DEC STANDARD 167 REV. A

VOL. IDENT. FOR REMOVABLE DISK PACK

TITLE: Volume Identification for Removable Disk Pack Disk Systems

ABSTRACT: This standard defines the format and location of the volume identification block required to allow disk packs of removable disk-pack systems to be identified in all CPU families. This block will enable operating systems to identify the origin and format of a volume and decide if the volume can be processed. This standard also defines a standard error message for volumes that can not be processed.

DATE	ECO #	AUTHOR	APPROVED	REV	SEC	PAGES
19-May-77	----	D.Lewine	C.Noelcke	A	--	--

Size	Code	Number	Rev.
A	DS	EL00167-00	A

digital

PAGE REVISION CONTROL

PAGE NO.	PAGE REVISIONS	PAGE NO.	PAGE REVISIONS
1	A		
2	A		
3	A		
4	A		
5	A		
6	A		
7	A		
8	A		
9	A		
10	A		
11	A		
12	A		
13	A		
14	A		
15	A		
SEC. REV	--	SEC. REV	
STD REV	A	STD REV	

DEC STANDARD 167

Volume Identification for Removable Disk Pack Disk Systems

19 May 1977

D. Lewine, Author

ABSTRACT

This standard defines the format and location of the volume identification block required to allow disk packs of removable disk-pack systems to be identified in all CPU families. This block will enable operating systems to identify the origin and format of a volume and decide if the volume can be processed. This standard also defines a standard error message for volumes that cannot be processed.

digital

1.0 INTRODUCTION

This standard is applicable for all removable disk-pack disk systems. For removable disk-pack disk systems that cannot comply for technial or business reasons, the implementor must secure a waiver under the provisions of DEC standard 001. The RX01 (Floppy) disk system is an example of a system that cannot comply for technical reasons. On the other hand, the RP03 and RP02 need not comply because they are mature products and changing them would be uneconomical.

1.1 Motivation

1.1.1 Why Have the Standard? This standard will allow all users of removable disk-pack disk systems to interchange data. It will prevent loss of data if the wrong pack is mounted and an attempt is made to access data.

1.1.2 Why Standardize Now - To have an effective standard, all users of removable disk-pack disk systems must comply with this standard. This standard is currently being used by TOPS-20, TOPS-10, and FILES-11 (RSX-11D and RSX-11M).

1.2 Goals

1.2.1 To identify all removable disk packs as to volume, owner name, and operating system type.

1.2.2 To produce a clear error message if an unsupported pack format is mounted.

1.2.3 To prevent loss from writing on the wrong pack.

1.2.4 To ensure that disk engineering provides a hardware error flag when a user attempts to read a disk pack in a mode other than the one in which it is written.

1.3 Non-Goals

1.3.1 It is not a goal to allow mixed disk word lengths on the same pack; however, nothing in this standard forbids that configuration.

1.3.2 It is not a goal of this standard to provide security or volume identification for other manufacturers disk packs unless the data are formatted as specified by this standard.

1.3.3 Until future disk technologies are defined within DIGITAL, this standard will not be applicable to said technologies. As new

technologies are defined and adapted by DIGITAL, this standard will be revised to accomodate them.

1.4 Scope

1.4.1 To What Does This Standards Apply - This standard applies to all software that accesses systems with removable disk packs. If the software accesses the disk via an operating system that complies with this standard, no special action is necessary.

1.4.2 Customer Conversions - Customers utilizing the following operating systems will not require conversions to comply with this standard:

TOPS 10
TOPS 20
RSX-11M (V03)
IAS (V01)

All other customers will have to convert to comply with this standard if they desire standardization. All software releases after the effective date of this standard will support this standard if removable disk-pack disk systems are supported in the software release.

1.5 History of Previous Standardization Efforts

This document was first written in September, 1974, to solve an immediate product need of TOPS-10 and FILES-11. This edition improves the description of the standard and modifies the content of the standard to limit its subject matter to the identification of disk pack volumes. Additional systems specific information is contained in systems specific sections of this standard.

1.6 Related Current Standards

Floppy Disk Standard, DEC Standard 154
System Messages Standard (being developed)

1.7 Future Standards Activities

A more general volume identification standard should be written to cover all media. In addition, when the System Messages Standard becomes available, the appropriate Engineering Change Order will be

initiated to ensure conformity of messages contained in this standard. A need also exists for a standard defining the blocks that must be provided error free on new packs.

