bad track format section in this document.

** Also note this is the only command where the address given in the CDB is in terms of physical address instead of logical. **

CDB

	7	6	5	4	3	2	1	0
byte 0	0	0	0	0	1	1	1	0
1	⟨ LUN ⟩			0	0	0	0	0
2	bad track head adr				!	M.S. cyl. adr		
3	L.S. cylinder adr							
4	alt track head adr				!	M.S. cyl. adr		
5	L.S. cylinder adr							

opcode 0F Erase tape - The controller erases the whole tape and positions the tape to Beginning Of Tape (BOT) position when finished.

CDB	7	6	5	4	3	2	1	0
byte 0	0	0	0	0	1	1	1	1
1	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0
4	(TP LUN)	0	0	0	0	0	0	
5	0	0	0	0	0	0	0	0

opcode 10 Request statistic - This command is issued to inquire the number of rewrite counts after backup or number of reread counts after restore. The controller returns 4 byte response upon this request.

CDB	7	6	5	4	3	2	1	0
byte 0	0	0	0	1	0	0	0	0
1	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0
4	(TP LUN)	0	0	0	0	0	0	
5	0	0	0	0	0	0	0	0

4 byte statistic response definition

	7	6	5	4	3	2	1	0
byte 0	0	0	0	0	0	0	0	0
1	(TP LUN)	0	0	0	0	0		
2	rewrite or reread count byte 1							
3	rewrite or reread count byte 0							

opcode 11 Position to End of File - The controller positions the tape to the end of the file.

CDB	7	6	5	4	3	2	1	0
byte 0	0	0	0	1	0	0	0	1
1	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0
4	(TP LUN)			file I.D. number				
5	*	$	0	0	0	0	0	0

* = Tape retension flag
$ = Suppress BOT flag; when set, avoid BOT positioning

opcode 12 Disk Capacity inquiry - This command allows the host to inquire about the maximum logical address existing on the specified disk drive. The controller will return 4-byte response to the host.

CDB

	7	6	5	4	3	2	1	0
byte 0	0	0	0	1	0	0	1	0
1	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0

4 byte response

	7	6	5	4	3	2	1	0
byte 0	0	0	0	0	0	0	0	0
1	(LUN)				Max logical adr 2			
2	Max logical adr 1							
3	Max logical adr 0							

Note: Maximum logical address bytes reflects the maximum capacity of the drive as dictated by the max cylinder adr and max head adr. Therefore, the actual capacity would be smaller if there are some bad tracks existing in the drive.

opcode 13 Controller Type Inquiry - This command allows the host to inquire about what Class and Type of this controller and also what revision of the firmware in it.
The controller will return 4-byte response to the host.

CDB

	7	6	5	4	3	2	1	0
byte 0	0	0	0	1	0	0	1	1
1	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0

4 byte response

	7	6	5	4	3	2	1	0
byte 0	# (Other reserved bits for classes)							
1	Controller type byte							
2	Reserved							
3	Controller's firmware revision							

\# = 1 for SC2111E
controller type byte = 21 hex for SC2111E

opcode 14 Seek progress inquiry - This command allows the host to inquire about the seek progress on the disk drive. This is useful to determine if the last overlapped seek has been completed.
The controller will return 4-byte response to the host.

CDB

	7	6	5	4	3	2	1	0
byte 0	0	0	0	1	0	1	0	0
1	⟨ LUN ⟩			0	0	0	0	0
2	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0

4 byte response

	7	6	5	4	3	2	1	0
byte 0	*	⟨			Reserve			⟩
1	⟨ LUN ⟩				Reserve			
2	Reserve							
3	Reserve							

* = if set, the seek still in progress
= if reset, the seek is completed

opcode 15 Maximum Tape Track Inquiry - This command echoes back Maximum Track Address and Tape Drive Type to the host, as previously defined by the user in ' Define Tape Drive Type ' command.

The controller will return 4-byte response to the host.

CDB

	7	6	5	4	3	2	1	0
byte 0	0	0	0	1	0	1	0	1
1	‹ LUN ›			0	0	0	0	0
2	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0

4 byte response

	7	6	5	4	3	2	1	0
byte 0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0
2	Maximum Tape Track Address							
3	Tape Drive Type							

Class 1 Commands in this class are 12-byte long

opcode 00 Disk Copy - This command copies the data blocks from the specified source disk starting location to the specified destination disk starting location. The number of data blocks to be transferred is specified in bytes 7-9.

CDB	7	6	5	4	3	2	1	0
byte 0	0	0	1	0	0	0	0	0
1	(src. LUN)			logical adr 2				
2	logical adr 1							
3	logical adr 0							
4	(dst. LUN)			logical adr 2				
5	logical adr 1							
6	logical adr 0							
7	number of blocks byte 2							
8	number of blocks byte 1							
9	number of blocks byte 0							
10	0	0	0	0	0	0	0	0
11	*	**	0	0	0	0	0	0

* = Disable retry flag
** = Disable data correction flag

opcode 01 Restore - This command restores the tape data specified in tape file ID # into the Disk from starting disk logical adr. The number of blocks to be restored is specified in bytes 7-9.
Bytes 5 and 6 are the block offset. This allows the controller to skip a certain number of blocks before reading the tape. In other words, the user can randomly access to the file. But the user should be aware that the maximum offset is 65,536 blocks where the maximum number of blocks in the file could be greater than this value.

Refer to Section 11.0 for restore operational description.

CDB

	7	6	5	4	3	2	1	0
byte 0	0	0	1	0	0	0	0	1
1	(disk LUN)			logical adr 2				
2	logical adr 1							
3	logical adr 0							
4	(TP LUN)	file I.D. number						
5	tape read block offset byte 1							
6	tape read block offset byte 0							
7	number of blocks byte 2							
8	number of blocks byte 1							
9	number of blocks byte 0							
10	*	$	%	0	0	0	0	0
11	#	0	0	0	0	0	0	0

* = Tape retension flag
$ = Suppress BOT flag; when set, avoid BOT positioning
% = Read next N block flg; if this is set, the controller WILL IGNORE File I.D. and block offset, assuming to read from the position of the current file.
# = Disable retry flag

opcode 02 Backup - This command saves the disk data from the starting disk logical adr into the tape. These data are written onto the tape as a tape file and the file is also given the file ID number specified in the CDB.
The number of blocks to be saved is specified in bytes 7-9.

Refer to Section 11.0 for the backup operational description.

