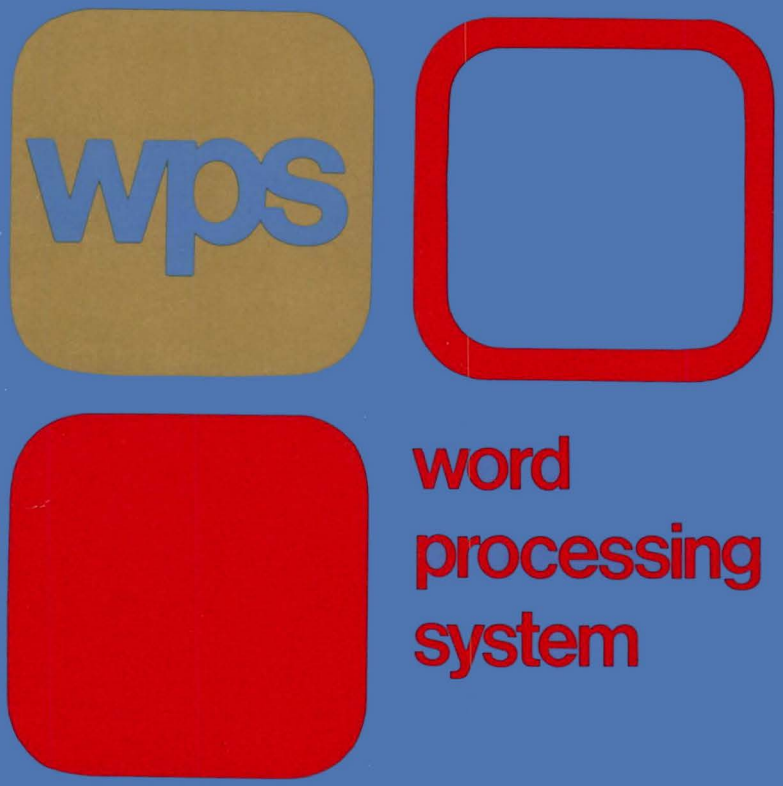


word
processing
system

Interface Handbook

digital

AA-J356A-TK

Interface Handbook

DIGITAL EQUIPMENT CORPORATION
Maynard, Massachusetts

AA–J356A–TK
First Printing, August 1980

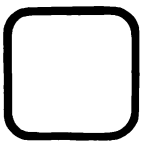
The postage prepaid Reader's Comments Form on the last page of this document requests the user's critical evaluation to assist us in preparing future documentation.

The following are trademarks of Digital Equipment Corporation:

COMPUTER LABS	DIBOL	LAB–8
DDT	DIGITAL	MASSBUS
DEC	FLIP CHIP	PDP
DECtape	FOCAL	TYPESET–8
DECUS	INDAC	UNIBUS

To order additional copies of this document, contact the Software Distribution Center, Digital Equipment Corporation, Maynard, Massachusetts 01754

Preface



PRODUCT NAME: 78 and 80 Series (Version 3.4)
200 Series (Version 4.3)
WPS–11M (Version 3.3)

This manual contains information on techniques WPS uses to store and communicate documents. It is addressed to programmers who want to write programs to interface WPS units or WPS storage media to other systems.

DIGITAL presents this information as a guide to existing versions of WPS software, but reserves the right to change any of the formats described.

Character codes given in this manual are written in *octal* (base 8) notation.

Contents

Format of Stored Documents

This chapter discusses the codes WPS uses to store text, invisible characters, and rulers in a document. The techniques differ among WPS models. Chapters 2 through 4 show how particular WPS models store those codes on their disks and diskettes.

WPS–8 and WPS–11 Formats

WPS stores document text in two different ways:

- *WPS–8 format* is used by the 78, 80, and 200.
- *WPS–11 format* is used by WPS–11M to store documents, and by WPS models using AX, DX, or DX Products to communicate with one another.

The DIGITAL PDP–8 computer, on which the 78, 80, and 200 Series are based, uses a 12–bit word. The 6–bit WPS–8 storage technique lets WPS fit two characters in each PDP–8 word. This technique is comparable to that of COS–310, a DIGITAL operating system for the PDP–8.