2.0 TERMINOLOGY

2.1 Mode

In this standard, "mode" refers to the disk-pack data format, i.e., the number of bits per word, the number of words per sector and the number of sectors per track. The two modes currently in use are 16-256, 22-sector mode and 18-256, 20-sector mode. These two modes are commonly referred to as "22-sector mode" and "20-sector mode," respectively. New modes will be introduced as new disk systems are developed and become standard DIGITAL disk products. The format for specifying mode is as follows:

"nn-aaa, bb sector mode"

nn = number of bits per word in decimal
\- = hyphen
aaa = number of words per sector in decimal*
, = comma
bb = number of sectors per track in decimal

Example:

16-128, 32 sector mode
This is read, "Sixteen dash one twenty eight, thirty-two sector mode".

It means there are

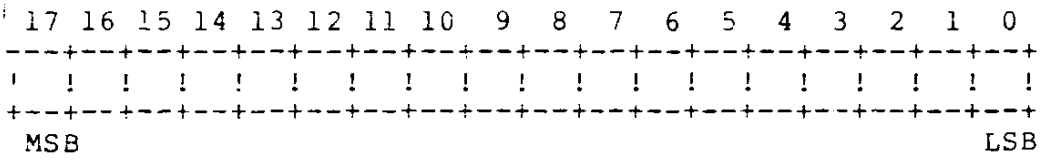
a. 16 bits per word,
b. 128 words per sector, and
c. 32 sectors per track.

2.2 Word

In this standard, a word is the basic unit of data written to or read from a disk. A disk word is currently a sequential series of 16 or 18 bits. Future disk systems may introduce different word lengths. It should be noted that even though processor word lengths vary (32,36, etc.), the word length written to the disk is always the length specified in the mode of the disk defined in Section 2.1. A disk word format is as follows:

---

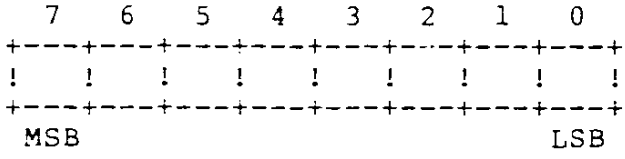
*Words per sector means data words available to software. It does not include words contained in the hardware header information.



This illustration is an example of an 18-bit disk word. A 16-bit or any other disk word length would follow this convention. The word is right-justified with the LSB in bit position 0.

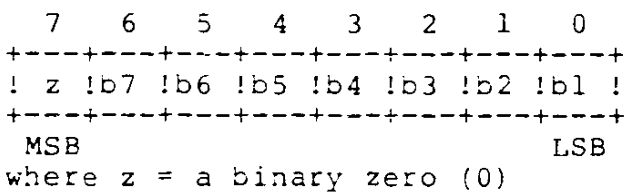
2.3 Byte

A byte is an 8-bit data item. The bits of a byte are formatted as follows:



All data written to a disk as specified by this standard are written in words containing two bytes. Each byte contains one ASCII 7-bit character.

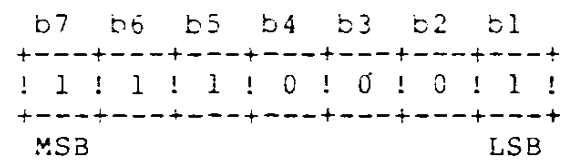
2.3.1 ASCII 7-Bit Character in Byte Format - The bits in a 7-bit ASCII character are labeled b1 to b7 starting from the least significant end. The mapping between the ASCII notation and the bit notation for bytes on a disk is given by



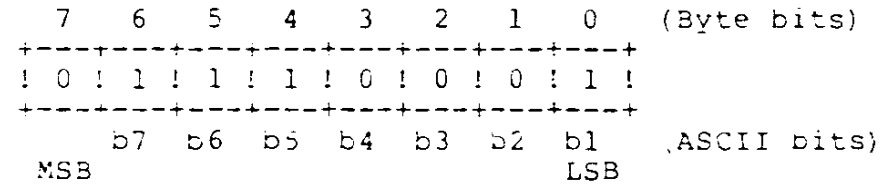
When an ASCII character is put in byte format it appears as follows:

Assume the ASCII character "q" 161(8)

The binary representation is

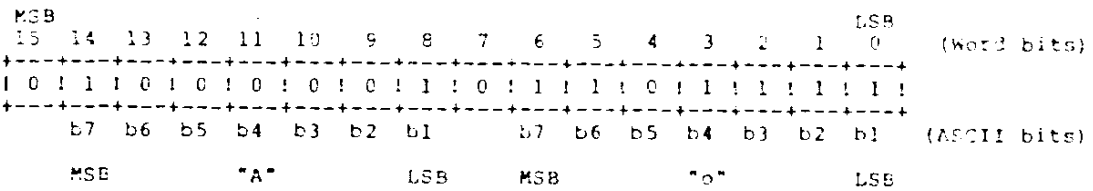


In byte format it appears

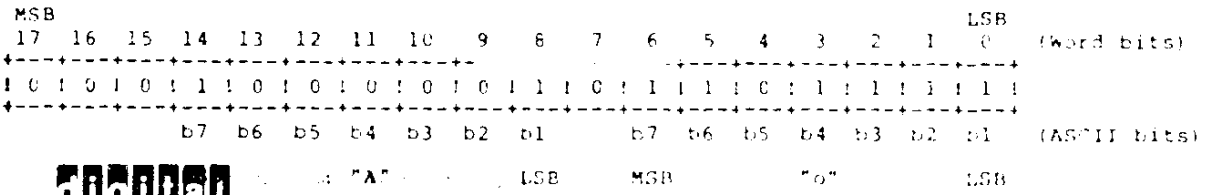


The ASCII character is right-justified in the byte format and Bit 7 of the byte format must be a zero. It is reserved for the future for 8-bit ASCII.

2.3.2 ASCII 7-Bit Characters in Word Format - Two ASCII characters contained in a 16-bit word appear as follows: Assume the two ASCII characters are small "o" 157 (8) and capital "A" 101 (8). The characters would appear as follows with small "o" being the first character.



The two ASCII characters in an 18-bit word appear as follows:



2.4 DISK LOGICAL BLOCK

A disk logical block number uniquely identifies a block on the disk. That number is independent of the physical addresses required by the hardware to identify cylinder, track, and sector. A disk logical block is also independent of the sector size. Logical disk blocks are numbered consecutively starting at 0. The logical block numbering is independent of whether the number of sectors per track or the number of tracks per cylinder is an integer power of 2.

3.0 CONFORMANCE

DECsystem-10 and DECsystem-20 software must support 16-256, 22-sector mode and 18-256, 20-sector mode packs. For these modes, these systems will change to the appropriate mode and process the data as required. For any additional modes, DECsystem-20 and DECsystem-10 Product Management must decide whether to support them. The decision must be clearly documented in the project plan and the product specification for the software product in question. Support of multiple modes for all other systems is a product management decision. This decision must also be clearly documented in the project plan and the product specification for the software product in question.

4.0 DEFINITION OF THE STANDARD

4.1 Standardized Blocks

Block 1 is the standard identification block for disk systems supporting 16- and/or 18-bit disk-pack word lengths of 256 disk-pack words per logical block.

4.1.1 Nonstandardized Areas of Block 1 - Words 0 through 353(8) and Words 376(8) and 377(8) are reserved for software system specific information. This standard does not address the content of this area of Block 1.

4.1.2 Standardized Areas of Block 1 - This area starts at Word 354(8) and contains the following information stored in 7-bit ASCII code in the words listed below. Any character from 040(8) to 176(8) inclusive may be included in the data.

Pac[redacted]-361) This 12(10)-character field contains the

physical pack identity. The characters are stored in 7-bit ASCII bytes.

Owner Name (362-367) This 12(10)-character field contains the name of the owner of the pack stored in 7-bit ASCII bytes. For example, "Don Lewine" or "CS/2 45". This field may be spaces or "system" in systems where no owner is known.

Format Type (370-375) This 12(10)-character field stores the format in which the operating system wrote the pack. The information is stored in an ASCII string and not encoded. The following are some of the possible names:
TOPS-10
TOPS-20
FILES11
RT-11
DOS-11
RSTS
SCRATCH

The exact 12-character string that a file system writes into the home block is a part of the specification of that file system and need not be registered here.