CDB

	7	6	5	4	3	2	1	0
byte 0	0	0	1	0	0	0	1	0
1	(disk LUN)			logical adr 2				
2	logical adr 1							
3	logical adr 0							
4	(TP LUN)			file I.D. number				
5	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0	0
7	number of blocks byte 2							
8	number of blocks byte 1							
9	number of blocks byte 0							
10	0	0	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0

opcode 03 Read file from tape to host - This command is similar to restore command except the the data read from the tape is directly sent to the host instead of to the disk.

**** WARNING **** If the host transfer rate is slower than 11 msec per block, the user MUST NOT specify more than 14 blocks at a time.
Refer to section 11.0 for the operational description.

CDB

	7	6	5	4	3	2	1	0
byte 0	0	0	1	0	0	0	1	1
1	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0
4	(TP LUN) file I.D. number							
5	tape read block offset byte 1							
6	tape read block offset byte 0							
7	number of blocks byte 2							
8	number of blocks byte 1							
9	number of blocks byte 0							
10	*	$	%	0	0	0	0	0
11	0	0	0	0	0	0	0	0

* = Tape retension flag
$ = Suppress BOT flag; when set, avoid BOT positioning
% = Read next N block flg; if this is set, the controller WILL IGNORE File I.D. and block offset, assuming to read from the position of the current file.

Note: 000000 in bytes 7,8,9 (number of blocks bytes in Class 1) is ILLEGAL. These three bytes range from 000001 to FFFFFF (16777215).

opcode 04 Read tape file header - This command is to read the tape I.D. block and the first few data blocks of the tape files and transfer them to the specified disk address.
By using this command, the user can quickly retrieve the header information on the tape.

CDB

	7	6	5	4	3	2	1	0
byte 0	0	0	1	0	0	1	0	0
1	⟨disk LUN⟩				logical adr 2			
2	logical adr 1							
3	logical adr 0							
4	⟨TP LUN⟩			number of files				
5	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0
9	0	0	0	0	0	number of blocks		
10	*	$	0	0	0	0	0	0
11	#	0	0	0	0	0	0	0

* = Tape retension flag
$ = Suppress BOT flag; when set, avoid BOT
# = Disable retry flag

number of files = number of file headers to be read

number of blocks = the size of each file header. It ranges from 1-5, inclusive of File I.D. block.

Class 6

opcode 00 Reserved

opcode 01 Reserved

opcode 02 Define tape drive type - This command is to define the model of the tape drive and whether operated as a Whole Drive mode or as a 'Split Drive' mode.

Split Drive is to divide the tape drive into portions as if there were more than one tape drive available to the user.
With the Track Selection command, the user can then select that particular portion to speed up the access time to the tape data and consequently also expand number of files per cartridge beyond 256.

CDB

	7	6	5	4	3	2	1	0
byte 0	1	1	0	0	0	0	1	0
1	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0
3	Maximum Tape Track Address							
4	0	0	0	0	0	(TP DR TYPE)		
5	0	0	0	0	0	0	0	0

The Drive type 0 = split into portions of 2 tracks each
1 = split into portions of 4 tracks each
2 = split into portions of 8 tracks each
3 = split into portions of 12 tracks each
(is equivalent to all 9 tracks in 9045B)

The Maximum Tape Track Address is actually the Tape Drive Model in the configuration.

For example, the user has 9045B in his system and wants to split it into 4,4,1 tracks. He should issue this command with the following :

```
Maximum Tape Track Address  =  08  ( 9 track drive )
Tape Drive Type             =  01  ( 4 tracks each )
```

Should the user never issue this command after power-up, the controller WILL ASSUME maximum track address 03 (9020B drive) and drive type 01 (whole 4 tracks). And if the user defines the maximum track address 00, the controller WILL ASSUME maximum track address 03 also, and this is to be software compatible with the existing products.

When the user backups disk data onto the tape, the tape drive type information is also recorded in the 6th byte of ID block of the tape file. By using 'read tape file header' command, he can retrieve this byte and determine whether it was saved in the whole drive mode or split drive mode at the time of backup. (See Section 11.1 'read tape file header' explanation)

opcode 03 Define disk parameters - This command is an added feature in Extended Series 2000 controller. It allows the controller to interface to any ST506 interface compatible disk drives.
The user MUST specify disk parameters of all LUN's in the system at the power-up. The controller will not response to any other commands without those parameters.

CDB

	7	6	5	4	3	2	1	0
byte 0	1	1	0	0	0	0	1	1
1	(LUN)			Max cyl adr byte 1				
2	Max cyl adr byte 0							
3	Max head adr per cyl							
4	Seek step rate interval							
5	0	0	*	0	0	0	0	0

* = Disable reduced write current/ precomp flag

Max cyl address - For example, there are 306 cylinders in ST412 but the max cyl address is 305. Therefore hex entries by the user should be 0131H.

Max head adr per cyl - There are 4 heads per cyl in ST412 but the max head address is 03.

Seek step rate interval - This entry is used to specify the time interval between seek pulses. Every count in this byte is equal to 15 microseconds.
For example, ST412 is specified 30 microseconds between pulses in buffered seek mode and 02 should be entered.
For ST506, only slow seek mode of 3 milliseconds is available to the user. The entry in this byte should then be C8H (or 200 in decimal).

Disable reduced write current / precomp flag -
For ST412, reduced write current is not available, but precomp is required as specified, therefore this flag should be '0' to enable precomp function.
For ST506, this flag should then also be '0' for the reason of reduced write current requirement.

5.0 CONNECTOR DEFINITION

5.1 HOST INTERFACE CONNECTION

PIN#	NAME	DRIVEN BY	DESCRIPTION
02	DB0-	HT/CL	Bidirectional data bus 0
04	DB1-	HT/CL	Bidirectional data bus 1
06	DB2-	HT/CL	Bidirectional data bus 2
08	DB3-	HT/CL	Bidirectional data bus 3
10	DB4-	HT/CL	Bidirectional data bus 4
12	DB5-	HT/CL	Bidirectional data bus 5
14	DB6-	HT/CL	Bidirectional data bus 6
16	DB7-	HT/CL	Bidirectional data bus 7
18	PARITY-	HT/CL	Bidirectional parity bit
20	N/A		Not used
22	N/A		Not used
24	N/A		Not used
26	N/A		Not used
28	N/A		Not used
30	N/A		Not used
32	N/A		Not used
34	N/A		Not used
36	BSY-	CL	Busy
38	ACK-	HT	Acknowledge
40	RST-	HT	Reset
42	MSG-	CL	Message
44	SEL-	HT	Select controller
46	C/D-	CL	Command/Data mode select
48	REQ-	CL	Request
50	I/O-	CL	Input/Output mode select

Note: All odd pins are connected to ground.
HT = Host
CL = Controller
- = negative true signals

Signals received at the host side should be terminated with 220 ohms to +5V and 330 ohms to Gnd.