The 6–bit WPS–8 code allows 64 possible characters. This is not enough to represent all the displayable characters. Therefore, capital letters are not assigned codes. Instead, two codes are assigned to mean *shift* and *unshift*. If *shift* occurs in a document, then any letter codes that follow refer to the capital letters. Consequently, any program reading a document in WPS–8 format must remember whether it read *shift* or *unshift* most recently.

WPS–11 format is based on 7–bit characters because the DIGITAL PDP–11 computer and all WPS communication lines handle units of 8 bits. (The high-order bit is typically unused.) The WPS–11 storage technique resembles the ASCII character code.

Modes

Special rules apply to a *mode* condition. In *shift mode* every letter code stands for a capital letter instead of a lowercase letter. One of the 64 WPS–8 codes *enters shift mode*; another *exits shift mode.*

The 7–bit WPS–11 format allows 128 possible characters. Since this is enough to give capital and lowercase letters separate codes, WPS–11 format does not have a shift mode.

However, more than 128 different characters can occur in a document. For instance, any character can be underlined. Instead of having a separate code to represent every underlined character, WPS–8 and WPS–11 use *underline mode.* A special code enters this mode. A program reading a document must underline every character it reads until it finds the code to exit underline mode. There are a total of seven modes which modify the meaning of codes or attach special attributes to characters.

WPS–8 Code (Sixbit)

The WPS–8 sixbit codes are shown in the following table. Except for code 74 to code 77, they are the same as the COS–310 codes.

Code	Char.	Code	Char.	Code	Char.	Code	Char.
00	null	20	/	40	?	60	O o
01	space	21	0	41	@	61	P p
02	!	22	1	42	A a	62	Q q
03	"	23	2	43	B b	63	R r
04	#	24	3	44	C c	64	S s
05	$	25	4	45	D d	65	T t
06	%	26	5	46	E e	66	U u
07	&	27	6	47	F f	67	V v
10	'	30	7	50	G g	70	W w
11	(	31	8	51	H h	71	X x
12	)	32	9	52	I i	72	Y y
13	*	33	:	53	J j	73	Z z
14	+	34	;	54	K k	74	shift
15	,	35	<	55	L l	75	reserved
16	–	36	=	56	M m	76	unshift
17	.	37	>	57	N n	77	cmd

Code 00 has no meaning. WPS uses 00 as a filler when not enough document text exists to take up the allotted space.

Codes 41 through 73 have two possible meanings. The character on the left is the *shifted* value of the code; the character on the right is its *unshifted* value. Code 74 enters *shift mode* and indicates that subsequent codes take their shifted values. Code 76 exits shift mode and indicates that subsequent codes take their unshifted values.

For instance, if the text "Hello, John" appeared in a document, WPS would store it as:

shift H unshift ELLO, space shift J unshift OHN

Code 75 is reserved for future use by DIGITAL.

WPS–11 Code (Modified ASCII)

The following table lists the WPS–11 code. Except for code 173 to code 176, the WPS –11 codes are the same as ASCII (the American Standard Code for Information Interchange).

WPS–11 does not use codes 000 through 037 and 177. These codes are the ASCII control codes. Codes 173 and 174 indicate that the following character has a special meaning. These will be discussed shortly.

Code	Char.	Code	Char.	Code	Char.	Code	Char.	Code	Char.
040	space	063	3	106	F	131	Y	154	l
041	!	064	4	107	G	132	Z	155	m
042	"	065	5	110	H	133	[	156	n
043	#	066	6	111	I	134	\	157	o
044	$	067	7	112	J	135	]	160	p
045	%	070	8	113	K	136	^	161	q
046	&	071	9	114	L	137	—	162	r
047	'	072	:	115	M	140	`	163	s
050	(	073	;	116	N	141	a	164	t
051	)	074	<	117	O	142	b	165	u
052	*	075	=	120	P	143	c	166	v
053	+	076	>	121	Q	144	d	167	w
054	,	077	?	122	R	145	e	170	x
055	–	100	@	123	S	146	f	171	y
056	.	101	A	124	T	147	g	172	z
057	/	102	B	125	U	150	h	173	mode
060	0	103	C	126	V	151	i	174	vb
061	1	104	D	127	W	152	j	175	reserved
062	2	105	E	130	X	153	k	176	reserved

Two-Character Sequences

Code 77 (*cmd*) in WPS–8, and codes 173 (*mode*) and 174 (*vb*) in WPS–11, indicate that the following character has a special meaning.

