SMA STANDARD

Magnetic Media Interchange Specification

SMA: 201

March 1987

SPECTRUM MANUFACTURERS ASSOCIATION

sma

-- NOTICE --

SMA standards are designed to serve the public interest through eliminating misunderstandings between manufacturers and purchasers, facilitating interchangeability and improvement of products, and assisting the purchaser in selecting and obtaining, with minimum delay, the proper product for his particular need.

Some material contained herein is designated as proprietary by individual member companies of SMA listed below. Any unauthorized use of such proprietary information is prohibited.

Published by
SPECTRUM MANUFACTURERS ASSOCIATION
9740 Appaloosa Rd., Suite 104
San Diego, CA 92131

Foreword: The purpose of this document is to establish a standard for interchangeable magnetic media within the Spectrum Manufacturers Association community. The goal of the standard is to assure continuing data compatibility between different SMA implementations which have physically compatible media. Additionally, it is meant to serve as a description of SMA magnetic media protocols so that other operating systems may exchange data with an SMA system through the off-line tape and diskette mechanisms.

Within this document, it becomes evident that inconsistencies and incompatibilities exist on both hardware and software levels. In some of these cases, the standard outlines restrictive guidelines which should be followed to maximize portability. The Technical Committee of SMA is agressively pursuing these areas of deficiency, with the intent of establishing and publishing standards which will improve the situation in these areas of the media interchange.

Change Notice: Draft 1.5 contains changes made to sections 4 through 7 in regards to 1/4" tape cartridge standards. These changes were approved by the SMA Technical Standards Committee on August 6, 1986.

Draft 1.7 contains changes made to wording within the changes made as Draft 1.5. These changes were approved by the SMA Technical Standards Committee on October 2, 1986. Other changes were made to sections 3.2, 4.4 (inserted), 4.11 and 4.15.

Draft 1.8 contains a warning added to section 4.16.

The SMA Executive Board wishes to thank the following individuals and organizations for their contributions to the preparation of this document:

D. Credicott,	Nixdorf Computer Software Co.
I. Sandler,	CIE Systems, Inc.
J. Gallant,	Prime Computer, Inc.
D. Harman,	Systems Management, Inc.
C. Saunders,	Fujitsu Microsystems Of America, Inc.
C. Wilson,	General Automation, Inc.
T. Steforos,	Altos Computer, Inc.

CONTENTS

CONTENTS

1.0 Introduction

1.1 Inclusions: On an SMA system, data can be produced in several ways, which affect the format of the actual data on the magnetic media. This standard covers the two most common mechanisms: (1) the TCL "T-DUMP" verb, which is used to dump file items to a tape; and (2) direct input and output from programs written in the SMA/BASIC language.

Additionally, the standard includes a list of tape handling TCL verbs, with a brief functional description of each one.

1.2 Exclusions: Excluded from this version of the standard are the two other techniques of creating data tapes under an SMA system. These include: (1) the TCL "SAVE" verbs; and (2) the commands which deal with spooler input and output on tape. These two areas will be covered in future revisions of the standard.

1.3 Philosophy: Because of the wide variety of removable media utilized on the SMA systems, complete interchange and maximum functionality cannot always be achieved. As a hypothetical example, one class of device may support a physical blocksize of up to 32000 bytes, whereas another may support a maximum of 8000 bytes. In this case, the standard would be compelled to adopt a blocksize upper limit of 8000, in order to assure that tapes produced by either system could be processed by the other.

The restrictions, limitations, and recommended operating procedures set forth in this document have been selected by a "majority rules" philosophy. That is, when more than one choice exists within an area that requires a unique standard, the selection was determined on the basis of maximizing benefit to the largest number of users and manufacturers. In many cases, the resulting standard is a "lowest common denominator" within the community, and may be exceeded by several of the implementations.

It must be emphasized that nothing within this document was intended to be, or should be construed as, a favorable or unfavorable comment on the equipment, software, or expertise of any vendor.

1.4 Changeability: For a number of reasons, this standard must be considered as a dynamic document, subject to revision and extension. First of all, there are technical areas upon which a standard has not yet been agreed, as outlined in the "Exclusions" paragraph. Secondly, the content of the standard is heavily influenced by the hardware and software capabilities of the various manufacturers, both of which are constantly changing. Thirdly, there is substantial diversity among the tape and diskette devices regarding the handling of certain conditions, such as end-of-volume detection. Establishing a common standard in areas such as these may require operating system modifications across all vendors.