5.2 TAPE INTERFACE CONNECTION - Archive Sidewinder

PIN#	NAME	DRIVEN	DESCRIPTION
02	GO-	CL	Go control for capstan servo
04	REV-	CL	Direction control for capstan servo
06	TR3-	CL	Track select bit 3
08	TR2-	CL	Track select bit 2
10	TR1-	CL	Track select bit 1
12	TR0-	CL	Track select bit 0
14	RES	CL	Reserved for cartridge lock control
16	DS3-	CL	Drive 3 select control
18	DS2-	CL	Drive 2 select control
20	DS1-	CL	Drive 1 select control
22	DS0-	CL	Drive 0 select control
24	RDL-	TP	Read level output - a digitized derivative of the analog read signal
26	RDP-	TP	Read pulse output - a pulse per flux transition
28	UTH-	TP	Upper tape position code
30	LTH-	TP	Lower tape position code
32	SLD-	TP	Selected response from selected drive
34	CIN-	TP	Cartridge in place
36	USF-	TP	Unsafe - cartridge safe plug in "unsafe" position (i.e. writing is enabled)
38	TCH-	TP	Capstan tachometer pulse - each pulse equals 112 +/- 3% mils of tape movement
40	WDA-	CL	Write data signal
42	WDA+	CL	Inverse write data signal
44	N/A		Not used
46	HSD-	CL	High speed select control for model 3020B
48	WEN-	CL	Write enable control
50	EEN-	CL	Erase enable control

Note: All odd pins are signal returns, are connected to signal GND at the drive and should be connected to signal GND at the Host.

5.3 DISK DRIVE INTERFACE - CABLE A CONNECTION

PIN#	SIGNAL NAME	DRIVEN BY
01	GND	
02	-REDUCED WRITE CURRENT	CL
03	GND	
04	RESERVED (-HEAD SELECT 2)	CL
05	GND	
06	-WRITE GATE	CL
07	GND	
08	-SEEK COMPLETE	DK
09	GND	
10	-TRACK 00	DK
11	GND	
12	-WRITE FAULT	DK
13	GND	
14	-HEAD SELECT 0	CL
15	GND	
16	-RESERVED	
17	GND	
18	-HEAD SELECT 1	CL
19	GND	
20	-INDEX	DK
21	GND	
22	-READY	DK
23	GND	
24	-STEP	CL
25	GND	
26	-DRIVE SELECT 1	CL
27	GND	
28	-DRIVE SELECT 2	CL
29	GND	
30	-DRIVE SELECT 3	CL
31	GND	
32	-DRIVE SELECT 4	CL
33	GND	
34	-DIRECTION IN	CL

DISK DRIVE INTERFACE - CABLE B CONNECTION

01	-DRIVE SELECTED	DK
02	GND	
03	RESERVED	
04	GND	
05	RESERVED	
06	GND	
07	RESERVED	
08	GND	
09	RESERVED	
10	RESERVED	
11	GND	
12	GND	
13	+MFM WRITE DATA	CL
14	-MFM WRITE DATA	CL
15	GND	
16	GND	
17	+MFM READ DATA	DK
18	-MFM READ DATA	DK
19	GND	
20	GND	

6.1 MECHANICAL SPECIFICATION

Width	8.25	inches
Length	15.0	inches

6.2 ELECTRICAL SPECIFICATION

Voltage requirement	+5VDC +/- 5%
Current requirement	+5VDC @ 4.5 Amps (nominal)
	+5VDC @ 6.0 Amps (maximum)

6.3 ENVIRONMENTAL REQUIREMENTS

	Operating	Storage
Temp. degree F/C	32/0 to 131/55	-40/-10 to 167/75
Relative Humidity @ 40 degrees F	10% to 95%	10% to 95%
Altitude from sea level	0 to 10,000 ft	0 to 15,000 ft

7.0 HOST BUS INTERFACE

7.1 Signal Definition

7.1.1 Bidirectional signals

DB0 - DB7 — Data Bus betweeen the controller and the host.

ODD PARITY — Asserted when number of asserted bits on DB0-7 is even; deasserted when number of asserted bits on DB0-7 is odd.

7.1.2. Unidirectional signals driven by controller

I/O — Input/Output. When asserted the data on the bus is input to the host; when deasserted the data on the bus is output by the host.

C/D — Command/Data. When asserted the data transmitted across the bus will be the command bytes; when deasserted the data will be the data bytes.

BUSY — When deasserted, it indicates the controller is ready to be selected by the host and it is asserted after being selected.

MSG — Message. When asserted, it indicates that the command is completed.

REQ — Request. After all control signals been set up, REQ in conjunction with ACK are used for data and command transfer handshake.

I/O	C/D	MSG	Control mode
d	a	d	Get command from host
d	d	d	Get data from host
a	d	d	Send data to host
a	a	d	Send completion status byte to host
a	a	a	Send command done message byte to host

a = asserted
d = deasserted

7.1.3 Unidirectional Signals driven by Host

ACK	Acknowledge. This signal is asserted as a handshake response to controller's REQ.
SEL	Select. The host uses this signal to gain controller's attention at the beginning of command processing.
RST	Reset. Asserted-then-deasserted on this bit reinitializes the controller. Minimum requirement of Reset pulse (duration of this assertion) is 20 microseconds.

7.2 Operational Description

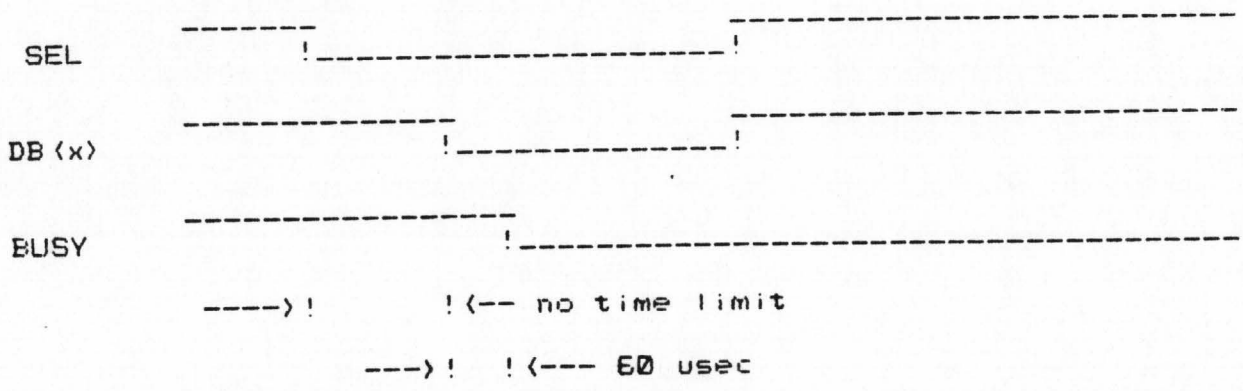
The following outlines the sequence of steps to issue a command to the controller;

1) Host polls on BUSY signal until deasserted to show the controller is ready to accept command.