SPECIAL CHARACTERS		
WPS–8	**WPS–11**	**Meaning**
cmd %	vb I	tab
cmd *	vb J	end of line
cmd +	vb L	end of page
cmd 6	vb G	line modified
cmd 7	vb N	start of ruler
cmd 8	vb O	end of ruler

MODE CHANGES		
WPS–8	**WPS–11**	**Meaning**
cmd !	mode#	enter boldface
cmd "	mode"	exit boldface
cmd #	mode%	enter underline
cmd $	mode$	exit underline
cmd &	vb H	enter composite
cmd '	vb M	exit composite
cmd ,	mode)	enter superscript
cmd –	mode(	exit superscript
cmd .	mode+	enter subscript
cmd /	mode*	exit subscript
cmd (	mode'	enter auxiliary
cmd)	mode&	exit auxiliary

VISIBLE CHARACTERS WHICH DON'T HAVE SINGLE CODES		
WPS–8	**WPS–11**	**Meaning**
cmd 1	vb ;	{ (on WPS–8, | if shifted)
cmd 2	vb <	| (on WPS–8, \ if shifted)
cmd 3	vb =	} (on WPS–8,] if shifted)
cmd 4	vb >	~ (on WPS–8, ^ if shifted)
cmd 5		_ (underline has a single code on WPS–11)

Twelve of the codes above are commands to either enter or exit one of six modes. The codes for bold mode, underline mode, superscript mode, subscript mode, and composite mode surround characters which have been given special attributes by use of the keys BOLD, UNDER LINE, Gold SUPER SCRIPT, Gold SUB SCRIPT, and Gold DEAD KEY, respectively.

WPS uses auxiliary mode together with other codes to indicate invisible characters.

Summary of Modes

There are seven *modes* which affect the meaning of codes or assign special attributes to characters. Each mode has one code or sequence which *enters* it and another which *exits* it. The codes between these two points are the *range* over which the mode takes effect. At the start of any document, all modes are exited.

Mode	WPS–8 Command Enter	WPS–8 Command Exit	WPS–11 Command Enter	WPS–11 Command Exit	What Happens with Mode in Effect
auxiliary	cmd (	cmd)	mode'	mode&	Many characters take special meanings
bold	cmd !	cmd "	mode#	mode"	Characters are boldfaced
composite	cmd &	ncmd '	vb H	vb M	All characters in the range are dead keys, printed in the same spot to form a composite character
shift	shift	unshift	...	...	Codes 41 through 73, cmd1, cmd2, cmd3, and cmd4 take their shifted values
subscript[1]	cmd .	cmd /	mode +	mode*	Characters are subscripts
superscript[1]	cmd,	cmd –	mode)	mode(	Characters are superscripts
underline[1]	cmd #	cmd $	mode%	mode$	Characters are underlined

[1] These three modes are mutually exclusive: before entering one of these modes, a document will contain commands needed to exit the other modes.

Invisible Characters

WPS uses auxiliary mode to change the meaning of other characters and thereby represent invisible characters and other markers that can occur in a document.

The character	is represented by the code for	with the following modes in effect
Hard return	end of line	none
Word wrap return at end of word	end of line	auxiliary
Word wrap return at hyphenation point	end of line	auxiliary and underline
Paragraph marker	end of line	superscript
Centering mark	end of line	subscript
*	end of line	auxiliary and subscript
Space	space	none
Soft space	space	auxiliary
*	space	auxiliary and subscript
*	space	auxiliary and superscript
NEW PAGE mark	end of page	none
PAGE MARKER	end of page	auxiliary
Start Print Ctrl	end of page	superscript
End Print Control	end of page	subscript
Nonbreaking hyph.	hyphen	none
Breaking hyphen	hyphen	auxiliary
Invisible hyphen	previous character	auxiliary
Word wrap needed to end of paragraph	line modified	not bold
Word wrap needed to next ruler	line modified	bold

* WPS uses these combinations as internal markers. They should never appear in a filed document.

Rulers

At the top of a document and at any point where the ruler settings change, WPS stores a sequence that shows the ruler settings before and after the change. The sequence is recognizable from either direction:

(a)	(b)	(c)	(d)	(e)
start ruler code	ruler settings in effect for preceding text	«	ruler settings in effect for following text	end ruler code

In the ruler at the top of the document, area (b) in the preceding diagram is omitted since no "preceding text" exists.