1.5 Restrictions: Because of the wide variety of removable media utilized on the SMA systems, complete interchange is limited. This limitation is most pronounced in the ways which the hardware deals with the end-of-volume situation. As a consequence, only single-volume media can be processed with any reliability across different vendor equipment. The long term solution for this deficiency will most likely require a common software circumvention within the operating system itself.

In the meantime, to maximize transportability, media should be constructed in such a fashion that no data file spans across more than one volume. If the amount of data physically requires multiple volumes, then programmer action should be taken to subdivide the data into groups such that each group can be contained on a separate volume as an integral file.

2.0 Definitions

2.1 Scope: This chapter provides definitions of the terms, phrases, and notational abbreviations that are used within this document. Some of these definitions carry a broader scope in the context of this standard than they usually imply, and are therefore included.

2.2 Terms and Phrases:

Attachment size: When a magnetic media device is attached (made available) to a process, the logical blocksize is either specified in the command or implied by a default value. The value remains constant until it is explicitly changed or until the device is detached from the process. All data blocks (excluding any label records) will be written at this logical blocksize. Whenever possible, the physical blocksize should be the same as the logical blocksize.

2.2 Terms and Phrases (continued):

EOF: Acronym for "end of file".

Interchangeable magnetic media: This term refers to a data recording capability, utilizing magnetic technology, in which the component actually storing the data can easily be dismounted from one computer system and mounted on another. Within this standard, the term refers specifically to tape and floppy diskette facilities.

System delimiters: SMA systems utilize certain hexadecimal characters as delimiters within the file system. The name, acronym, hexadecimal value, and usage are summarized below:

Segment Mark	SM	X'FF'	Delimits items (records)
Attribute Mark	AM	X'FE'	Delimits attributes (fields) within an item (record)
Value Mark	VM	X'FD'	Delimits multiple values within an attribute (field)
Sub-Value Mark	SVM	X'FC'	Delimits multiple sub-values within a value
Buffer Mark	BM	X'FB'	Buffer control

Tape: Within the context of this standard, the term "tape" refers to any of the commercially available magnetic media which is interchangeable. Specifically, it includes not only conventional 1/2" reel-to-reel tape facilities, but also includes 1/4" cartridge tape and floppy diskette technology.

TCL: Acronym for "terminal control language".

2.3 Abbreviations:

Within this standard, the following abbreviations are used in describing the syntax of tape handling commands:

blocksize	Tape attachment size, in bytes
file.name	Source (T-DUMP) or destination (T-LOAD) file
item.list	List of item identifiers
mod.list	List of modifiers for special functions
records	Number of records
sel.list	List of selection criteria
seq.list	List of parameters to specify sort sequence

3.0 TCL Tape Handling

3.1 Scope: This chapter lists the tape handling verbs which are available within TCL, illustrates the format of the command, and gives a brief overview of the function performed. This information is provided for guidance purposes only, and is not intended as a complete syntactical or functional description. Detailed information regarding these verbs should be obtained from the relevant vendor documentation.

3.2 TCL Tape Handling Verbs:

S-DUMP filename {item.list} {seq.list} {sel.list} {mod.list} {HEADER "text"} {(options)}

Copies selected file items to tape, in sorted sequence.

T-ATT {blocksize}

Attaches a tape drive and establishes the blocksize.

T-BCK {records}

Backspaces tape by number of records. If records is not specified, the tape is moved back to the last previous filemark, or beginning of tape, if there are no filemarks. If records is specified, the tape will stop if it encounters a filemark or the beginning of tape. See Section 7.0 for restrictions on use of this verb.

3.2 TCL Tape Handling Verbs (continued):

T-DET

Detaches a tape drive from a process.

T-DUMP filename {item.list} {sel.list} {mod.list}
{HEADER "text"} {(options)}

Copies selected file items to tape.

T-EOD

Moves tape forward to end of data.

T-FWD {records}

Moves tape forward by the number of records. If records is not specified, the tape is moved forward to the next subsequent filemark, or to the end of tape if there are no filemarks. If records is specified, the tape will stop if it encounters a filemark.

T-LOAD filename {item.list} {sel.list} {mod.list}
{(options)}

Loads selected items from tape into disk file.