2) Host asserts SEL and DB(x) (controller address bit) to gain the controller's attention and waits for the BUSY asserted by the controller indicating it has recognized the selection.

3) Host deasserts SEL and DB(x) . The controller from now has the control of the bus and is ready to accept Command from the host.

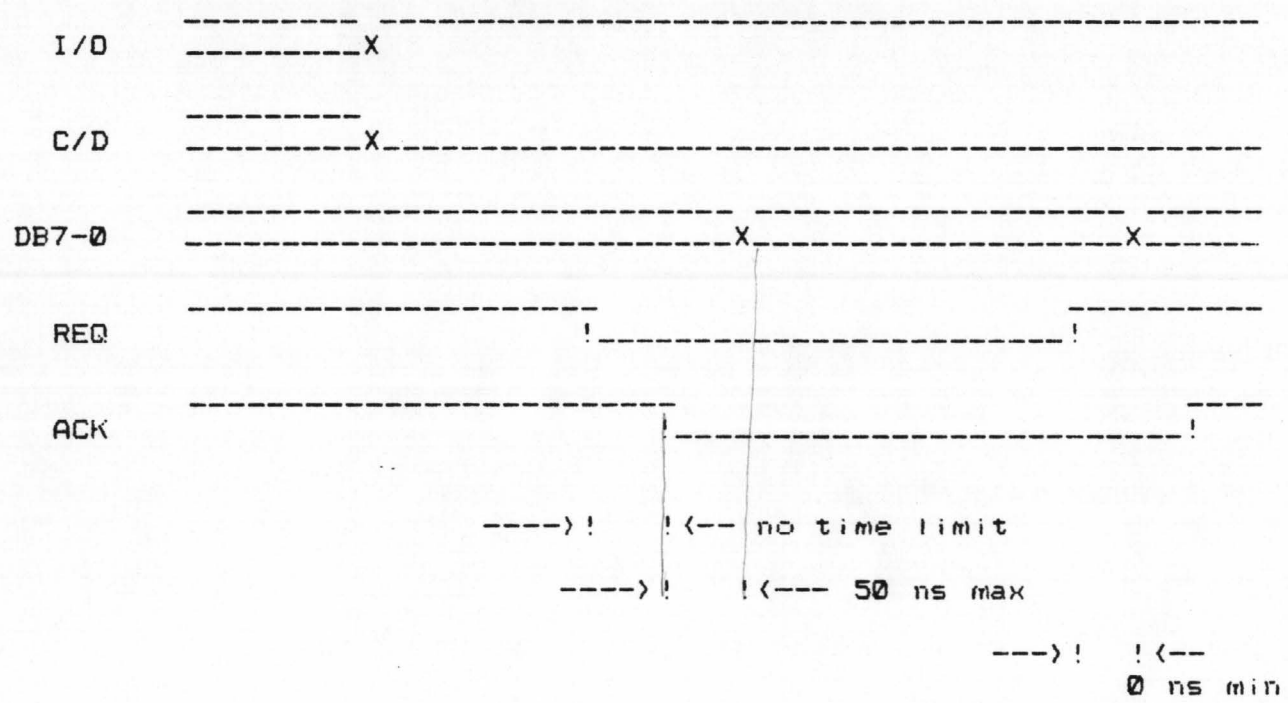
SELECTION TIMING ILLUSTRATION



DB(x) = controller address bit; DB0, 1, 2, 3, 4, 5, 6, or 7. Depending on the controller address ID from the dip switch, the corresponding controller will response to the Selection.

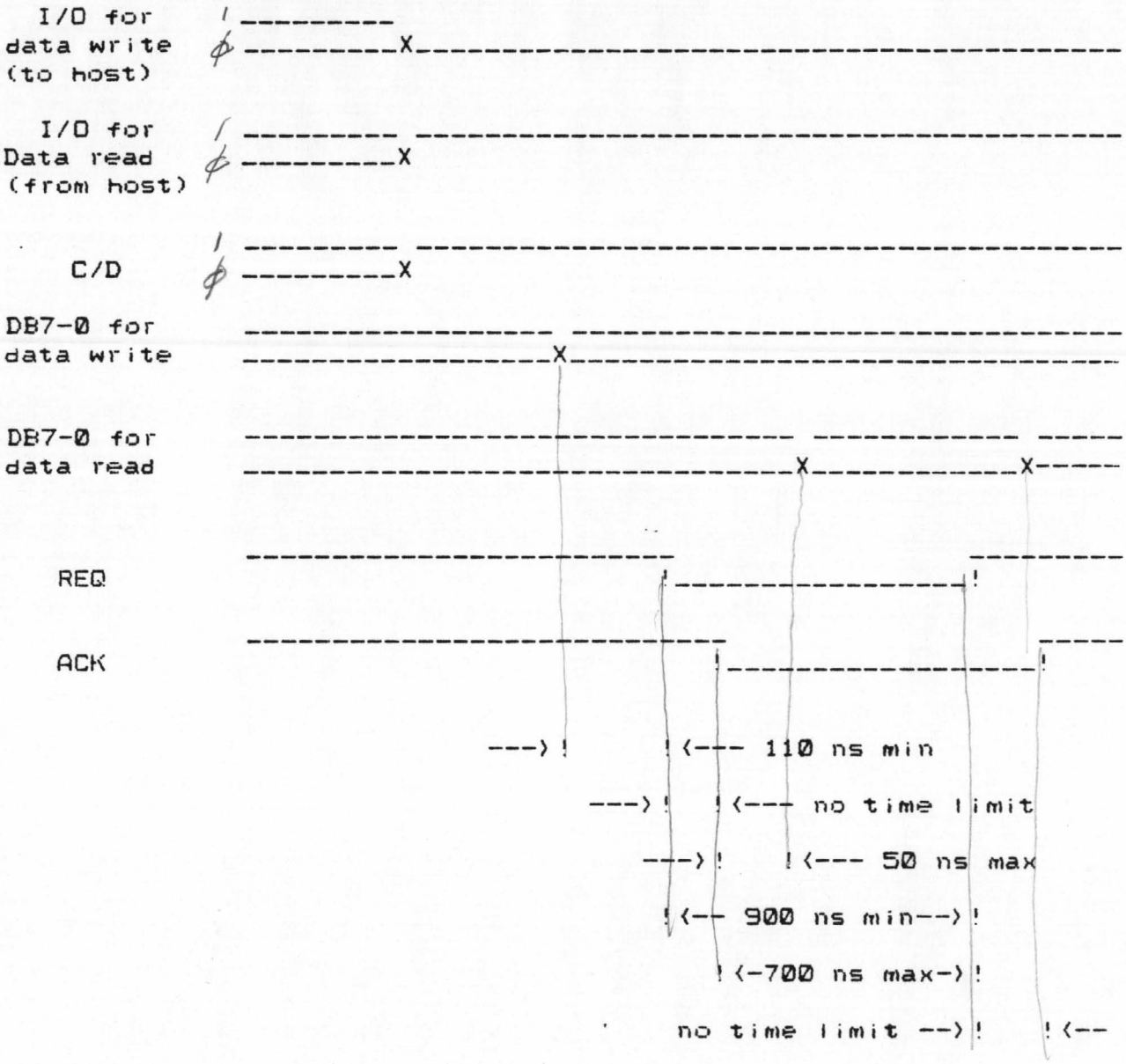
4) The controller then asserts C/D and deasserts I/O to indicates commands bytes output from the host the controller. Six command bytes are requested and transferred over the host bus with REQ/ACK handshake protocol. The controller determines from the first command byte what class it belongs. If it is class 1, additional six command bytes will be requested and transferred.