Areas (b) and (d) both consist of a list of ruler settings. There are always at least two settings, representing the left and right margins. WPS never stores a ruler where (b) and (d) are identical.

Each setting consists of a code for a column number, followed by a code for the setting which takes effect at that column.

Column Numbers

WPS uses one or two characters to represent a column number. It chooses from the numerals, :, ;, <, =, >, and ?. Each character supplies four bits of the column number, producing a hexadecimal code for all 158 column numbers. The following table shows the characters for each column from 1 to 158. The actual codes differ between WPS–8 and WPS–11.

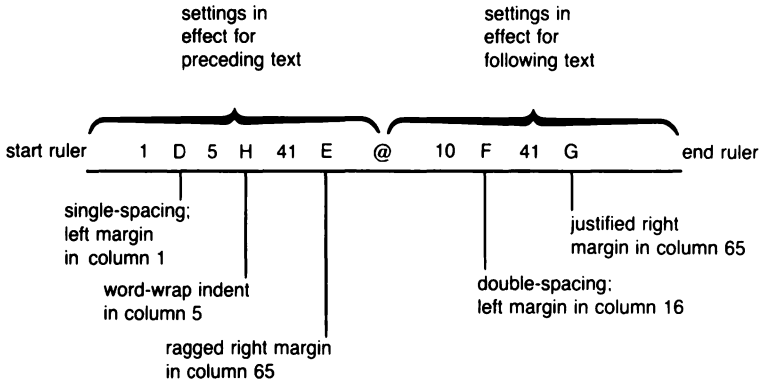
	0	1	2	3	4	5	6	7	8	9
1– 9		1	2	3	4	5	6	7	8	9
10– 19	. :	;	<	=	>	?	10	11	12	13
20– 29	14	15	16	17	18	19	1:	1;	1<	1=
30– 39	1>	1?	20	21	22	23	24	25	26	27
40– 49	28	29	2:	2;	2<	2=	2>	2?	30	31
50– 59	32	33	34	35	36	37	38	39	3:	3;
60– 69	3<	3=	3>	3?	40	41	42	43	44	45
70– 79	46	47	48	49	4:	4;	4<	4=	4>	4?
80– 89	50	51	52	53	54	55	56	57	58	59
90– 99	5:	5;	5<	5=	5>	5?	60	61	62	63
100–109	64	65	66	67	68	69	6:	6;	6<	6=
110–119	6>	6?	70	71	72	73	74	75	76	77
120–129	78	79	7:	7;	7<	7=	7>	7?	80	81
130–139	82	83	84	85	86	87	88	89	8:	8;
140–149	8<	8=	8>	8?	90	91	92	93	94	95
150–158	96	97	98	99	9:	9;	9<	9=	9>	

Ruler Settings

WPS uses other characters to represent the various settings that may appear in a ruler. They are not the same characters seen in the ruler. The actual codes for the characters differ from WPS–8 to WPS–11.

WPS–8	WPS–11	Char.	Meaning	Ruler Symbol Shown on Screen
42	101	A	decimal tab stop	
43	102	B	right-justified tab stop	>
44	103	C	normal (left-justified) tab	T
45	104	D	left margin, single-spaced	L
46	105	E	right margin, ragged	R
47	106	F	left margin, double-spaced	D
50	107	G	right margin, justified	J
51	110	H	word wrap indent	W
52	111	I	paragraph indent	P
53	112	J	centering point	C
54	113	K	left margin, space-and-a-half	N
55	114	L	hyphenation zone	H
56	115	M	left margin, half-spaced	F

Example Spaces are for illustration only.



Formatting Ruler settings tell WPS what shape the text should take. As a document is typed, WPS inserts *word wrap returns* to make the text the desired shape. If the typist uses the TAB key, then as well as inserting a tab character in the document to mark the space relationship between the text, WPS inserts *soft spaces* to align the text as shown in the ruler. WPS also inserts soft spaces after column one of a line with a paragraph, word wrap indent, or indented left margin.

When a typist edits a document, WPS removes word wrap returns and soft spaces from the region of text whose shape may change. But WPS manuals teach users to move the cursor through this area so that WPS reinserts them where they now belong. Assuming users follow this guideline, programs reading WPS documents can use soft spaces and word wrap returns as indicators of the desired shape of the text; they need not interpret rulers nor contain text-wrapping routines.