T-RDLBL

Reads and displays label information.

T-READ {(options)}

Reads and displays tape data record(s).

T-REW

Rewinds tape to load point.

T-WEOF

Writes a filemark on tape.

T-WTLBL {text}

Writes a tape label.

4.0 T-DUMP Tapes

4.1 Scope: The TCL verbs T-DUMP, S-DUMP, and T-LOAD provide a means of transporting selected file items from one system to another. This section describes the format of tapes produced by the T-DUMP verb and its companion S-DUMP for sorted output, and identifies the conventions to be followed which will maximize the portability across the various implementations. Other than appearing in sorted sequence, the data on an S-DUMP tape is identical to the T-DUMP version. Thus, all references to T-DUMP apply to S-DUMP, except for issues of item sequence.

4.2 File Layout: A T-DUMP file consists of a label record, zero or more data items, an end-of-file code, and a terminating filemark. The filemark implies the end of file condition, indicating that no more records are associated with this logical data file.

<Label record>

<Data item(s)>

<EOF code>

<Filemark>

4.3 Multiple Files: A tape may contain one or more logical data files, each of which follows the structure defined in the preceding paragraph. That is, each logical file consists of a label record, data record(s), its EOF marker, and a terminating filemark.

4.4 Empty Files: Files which contain no data shall be written in the follow structure:

<Label Record>

<EOF Code>

<Filemark>

4.5 Leading Filemarks: Leading filemarks on the tape are not supported in the T-DUMP format. A label record is expected to be the first block on the tape.

4.6 Trailing Filemarks: Two consecutive filemarks serve as an indication that no more files are recorded on the media.

4.7 Label Records: Label records are normally created by the T-DUMP process, and read by the T-LOAD process. The size and content of the label is described in section 6, Tape Labels, and in section 7, Hardware Capabilities, within this standard.

4.8 Block Size: The TCL command "T-ATT" sets the attachment blocksize. All data records are normally written with both physical and logical blocksize equal to the attachment blocksize. Label records are normally written with both physical and logical blocksize equal to 80 bytes. Certain exceptions are made for the characteristics of various devices and are detailed in section 7, Hardware Capabilities, within this standard.

4.9 Attachment Size: Although a wide range of physical blocksizes are possible on the various devices, the following sizes should be used for maximum portability:

1/2-inch tape	4000 bytes
1/4-inch tape	8192 bytes
Floppy diskette	500 bytes

4.10 Data Format: The format of the data records on a T-DUMP tape consist of file items, placed end to end, spanned across physical blocks as necessary. Unless the output was created with the S-DUMP verb or via a select list, the items are recorded in the same hashing sequence in which they are contained in the original source file. Special codes are used to represent the logical end of file condition. If the last data block in a file is not completely full, it is padded with a "fill" character.

4.11 Data Item Format: Generally speaking, a single item on the tape consists of the item-id terminated by an attribute mark (X'FE'), the datafield attributes of the item (including any attribute marks, value marks, and subvalue marks), followed by a buffer mark (X'FB') which ends the individual item. Graphically, this can be illustrated:

<item.id>	Item id Format: variable length character string
X'FE'	Attribute mark, terminates item id
<Attribute(s)>	Data fields within item Format: Each attribute is a variable length character string terminated by its own attribute mark, with any value or subvalue marks left in their original position
X'FB'	Buffer mark, terminates item The X'FB' immediately trails the terminating attribute mark of the last attribute in the item. Sequences of X'FEFB' cannot be embedded within items.

4.12 Items Larger than 120 Bytes: The format described above is modified slightly for items whose overall length (including item-id and all system delimiters) is greater than 120 bytes. In this case, a special two-byte buffer control code is inserted into the data after every 120 bytes of data. The buffer control code consists of a segment mark (X'FF') and a buffer mark (X'FB'). The interrupted data resumes immediately behind the two control bytes.

4.13 Pointer-Item Format: Transmitting pointer-type items via magnetic media may cause unpredictable results across SMA implementations, and should be avoided. A reliable standard for this area is yet to be determined.

4.14 Multiple Volumes: Because of the high diversity with which tape drives and disk drives detect and handle the end-of-volume condition, no single logical file should span across more than one volume when transportability is needed. If the amount of data requires more than one volume, it is advisable to subdivide the data into groups such that each group can be contained on a separate volume as an integral file.