COMMAND TRANSFER TIMING ILLUSTRATION



5) For commands involving Data transfer, the controller deasserts C/D to indicate data mode transfer on the bus. I/O is to asserted for READ command and deasserted for WRITE command. REQ/ACK form the same handshake as in command transfer case.

DATA TRANSFER TIMING ILLUSTRATION



6) Upon completion of any command, the controller puts a completion status byte on the data bus, with C/D and I/O asserted. REQ/ACK handshake is used to transfer this completion status byte to the host.

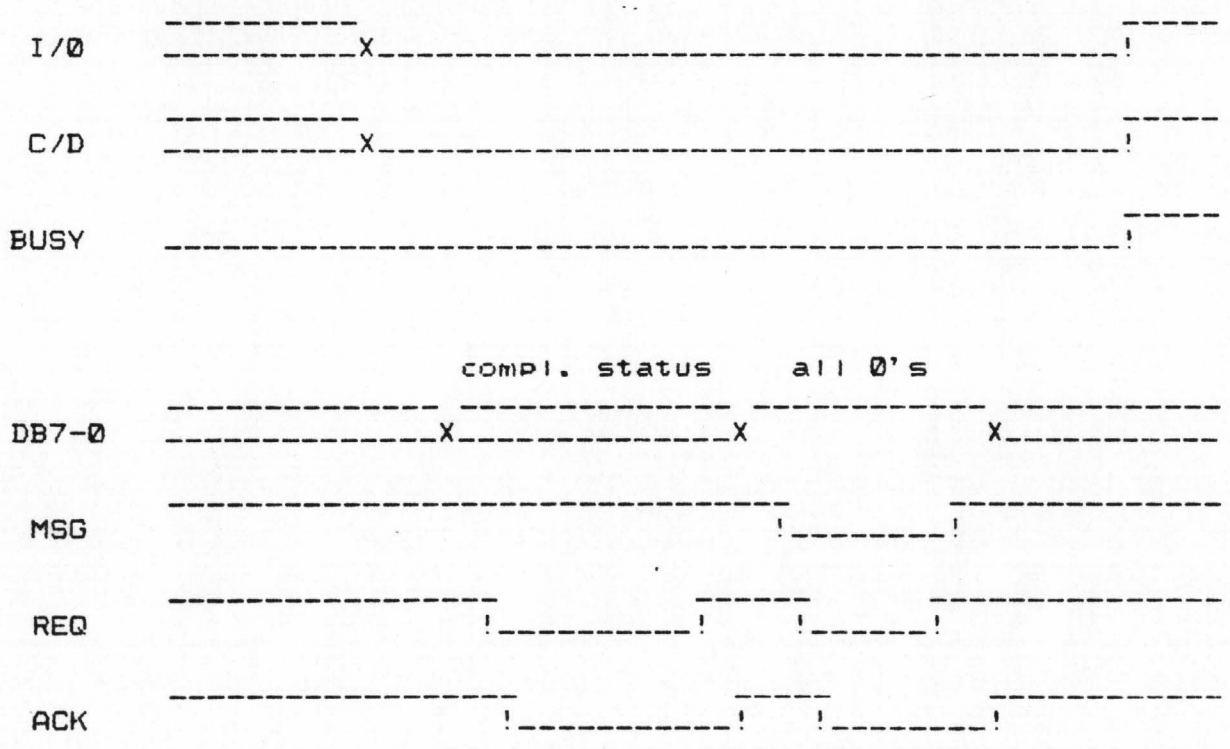
Completion status byte definition

7	6	5	4	3	2	1	0
(LUN)			(SPARE)	$	*	**	

* = composite error bit
** = host bus parity error bit
$ = tape related error flag

7) Then the controller places all zero's on the data bus with MSG asserted. The same REQ/ACK handshake to transfer this Message byte to the host.
Such message byte returned by the controller can be used to generate an interrupt to system to signify the completion of the command.

COMPLETION STATUS AND MESSAGE BYTES TRANSFER TIMING ILLUSTRATION



timing requirements are same as I/O data write case above

8) After sending Message to the host, the controller deasserts REQ, MSG, I/O, C/D , BUSY and returns to the loop where can be again selected for the next command.