For disk systems supporting other than 16 and/or 18 bits per disk pack word, 256 disk pack words per logical block, refer to Section 5.0 (Disk System Specific).

4.2 Error Messages

Any program that accesses a removable disk pack must generate an error message under the following conditions:

4.2.1 Incompatible Modes - When a user attempts to read a pack in a mode other than the one in which it was written, the hardware will generate a mode error indication. The software system uses this mode error indication to determine what course of action to take next.

4.2.1.1 Software Systems Supporting Single Modes Only - Software systems supporting single modes only will generate the following error message upon detection of the mode error indication:

"?The pack on drive ---- is not written in a mode supported by this software system."

where: (----) equals drive number

4.2.1.2 Software Systems Supporting Multiple Modes - Software systems

supporting multiple modes will implement the following algorithms upon detection of the mode error indication (Figure 4-1).

a. Read the pack.

b. If a mode error is detected and untried common modes remain, go to d.

c. If a mode error is detected and no untried common modes remain, go to e.

d. Change modes to the next common mode and go to a.

e. Generate mode error message and exit.

The mode error message is formatted as follows:

"?The pack on drive ---- is not written in a mode supported by both the software and the disk hardware."

where ---- = the disk drive number

4.2.2 Incompatible File Formats - When the file structure of data written on a disk pack is not supported by the software accessing the disk pack, an error defined as a "format error" is said to exist.

Under this condition an error message must be issued containing all of the following information:

a. Pack identification,
b. Owner name,
c. Format type,
d. Name of operating system

The recommended error message format is

"?Pack Foo owned by PDP-11 SQE is in DOS-11 format and cannot be used on RT-11."

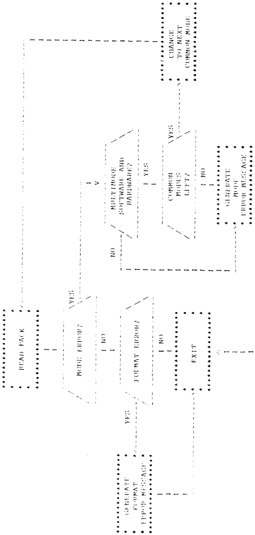


FIGURE 4 1
MODE AND FORMAT ERROR
MESSAGE GENERATION

4.3 Disk System Design Requirements

4.3.1 If a software system is to know that a pack is written in a mode other than one it supports, it must be so notified by the hardware. This standard requires that all future removable disk-pack disk systems provide means for notifying the operating system via an error indication whenever a user attempts to read a pack in a mode other than the one in which it is written. Modes other than ones defined in this standard may not be identified and the results are unpredictable.

4.3.2 In order for software systems to support the reading of multiple modes, the disk subsystem must provide the software with the facilities to change modes purely by software commands. Future removable disk-pack disk systems must address this issue during the product development process. The modes the product can select and the method by which these modes are invoked must be determined and documented in the product design specification. Providing this optional feature is a decision to be made by Disk Engineering and Product Management.

4.4 Diagnostics

Diagnostics must verify that they are using a scratch pack or ask the operator to allow the data on the pack to be destroyed. A scratch pack is one with "SCRATCH" written in the field "FORMAT TYPE." If the pack is not a SCRATCH pack, the following question must be asked by diagnostics: "IS IT OK TO WRITE ON PACK (PACK ID) MOUNTED ON DRIVE ? THE OWNER IS (OWNER NAME) AND THE PACK IS FORMATTED IN (FORMAT NAME): "IS IT OK TO WRITE ON THIS PACK?"
In no case should a diagnostic write on a nonscratch pack without asking.

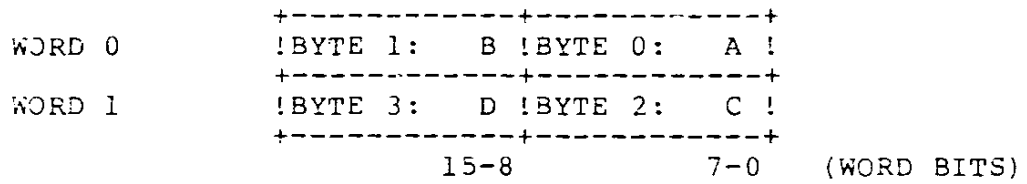
The error message requirements of 4.2 are also applicable to diagnostics.