4.15 Block Padding: As described in the paragraph on blocksizes, all physical records are written on the tape at a fixed length. Any unused buffer space behind the EOF code up to the attachment size will be filled with a buffer mark (X'FB').

On some systems, the double buffering routines in the tape drivers will cause an additional block to be written following the one containing the EOF code. This block is padded completely with the buffer mark character (X'FB').

4.16 Codes: Special codes are utilized in the T-DUMP tape format. Summarized below, they are:

_L	Identifies label record Format: segment mark (X'FF') and the character 'L'
_X	Identifies logical end of file (EOF Code) Format: segment mark (X'FF') and the character 'X'
X'FFFB'	Buffer control (after every 120 bytes) Format: segment mark (X'FF') and buffer mark (X'FB')
X'FEFB'	Item terminator. Format: attribute mark (X'FE') and buffer mark (X'FB'). Following buffer marks (X'FB') fill to attachment size.

Warning: These codes cannot be embedded within data on T-DUMP format tapes. It is especially important to note that the X'FEFB' sequence implies that X'FB' codes should not be stored as data where they can occur as the first character of an attribute.

5.0 SMA/BASIC Tapes

5.1 Scope: The SMA/BASIC programming language provides for input and output on a sequential magnetic media. This standard describes the conventions which, if followed, will maximize the portability of data between different vendor implementations.

5.2 File Layout: An SMA/BASIC file consists of zero, one, or more data records followed by a filemark. The filemark implies the end-of-file condition, indicating that no more records are associated with this logical data file.

<Data record(s)> .

<Filemark>

5.3 Multiple Files: A tape may contain one or more logical data files, stacked one behind another. There is no inherent coding within the files to identify them from each other; the programs which read the data files must process them in the same order in which they were created. Each individual file must follow the structure defined in the preceding paragraph. That is, each logical file consists of data records (zero, one, or more) terminated by a filemark.

5.4 Leading Filemarks: A leading filemark at the immediate beginning of the tape will imply that the first file on the tape contains no data records.

5.5 Label Records: A standard for the writing of labels under SMA/BASIC is yet to be determined. However, SMA/BASIC will automatically bypass any existing SMA label records when a tape is read.

5.6 Block Size: The TCL command "T-ATT" sets the attachment blocksize. All data records are normally written with both physical and logical blocksize equal to the attachment blocksize. Label records are normally written with both physical and logical blocksize equal to 80 bytes. Certain exceptions are made for the characteristics of various devices and are detailed in section 7, Hardware Capabilities, within this standard.

5.7 Attachment Size: Although a wide range of physical blocksizes are possible on the various devices, the following sizes should be used for maximum portability:

1/2-inch tape	4000 bytes
1/4-inch tape	8192 bytes
Floppy diskette	500 bytes

5.8 Data Format: The expression referenced in the SMA/BASIC "WRITET" statement is written on tape in an individual block padded on the right with spaces up to the attachment blocksize. A null or oversized block invokes the "ELSE" clause, and is not written to the tape. The content of the data itself is determined exclusively by the logic of the SMA/BASIC program which generates the tape.

5.9 Multiple Volumes: Because of the high diversity with which tape drives and disk drives detect and handle the end-of-volume condition, no single logical file should span across more than one volume when transportability is needed. If the amount of data requires more than one volume, it is advisable to subdivide the data into groups such that each group can be contained on a separate volume as an integral file.

5.10 Codes: The data in an SMA/BASIC tape file is scanned for only one special character or code, the segment mark (X'FF'). The label record is identified by a segment mark (X'FF') followed by the character 'L'. The content of the data itself is determined exclusively by the logic of the SMA/BASIC program which generated the tape. However, any data following an imbedded segment mark (X'FF') in the data block is truncated during a read operation.

5.11 SMA/BASIC Tape Commands: There are four statements available in the SMA/BASIC programming language for the manipulation of tape files. Refer to the SMA/BASIC standard for syntactical and usage rules.

WRITET	Write tape
READT	Read tape
WEOF	Write end of file (filemark)
REWIND	Rewind tape

6.0 Tape Labels

6.1 Scope: This section describes the content and format of SMA tape labels.

6.2 File Layout: When present, tape labels precede the first data block in a file. No filemark automatically precedes or follows the label. Thus, the structure of a labeled file is:

<Label record>

<Data block(s)>

<Filemark>

6.3 Label Record Format: Label records are always considered to have a logical blocksize of 80 bytes. The physical blocksize and layout of the label with that physical block are detailed in section 7, Hardware Capabilities. The contents and format of the label block is indicated below. A single blank separates each field, except that two blanks separate the date and time fields.