8.0 DATA FORMAT

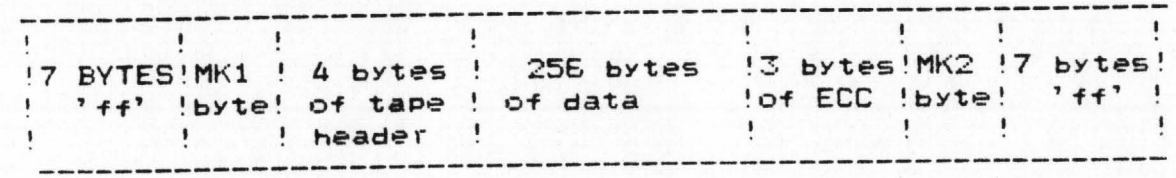
8.1 Sector format on the disk

The track is typically formatted for 32 sectors per track. The layout for every sector is as following:

Good track format	Bytes	Description
Gap 1	56	'4E'
Preambles	13	'00'
ID A.M.	2	
ID field	4	Cylinder hi, lo, head, sector
ECC	3	ECC bytes for ID field
Gap 2	2	'00'
Gap 2	12	'00'
Data A.M.	2	
Data	256	
ECC	3	ECC bytes for data field
Gap 3	2	'00'
Gap 4	15	'4E'

Bad track format	Bytes	Description
Gap 1	56	'4E'
Preambles	13	'00'
ID A.M.	2	
ID field	4	Cylinder hi, lo, head, sector
ECC	3	ECC bytes for ID field
Gap 2	2	'00'
Gap 2	12	'00'
Data A.M.	2	
Alternate ID	3	Alt. cylinder hi, lo, head
Data	253	
ECC	3	ECC bytes for data field
Gap 3	2	'00'
Gap 4	15	'4E'

8.2 Tape data block format



9.0 ERROR TYPE, ERROR CODE AND LED's DISPLAY DEFINITION

Should any error occur during operations, the controller posts error type and error code in Sense byte 0 and lights led's accordingly.

Type 0 (Disk Drive) Error codes		LED display
0	no error	<00>
1	no index pulse detected	<01>
2	no seek complete	<02>
3	write fault	<03>
4	drive not ready	<04>
5	drive not selected	<05>
6	no track 00 detected during recal	<06>
7	multiple drives selected	<07>
8	no sector pulse detected	<08>
9	alternate to a bad disk address	<09>

Type 1 (Controller) error codes		LED's display
0	ECC error in the ID field	<10>
1	uncorrectable data error in data field	<11>
2	reserved	
3	reserved	
4	record not found. Found correct cylinder and head but not sector.	<14>
5	seek error. R/W head postioned on a wrong cylinder or wrong head selected.	<15>
6	reserved	
7	reserved	
8	correctable data error in data field	<18>
9	all sectors on the track are bad, no valid alternate track information found.	<19>

A	reserved	
B	reserved	
C	incorrect track format - incorrect cyl. adr	(1C)
D	incorrect track format - incorrect head adr	(1D)
E	incorrect track format - incorrect sector adr	(1E)
F	ECC error in data field during read ID operation	(1F)

Type 2 (Command) Error codes		LED's display
0	invalid Command received from the host	(20)
1	illegal disk address, beyond max address	(21)
2	reserved	
3	drive type undefined	(23)

Type 4 (Tape transfer) Error codes		LED's display
0	excessive rewrite/reread errors during backup/restore - tape rewrite or reread count is more than 2048 during operations	(40)
1	tape drive is write protected	(41)
2	cartridge is not in	(42)
3	tape full - During backup operation, no more tape space on the cartridge while there are more disk blocks to be saved in the backup operation. Or during restore operation, no more tape space while there are more blocks to be restored according to the given command descriptor block.	(43)
4	detected a missing block error during restore operation	(44)
5	detected unexpected hole - controller detected unexpected BOT (beginning of tape) or EOT (end of tape) holes.	(45)
6	reserved	

7	reserved	
8	number of blocks to be restored, specified in the restore command, is greater than the number of blocks existing in the tape file.	(48)
9	file mark is not found in restore operation	(49)
A	cartridge been changed, repostion or erase first before do another backup; or illegally suppress BOT flag during tape read operations.	(4A)
B	tape drive is not ready	(4B)
C	timeout in tape read - controller did not detect valid data in tape read operations for more than 2 seconds.	(4C)
D	illegal tape drive track or illegal tape drive type	(4D)

type 5 (controller operation)error codes		LED's display
0	controller operation error 0	(50)
1	controller operation error 1 or may also caused by drive error or track not formatted in read/write operation.	(51)
	host bus parity error not posted in sense byte 0, just led's displayed and parity error bit set in completion status	(61)

10.0 MICRODIAGNOSTIC CAPABILITY

A diagnostic EPROM can be used to replace the Operational EPROM for Fault Isolation purpose. With this firmware, the user can initiate diagnostic routines, by switch setting to fault isolate the hardware on the controller. Any error from the diagnostic routine will be displayed on LED's. This option is useful in checking the board.

11.0 Tape related operational description

This section is devoted to help users better understand the tape command usage, their features, and limitations.

11.1 Tape command execution

(1) BACKUP - This controller can handle the File Backup as well as the traditional Image Backup.

a) File backup - In file backup, the user saves only part of the disk onto the tape. He defines the logical address on the disk where the backup starts, the number of blocks to be saved and the file I.D. # associated with this file.

An Erase command must be issued at the very first time of using the cartridge for backup operation. After an Erase pass, the controller positions the tape to the BOT (Beginning of Tape) positon and it is now ready for the first file backup on the tape.

The tape stops after the first file has been saved. If the second backup command is issued, the tape starts to move again and the next file is to be saved. The subsequent backups are handled similarly until the whole cartridge is filled.

Note: The user should try to save the files all at once because any interruption, such as power-down of the controller, removal of the cartridge and others, between backups may require extra procedure to realign the tape position. For instance, the controller was powered down after the last backup, the user must first reposition to the end of that file by using 'Position to End of File' command [see 11.2 of this section] . And then he may issue a file backup command for the next file. This is necessary because the controller lost vital information when powered down.

Should error occur at the completion, 'Sense request' is issued to get the error code. And if it is tape full error (error code = 43), the Sense bytes 1,2,3 contain the number of blocks that have been successfully saved onto the tape when the error was detected. By using these three-byte information, the user can easily configurate another CDB and continue this file backup onto another cartridge.

To further illustrate the above, an example is given as following:

a backup CDB is issued - logical address = 005000
file I.D. = 08
number of blocks= 002000

At the completion of backup and a sense request is issued and 4 sense bytes returned are -
sense byte 0 = 43
sense bytes 1,2,3 = 001FF0
This means tape is full and only 1FF0 blocks out of 2000 have been transfered.

Another backup CDB may be issued to save rest of the file onto another cartridge -
logical address = 005000 + 001FF0 = 00EFF0
file I.D. = 08
number of blocks= 002000 - 001FF0 = 000010

b) Image backup - Since the controller treats all backup operations in terms of files, the image backup is just a special case of the file backup where the whole disk is saved onto the tape. The user defines the starting logical address equal to 00, an arbitrary file I.D.#, and the maximum number of blocks available on the disk.

(2) RESTORE - With the built-in block offset feature in this command, the controller can perform the 'Unique block restore' in additional to the normal File Restore.

a) File restore - This command is to restore the data onto the disk. The user defines the file I.D. # , the starting logical address and the number of blocks to be restored on the disk. If the number of blocks to be restored specified by the user is greater than the number of blocks existing in the tape file, the controller still restores all blocks in the file and flags error (48) at its completion.