4.5 PDP-11 Specific

Disk Systems with 16- and 18-Bit Words.

If the ASCII string ABCD is written on the disk, A is in word 0, Bits 0 through 7, B is in Word 0, Bits 8 through 15, C is in Word 1, Bits 0 through 7, and D is in Word 1, Bits 8 through 15. If this string is read into PDP-11 memory, the characters appear in the order ABCD.

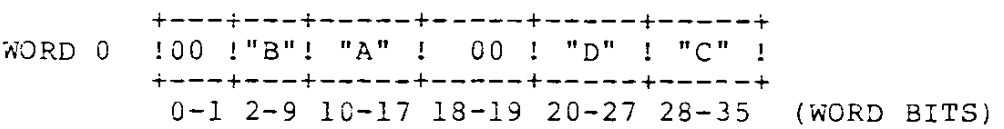
The string in PDP-11 address space looks like the following



4.6 DECsystem-10 and DECsystem-20 Specific

4.6.1 Disk Systems with 16- and 18-Bit Words - If the ASCII string defined in Section 4.5 is read into DECsystem-10 or DECsystem-20 memory, the characters appear in the order BADC. Bit positions 0, 1, 18, and 19 are not used.

The string in DECsystem-10 or DECsystem-20 address space looks like the following:



4.6.2 BLOCK 10 - DECsystem-10 and DECsystem-20 utilize block 10 (8) as a duplicate of Block 1. This feature provides a backup volume identification block should Block 1 become unusable. The utilization of block 10 (8) as a duplicate of block 1 is a requirement of this standard and is not optional.

5.0 DISK SYSTEM SPECIFIC

For disk systems with other than 16 or 18 bits per disk word and/or sectors with more or less than 256 words, the structure and location of the logical volume identification block(s) may or may not change from that specified in Sections 4.1.1 and 4.1.2. For this reason, this section defines how disk systems that do not have 16- or 18-bit word lengths and/or 256 words per sector must support this standard.

5.1 RL01 Disk System

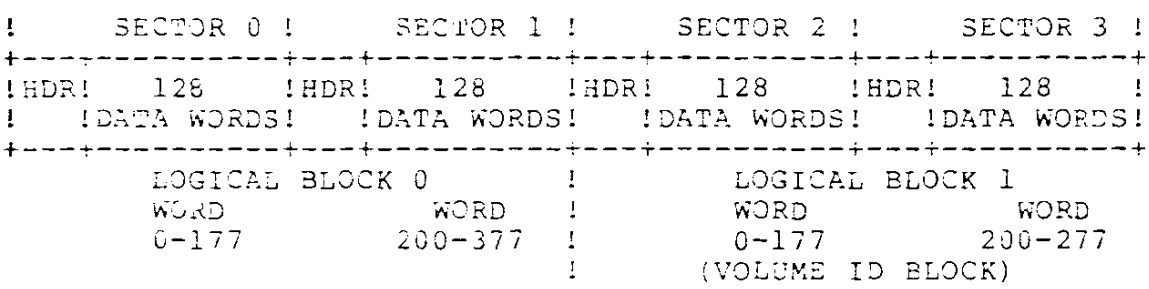
The RL01 disk system differs from disk systems discussed previously in this standard because it has only 128 words per sector. The word length is 16 bits. This standard requires that the volume identification block(s) contain at least 256 words.

5.1.1 RL01 Standard Volume Identification Block

The RL01 standard identification block is logical block 1. It is the same as specified in Section 4.1. This standard, as written, applies to the RL01 disk system.

5.1.2 RL01 Blocks and Sectors

On an RL01 a, logical block is comprised of two RL01 sectors (128 16-bit data words each). A sector and logical block appear on an RL01 pack as follows:



Software systems that must deviate from supporting or implementing the 256-word block format for the volume identification block must indicate a nonconformance to this standard in the project plan for the product as well as in the product design specification. Deviation from support of the 256-word block format for the volume identification block is a Product Management decision and must be documented during the product development phase.

[End of Standard]