When a typist has made changes to text but has failed to rewrap it, WPS leaves one of two markers in the text. The different markers show how much text ought to be rewrapped:

1 When a typist makes any changes to text, the shape of text in the remainder of the paragraph may change. WPS places a *line modified* code in the text with bold mode disabled.

2 When a typist changes a ruler, the shape of text up to the next ruler may change. WPS places a *line modified* code in the text with bold mode enabled.

If users remember to rewrap any region of text they have changed, the *line modified* markers will not appear in the document.

Encoding of Binary Data

Sometimes, especially in communications between WPS units, numbers occur as well as text. For instance, when a WPS unit sends a message to another WPS unit, it attaches a checksum (the sum of all characters sent so far) so the receiver can detect errors. WPS uses a special technique to represent a number using one or two text characters.

Sixbit Encoding

WPS–8 encodes numbers from octal 00 to 77 (up to decimal 63) by using a single 6–bit character.

For numbers which can range up to octal 7777 (decimal 4,095), WPS–8 uses two successive sixbit characters. The high 6 bits (left two octal digits) appear in the first character; the low 6 bits (right two octal digits) appear in the second.

Encoding in WPS–11 and Communications

When sending binary data between WPS units, WPS avoids using codes which are ASCII control codes (codes from 000 through 037). The WPS–11 format represents binary numbers using the same characters as WPS–8. For instance, the number WPS–8 represents as "G" is represented as "G" in WPS–11 as well. But the numeric code for "G" is different in the two formats.

- For a 6–bit number, obtain the WPS–11 code by adding octal 37 to the number. But if the number is 0, the WPS–11 code is 137.
- Decode a WPS–11 code by subtracting 37 and masking out bit 6 (ANDing with octal 77).

For 12–bit numbers, WPS–11 uses two characters. The first character represents the high 6 bits; the second represents the low 6 bits. Both characters are encoded as described above.

Algorithms

A BASIC–PLUS program could convert two characters in WPS–11 format (call them A$ and B$) to a 12–bit number (X%) by using the following routine. You would use this routine when reading data from the header of a WPS–11M file or from a DX communication line.

```
A% = (ASC(A$)–31) AND 63        !31 is 37 octal; 63 is 77 octal.
B% = (ASC(B$)–31) AND 63        !A% and B% are A$ and B$ decoded
X% = A%*64 + B%
```

The following routine converts a 12–bit number (X%) back to characters A$ and B$:

```
A% = INT(X$/64)                          !A% gets high 6 bits
B% = X$ AND 63                           !B% gets low 6 bits
IF A% = 0 A$ = '_' ELSE A$ = CHR$(A% + 31)
IF B% = 0 B$ = '_' ELSE B$ = CHR$(B% + 31)
```

WPS–8 Diskette Format 2

This chapter describes how WPS stores data on diskettes.

There are two types of WPS diskettes. A Document Diskette contains WPS–8 document text (see Chapter 1) for up to 200 documents. It also contains pointers and other information WPS needs to locate and use the document text. All WPS models can read and write a Document Diskette compatibly.

A System Diskette has the same general format, but also contains WPS software. Over half its blocks are reserved for software and are unavailable for document storage. A System Diskette can be used in place of a Document Diskette on all WPS models; System Diskettes contain information distinguishing them from Document Diskettes. The 78 and 80 Series (including the Models 211 and 212) require the use of a System Diskette.

Diskette Organization

WPS diskettes are single-sided, single-density, soft-sectored diskettes. The diskette spins in the diskette drive. The drive's read/write head can move to one of 77 linear positions. In each position the head can read or write text on one side of the diskette in a circular region called a track. The 77 tracks are concentric circles. The outermost track is number 0; the innermost track is number 76.

Each track is divided into 26 sectors numbered from 1 to 26. A head can read or write a sector at a certain time during each rotation of the diskette.

Each sector contains 128 8–bit bytes (characters) of information, plus other control information. The physical layout and format of sectors is identical to that of the IBM 3770 diskette.

Blocks

Recall that the DIGITAL PDP–8 computer deals in units of 12 bits. WPS organizes the diskette's sectors into blocks which contain 256 12–bit words. Three sectors form one block. WPS always reads and writes information one block at a time.