Upon receiving the command, the controller first positons the tape to BOT. If the tape retension flag is set in the CDB the controller runs the tape forward pass and reverse pass before positions it to BOT.
After BOT positioning, the controller then searches the file , restores it onto the disk and stops the tape.

If the user issues another restore command, the controller again positions to BOT before search and restore. However, he also has an option to set 'suppress BOT' flag to bypass BOT positioning or tape retension. This allows the user to quickly read the subsequent files following the current position.

b) Unique block restore - There is also a two-byte block offset defined by the user in the restore CDB. The controller can restore part of the file by using this offset to skip a certain number of blocks after file mark is found (file mark is the beginning of the file) and then it starts the restore operation from that block on.

Should error occur in the restore operation, 'sense request' is issued to get error code. If it is either tape full or missing block error (error code = 43 or 44) , the sense bytes 1,2,3 contain the number of blocks have been successfully restored onto the disk when the error was detected. By using these three- byte information, the user can easily configurate another restore CDB and continue this file restore from another cartridge or retry the missing block again.

To further illustrate the above, an example is given as following:

A restore CDB is issued- logical address = 005000
read block offset = 1000
number of blocks = 002000

At the completion of restore and a sense request is issued and 4 sense bytes returned are -
sense byte 0 = 44
sense bytes 1,2,3 = 001FF0
This means 1FF0 blocks have been restored when the missing block was detected.

Another restore CDB may be issued to retry the operation from the missing block to the end of the file
logical address = 005000 + 001FF0 = 006FF0
read block offset= 1000 + 001FF0 = 2FF0
number of blocks = 002000 - 001FF0 = 000010

(3) Read file from tape to host - This operation is very much similar to 'Restore' except blocks of data read from the tape are sent to the host rather than to the disk.

a) Read a file from tape to host - All concerns described in 'Restore' of this section are applicable here also.

b) Read a partial file from tape to host - All conerns described in 'Restore' of this section are applicable here also.

(4) Read tape file header - This operation is to read the tape I.D. block and the first few data blocks of the tape files and transfer them to the disk.

A 'Read tape file header' command as following is issued :

```
logical address      =   005000
number of files      =        8
number of blocks     =    4
suppress BOT flag    =    0
```

and assuming there are 40 files in the cartridge, namely file #0, #1, ... #38, #39. (the file #'s do not have to be in sequence as shown here)

Upon the completion of this command, the disk logical addresses containing:

```
5000 contains the I.D. block of File # 0
5001 contains the first data block in File # 0
5002 contains the second data block in File # 0
5003 contains the third data block in File # 0

5004 contains the I.D. block of File # 1
5005 contains the first data block in File # 1
5006 contains the second data block in File # 1
5007 contains the third data block in File # 1
       "
       "
501C contains the I.D. block of File # 7
501D contains the first data block in File # 7
501E contains the second data block in File  # 7
501F contains the third data block in File # 7
```

where
- 1st byte of I.D. block = 00
- 2nd = File #
- 3rd, 4th, 5th = Number of blocks existing in this file
- 6th = 01 if data was saved in whole drive mode
  - 00 if data was saved in split drive mode

Then the user can again issue the second 'Read tape file header' command with the 'Suppress BOT flag' set. Upon its completion, the disk logical addresses containing:

```
5000 contains the I.D. block of File # 8
5001 contains the first data block in File # 8
5002 contains the second data block in File # 8
5003 contains the third data block in File # 8

5004 contains the I.D. block of File # 9
5005 contains the first data block in File # 9
5006 contains the second data block in File # 9
5007 contains the third data block in File # 9

       "
       "
       "
501C contains the I.D. block of File # 15
501D contains the first data block in File # 15
501E contains the second data block in File  # 15
501F contains the third data block in File # 15
```

The example here shows that while there is not enough scatch disk area for holding all headers at one time, the user can utilize the 'Suppress BOT flag' to read 8 headers at a time.

There are two advantages of this commands.
Firstly, the user can retrieve the first few data blocks of multiple files in one command. Often time, these blocks contain the header information of that file.
Secondly, the user can read the headers of physically contiguous files on the tape and use them to build a tape file directory.

11.2 Appending data blocks on the existing cartridge -

This feature allows users to continue backups on the remaining portion of the cartridge tape after power-down, removal of cartridge, or other interruptions causing mis-alignment of the read head in the tape drive. Therefore, utilization of a whole cartridge is warranted.

Assuming there is file #51, which is the last file in the cartridge. In order to append data blocks, the user issues a 'Position to End of File' command to position to the end of the last file on the tape.

After receiving this command, the controller performs a search on File I.D. #51 and positions to the end of this file. At the completion of this command, it is ready to backup more files on the remaining unused portion of the tape.

11.3 Read next N blocks -

This feature allows the user to continue to read more blocks from the current file.

For example, assuming there is file #25 of 10 data blocks in the cartridge and a 'Read file from tape to host' command is issued to the controller:

```
          File I.D. =  # 25
number of blocks =  5
```

At the completion of this command, the user then decided there are more information on this file to be read. Now he can issue the second 'Read file from tape to host' command

```
          File I.D. =  Don't care
number of blocks =  5
read next N block flag = set
```

The controller will ignore the file search process and go ahead to read the next 5 blocks from file #25.

Since the file mark has been passed after the first command, another file search on this file would involve the tedious BOT. By using this flag, the user can greatly reduce his access time to the next 5 blocks of data on the tape.

The user should not confuse this flag with Suppress BOT flag. Under Suppress BOT flag, though the controller does not do BOT, it still performs the normal file search process and of course, file mark #25 can not be located.

11.4 Error Handling and Reporting

Request Statistics - The controller maintains a 11-bit counter(up to 2048) to keep track of the re-write or re-read tries in any single backup or restore operation. Regardless of error or not at the completion of the tape-related operations, it is advisable to issue 'Request Statistic' to get this count, which can be used to determine any degradation in the quality of the tape drive and cartridge.

11.5 User's concerns in tape operations

(1) Tape gaps between files - the controller puts a gap on the tape between files during backup operations. Therefore increasing number of files in the cartridge decreases the percentage in tape utilization.

(2) Interleave constant - To accomodate efficient disk/tape transfer operations, the disk should be formatted with an interleave constant of 11 (decimal value). Otherwise, there will be a significant penalty in tape utilization.

Interleave constant is a number used to assign relative positions to 32 sectors throughout the disk track. For example, an interleave constant of 11 assigns the sectors in the following sequence:

0, 11, 22, 1, 12, 23, 2, 13, 24, 3, 14, 25, 4, 15, 26, 5, 16, 27, 6, 17, 28, 7, 18, 29, 8, 19, 30, 9, 20, 31, 10, 21.