Element	Contents
_L	Label record code Format: segment mark X'FF' and character 'L' Positions: 1-2
bbbb	Block size Format: 4 hexadecimal characters Positions: 4-7
<time>	System time when created Format: HH:MM:SS Positions: 9-16
<date>	System date when created Format: dd mon yyyy Positions: 19-29
<labeltext>	Label text Format: content depends on usage (see below) Positions: 31-76
^rr	Reel number (beginning with 01) Format: attribute mark X'FE' followed by 2 digits Positions: 78-80

6.3 Label Record Format (continued):

The format and content of the <labeltext> element within the label record depends on whether the label was created by the "T-DUMP" or "T-WTLBL" command. Label records written by the spooler use a <labeltext> element with a special format and content.

T-DUMP: When the label is created by the T-DUMP command, the <labeltext> field in bytes 31 through 76 is formatted as follows:

<filename> Name of source file from T-DUMP
Format: variable length character string, ending with a blank
Positions: 31-variable

<heading> Quoted information in HEADER option of T-DUMP
Format: variable length character string
Positions: ends in byte 76

T-WTLBL: When the label is created by the T-WTLBL command, the <labeltext> field in bytes 31 through 76 is formatted as follows:

<text> Optional text following T-WTLBL command
Format: variable length character string
Positions: 31-variable

7.0 Hardware Capabilities

7.1 **Scope:** This section identifies the media capabilities of the various magnetic devices supported under SMA implementations. Only those gross categories allowing off-line interchanges are considered: tape, cartridge tape, and floppy diskette. Details pertaining to the specific device usage on a given implementation should be obtained directly from the manufacturer.

This chapter also identifies the hardware characteristics and usage conventions which will maximize portability across different systems.

7.2 Hardware Standard - 1/2 Inch Magnetic Tape:

Data width (tracks per byte):
Nine

Recording density (bytes per inch):
1600 (800, 3200, and 6250 are also supported on some equipment, but not universally)

Data Block Size:
4000 bytes, logical and physical

Label Block Size:
80 bytes, logical and physical

Indicator for beginning of tape:
Reflective marker on front edge of non-recording surface of tape

Indicator for end of tape:
Reflective marker on back edge of non-recording surface of tape

Filemark indicator:
ANSI standard tape mark written and detected by the hardware

Applicable ANSI standards:
X3.39-1973

7.3 Hardware Standard - 1/4 Inch Cartridge Tape

Hardware Device Standard:
QIC 24 standard for 9 track tape controllers

Recording directions:
Serpentine, as specified in the QIC 24 standard for 9 track tapes.

Data Block Size:
Physical block size: 512 bytes (QIC 24)
Logical block size: 8192 bytes (16 physical blocks)
Logical data blocks do not include label blocks or filemark blocks.

Label Block Size:
80 bytes logical, 1 physical 512 byte block where the first 80 bytes of the physical block is the logical label record and the remaining bytes are unused.
Logical data blocks do not include label blocks.

Filemark indicator:
QIC 24 standard tape mark. Logical data blocks do not include filemarks.

TCL Verb Usage Restriction:
The verb "T-BCK" is not supported. The hardware standards do not support backward movement of the tape by record or file.

7.4 Hardware Standard - 5.25 Inch Floppy Diskette

Format:
IBM PC/XT compatible (512 byte sectors, 9 sectors per track, double-sided, double-density with 40 tracks per side, recorded at 48 tracks per inch)

Data Block Size:
500 bytes logical, 512 bytes physical, where the first 4 bytes are reserved for the filemark indicator, and the next 8 bytes are unused (and reserved), and the remaining 500 bytes are used for the logical data.

Label Block Size:
80 bytes logical, 512 bytes physical, where the first 4 bytes are reserved for the filemark indicator, the next 8 bytes are unused (and reserved), the next 80 bytes are the logical label record, and the remaining bytes are unused.

Filemark indicator:
Character string "EOF" followed by segment mark (X'FF') in the first four bytes of block (recognized by the software but not special to the hardware).

This is the last page.