A WPS block is the same as a block under the DIGITAL COS–310 operating system. The first sector contains the high 4 bits of each of the 256 words in the block, in sequence. (In this sector, the first byte's high 4 bits are the high 4 bits of the first word in the block.

The first byte's low 4 bits are for the second word in the block. The second byte's high 4 bits are for the third word, and so on.) The second sector contains the low-order 8 bits of the first 128 words, in sequence. The third sector contains the low-order 8 bits of the second 128 words, in sequence.

Sectors are assigned to blocks using an interleave factor of 3. This gives WPS enough time to handle one sector before the next sector rotates into position. Here is the pattern by which sectors are arranged into blocks. In the following table, "3–5" means track 3, sector 5.

Block	is composed of the three sectors		
0	1–1	1–4	1–7
1	1–10	1–13	1–16
2	1–19	1–22	1–25
3	1–2	1–5	1–8
4	1–11	1–14	1–17
5	1–20	1–23	1–26
6	1–3	1–6	1–9
7	1–12	1–15	1–18
8	1–21	1–24	2–1
9	2–4	2–7	2–10

WPS does not use the outermost track (track 0) or the innermost three tracks (tracks 74, 75, and 76).

Types of Blocks

Block 0 is the *bootstrap block*. The information in this block is a program the PDP–8 computer reads when you press the START button. In turn, this program reads the WPS software from diskette. The first 8 words of block 0 are not part of this program; they are free to contain labeling information for the diskette.

Block 1 is unused by WPS. *Block 2* is the *home block*. The remaining blocks are available for storage of documents or software, except for *block 255*, which is the *allocation block*.

The words in every block are numbered from 0 to 255. Word 0 contains –255, the negative number of words which follow. Word 1 indicates the type of block. The information in the other 254 words varies depending on the type of block.

The following descriptions show what information is contained in each word of the various types of blocks. DIGITAL reserves the bit-positions and words not discussed for future use. (Remember that on the PDP–8, the high-order bit is labeled bit 0 and the low-order bit is bit 11.)

Home Block

The home block contains information which pertains to the entire diskette. It also shows WPS where to look for various documents.

Word	Meaning
1	Bits 2–5 contain the version number of the WPS storage technique (currently 1). Bits 6–8 contain 3, identifying this as the home block.
2–4	The diskette name typed when using a Maintenance Command to initialize the diskette: six 6–bit characters.
5	A diskette identification number, automatically assigned when you initialize the diskette.
6	The block number of the allocation block: It is always 255.
7–8	The date the diskette was initialized. Bits 0–5 of word 7 contain the day; bits 6–11 contain the month; word 8 contains the year minus 1900.
9	The negative number of potential documents which is always 200.
10–209	Each word contains a block number, pointing to the header block for document numbers 1 through 200, respectively. If a word contains 0, that document does not exist.

Allocation Block

The allocation block lets WPS determine which blocks are in use and which are available to store new text. A word in the home block shows which block serves as the allocation block. (This is currently block 255.)

Word	Meaning
1	Bits 6–8 contain 4, identifying it as the allocation block.
2	The total number of usable blocks on the diskette: 632.
3	The number of blocks not currently used.
4	If x is the number of words in the allocation table (79), this word contains –x–1: –80.
5–83	The allocation table.

In each word of the allocation table, the low 8 bits (bits 4–11) are used. The allocation table therefore contains 632 bits. Each bit stands for one of the diskette's 632 usable blocks and indicates whether or not that block is available for storage. A 1 bit means it is; a 0 bit means the block is already in use.

The first word of the allocation table refers to blocks 0 through 7. The second word refers to blocks 8 through 15. In general, bit j (4–11) of word i (5–83) of the allocation block refers to block number:

$$8(i–5)+(j–4)$$

Blocks 0 (the bootstrap block), 1, 2 (the home block), and 255 (the allocation block) are always marked as "in use." WPS is thus prevented from storing document text in these blocks. On System Diskettes, the blocks containing WPS software are likewise marked as "in use."

Document Header Block

The document header block contains settings which pertain to the entire document, such as Print Menu settings.

The blocks which hold the document text are not arranged in any particular order. As a result, you can add text at any point in a document without having to shift the position of unaffected text. The header block shows WPS which blocks contain the document text.

To find the location of the header block for a particular document, WPS looks in the home block.