(3) Host transfer rate - If 'Read File from Tape to Host' command is used, the user should make sure that the host can maintain a data transfer rate equal to that of the tape (about 11.7 usec/byte). However, for the files of less than 3584 bytes (14 blocks), this concern is waived because of the controller's sector buffer.

**** WARNING **** If the host transfer rate is slower than 11 msec per block, the user MUST NOT specify more than 14 blocks at a time.

12.0 USER'S GUIDE FOR REQUIRED JUMPERS AND SWITCH SETTINGS

12.1 Jumpers and switching settings on the disk drive

For Seagate ST506 & ST412 drives:

Drive select jumper DS1	for	LUN #0
Drive select jumper DS2	for	LUN #1
Drive select jumper DS3	for	LUN #2
Drive select jumper DS4	for	LUN #3
H shunt not selected	for	No Half Step option
D shunt selected	for	Defeat Auto Recal
R shunt not selected	for	No Radial Connection

12.2 Termination on the last disk drive

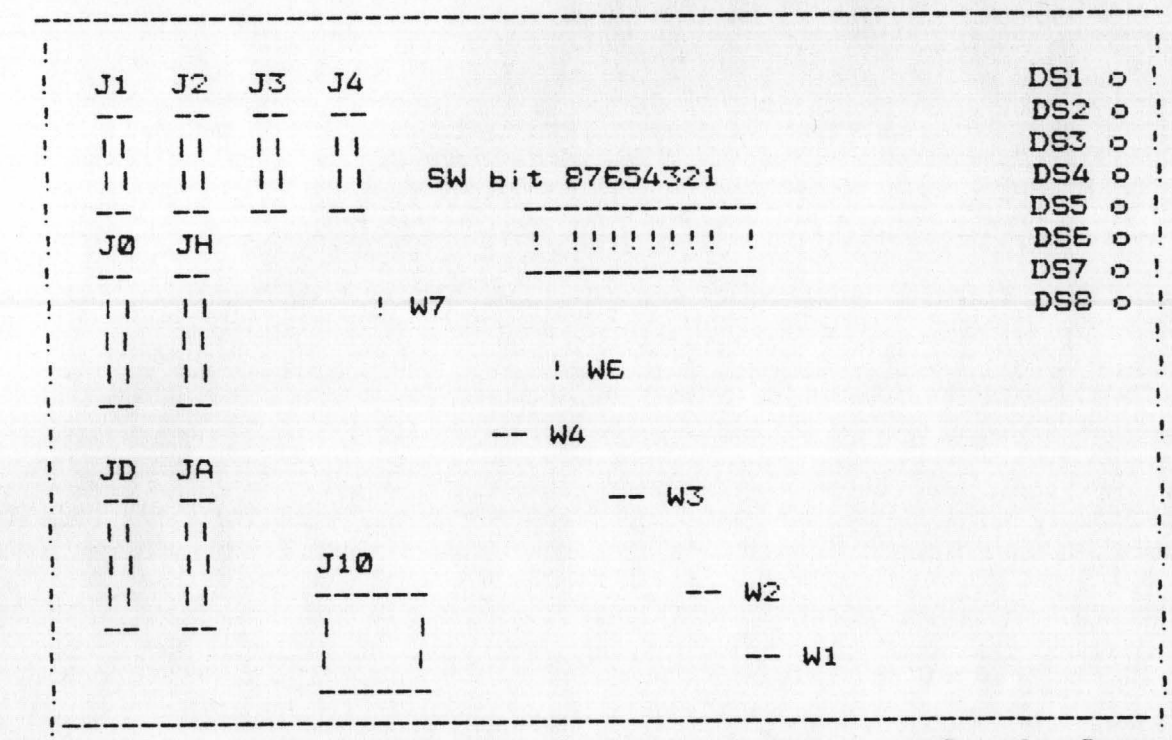
Only one drive in the system should be terminated. This drive should be physically located at the end of the cable and terminated by the removable 220/330 ohm resistor pack near the cable connector.

12.3 JUMPERS AND SWITCH SETTINGS ON THE TAPE DRIVE

This controller can only interface to one Archive tape drive, Tape LUN 0 should be selected as:

pin #4 & pin #11 shunt = Tape LUN 0

12.4 JUMPERS AND SWITCH SETTINGS ON THE CONTROLLER



J0	34 pin 0.10" spacing connector to Disk's A cable
J1, J2, J3, J4	20 pin 0.10" spacing connectors to Disk's B cables
JH	50 pin 0.10" spacing connector to Host cable
JD	not connected
JA	50 pin 0.10" spacing connector to Archive cable
J10	6 pin AMP #1-380999-0 connector to power cable where pin5 connected to +5V and pin6 to ground
DS 8-1	LED's for error code, where DS8 is the most significant bit position.

User should orient the board as shown above and check all shunts are in proper positions.

W1	Jumper left adjusted (Used in Manufacturing)
W2	Jumper right adjusted (Used in Manufacturing)
W3	Jumper left adjusted (Product Strap)
W4	Jumper left adjusted (Used in Manufacturing)
W6	open

W7 Jumper bottom adjusted (User to disable parity)
Jumper top adjusted (User to enable parity)

Switch

Keys 5,4,3,2,1 are DON'T CARE in this controller

Keys 8,7,6 are bits to define controller address ID. For example, the user wants to select controller ID as #4 which will response to DB4 in host selection, the setting is

Keys 8 = ON
7 = OFF
6 = OFF

See Section 7.2 for host selection operation.

12.5 DISK PARAMETER DEFINED BY USER

For Extended Series 2000 controllers, the user MUST issue 'Define Disk Parameter' command to the controller after power-up to define all LUN's in the system.
(See Class 6 command description in Section 4.2)

However, bacause the track 00 (logical address 000000 - 00001F) is often time used as a boot area, the controller allows the user to read or write to this track without those parameters defined.

It should also be noted that the controller can only allow up to two different disk drive types throughout four LUN's in the system. LUN 0 belongs to one type and LUN1,2,3 belong another type. Therefore, as the user issues this command to the controller for any one from LUN1,2,3 group, all three of them are defined with the same type simultaneously.

12.6 TAPE PARAMETER DEFINED BY USER

For this controller, the user MUST also issue 'Define Tape Drive Type' command to the controller after power-up to define the Drive Model and Whole/Splite drive modes in his system.
However, the controller ASSUMES it is 20 Mbytes drive and operated as whole 4-track mode if they are not defined.