Word	Meaning
1	Bit 1 is set if someone is currently editing the document. Bits 2–5 contain the version number of the WPS storage technique (currently 1). Bits 6–8 contain 1, identifying this as a header block.
2–3	One or two additional blocks may be needed to hold all the header information. If so, these words contain the block numbers of those additional blocks. These words contain 0 if no additional blocks are used.
4	Always contains 40.
5	Number of blocks allocated to this file, not including the header block(s).
6–7	Date the document was created. Bits 0–5 of word 6 contain the day; bits 6–11 contain the month; word 7 contains the last two digits of the year.
8–9	Date the document was last edited, in the same format as words 6 and 7.
10	Version number; the number of times the document has been edited.
11	Document number.
12	The time the document was last edited. Bits 0–5 contain the hour; bits 6–11 contain the minute.
13	Editor Menu setting CT for this document.
14	Elapsed time, in minutes, of the last edit.
15	Total elapsed time while this document has been edited.

Words 19 through 43 contain settings from the document's Print Menu.

Word	Meaning
19	Contains 0 if the Print Menu has never been used. If so, the rest of this section is undefined.
20	CP, number of copies
21	PM, print margin
22	EX, extra half-lines
23	TM, top margin
24	BM, bottom margin
25	PS, page size
26	PI, pitch
27	FR, from page number
28	TO, to page number
29	IP, initial page number
31	AP (0 = AP NO; 1 = AP YES)
34	SE (0 = SE NO; 1 = SE YES; 2 = SE FIRST)
36	DA (0 = DA NORMAL; 1 = DA DARK)
37	TW (0 = TW NO; 1 = TW YES)
38	DD (0 = DD LQP; 1 = DD DP; 2 = DD HOST)
39	CM
40	R1
41	R2

Words 44 through 255 are the block pointer list. Word 44 contains 0 to mark the start of this list. Subsequent words contain the addresses of blocks on the diskette. These are the blocks, in this order, which contain the text of the document. The first word past 44 which also contains 0 marks the end of the block pointer list.

Header Extension Blocks

If the document is larger than 211 blocks, all of the words 45 through 255 will be required; none will contain zero. In this case, the block pointer list will continue in one or two *header extension blocks*. Their addresses are contained in words 2 and 3 of the original header block.

Words 0 and 1 of each header extension block are identical to words 0 and 1 of the original header block. The remainder of the block is a continuation of the block pointer list. A 0 anywhere in these words marks the end of the list.

Document Text Blocks

The document text blocks are referred to in the block pointer list. Every document text block in use should be named in the header block(s) of exactly one document.

Word 1 of a document text block contains 0 to identify it as such. Words 2 through 255 contain document text in the WPS–8 format (see Chapter 1).

Named Documents

The format of WPS–8 diskettes refers to documents only by number (1 through 200). WPS also allows a typist to select a document by name. As *document number 1* on any diskette, WPS maintains an *index document* which associates document names with numbers. If a legal index document exists, then WPS allows a typist to use a document name. WPS uses the index document to convert the name to a number and proceeds as though the typist had typed the number. The format of the index document is discussed in Chapter 4 of the *WPS Reference Manual*, AA–J043A–TK.

WS200 Disk Format

This chapter describes how the 200 Series WPS units store data on the RL01 hard disk.

Just as there are two types of diskettes, there are two types of disks. A Document Disk contains 15 *sections*. Each section contains 632 blocks, the same number as a WPS diskette. The layout of a section is identical to the layout of a diskette presented in Chapter 2. When 200 Series users assign area numbers to certain sections, WPS lets them use those sections as though they were diskettes.

A System Disk contains only 12 sections. The remaining space on the disk contains:

- *WPS software* which governs the behavior of the 200 Series unit
- Tables in which WPS keeps track of what each terminal, each user, and each printer is doing
- Tables where WPS keeps other information pertaining to the entire unit

This information does not affect the storage of documents and is subject to change in subsequent versions of WPS. Therefore, this manual does not discuss in detail the format of these portions of disk.

Disk Organization

The RL01 disk is a removable, top-loading, 5440–type disk cartridge. Its single platter is a double-sided, single-density, hard-sectored medium. Hard-sectored means that, as the disk spins in the diskette drive, the drive senses the passage of grooves in the disk. This lets the drive sense the passage of the 40 sectors.

The drive's read/write arm can move to one of 256 linear positions. In each position the drive can read or write text in a circular region called a cylinder. Cylinder 0 is the outermost one; cylinder 255 is the one closest to the center of the disk. The drive has two heads; in each cylinder position "head 0" contacts the top side of the disk platter, while "head 1" contacts the bottom side.

Each cylinder has 40 sectors of information corresponding to the 40 grooves in the disk. If a head is in a particular cylinder position, it can read or write each of the 40 sectors at a certain time during each rotation of the disk. Sectors in a cylinder are numbered from 0 to 39 in the order in which the head will encounter them. But the sector numbering on the next cylinder is shifted. For instance, sector 0 on cylinder 18 does not line up with sector 0 on cylinder 17; it is almost halfway around the disk to compensate for the time it takes the head to move from cylinder 17 to 18. Sector numbering on the other surface of the platter is also shifted. It's designed this way to compensate for the time needed to move between cylinders.

The *RL01/RL02 User's Guide* (order no. EK–RL012–UG) contains details on the layout of the RL01. It is sufficient for the discussion in this chapter to know that WPS can read or write any sector on the disk by supplying the disk drive with a head number (0 or 1), a cylinder number (from 0 through 255) and a sector number (from 0 through 39).

WPS can write sectors in *12–bit mode* or *8–bit mode*. In 12–bit mode, WPS stores 128 12–bit words in a sector. In 8–bit mode, a sector contains 256 8–bit characters (bytes). For different information, WPS uses both storage techniques.

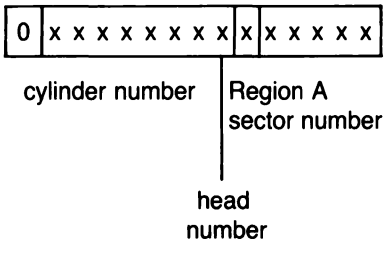
Sector Numbering

WPS assigns a unique serial number to each sector on the disk. This lets WPS deal in terms of the "next" or "previous" sector. It assigns serial numbers so that adjacent sectors are usually on the same or nearby cylinders. The numbering scheme is optimized for the limited arithmetic ability of the PDP–8 computer. The disk is divided into two *regions* where the numbering scheme works differently.

Sector numbers are 15–bit binary numbers. (This manual shows them as 5–digit octal numbers.) They go from 00000 to 47777. Sector numbers from 00000 through 37777 are *Region A*. Region A comprises 32 of the 40 sectors on each cylinder. Sector numbers 40000 through 47777 are *Region B*, which comprises the other 8 sectors on each cylinder.

Region A

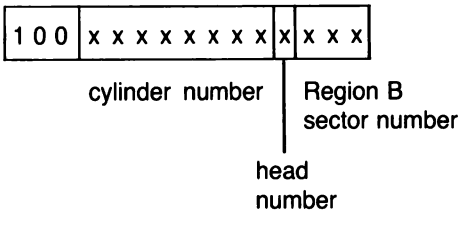
If a sector number is in the range from 00000 through 37777—that is, if the high bit is zero, the other 14 bits indicate what head, cylinder, and physical sector number to use to access that sector:



This scheme yields a cylinder number from 0 through 255 and a head number which is 0 or 1. These are exactly the cylinder and head numbers discussed earlier. The relationship between "Region A sector number" in the preceding figure and the sector numbers discussed earlier is as follows:

Region A Sector No.	RL01 Sector	Region A Sector No.	RL01 Sector
octal 00	0	octal 20	32
octal 01	2	octal 21	34
octal 02	4	octal 22	36
octal 03	6	octal 23	38
octal 04	8	octal 24	1
octal 05	10	octal 25	3
octal 06	12	octal 26	5
octal 07	14	octal 27	7
octal 10	16	octal 30	9
octal 11	18	octal 31	11
octal 12	20	octal 32	13
octal 13	22	octal 33	15
octal 14	24	octal 34	17
octal 15	26	octal 35	19
octal 16	28	octal 36	21
octal 17	30	octal 37	23

Region B If the sector number is 40000 or above—if its top 3 bits are 0–1–1—then the remaining 12 bits indicate what head, cylinder, and physical sector number to access that sector:



This scheme also yields a cylinder number from 0 through 255 and a head number which is 0 or 1. The relationship between "Region B sector number" in the preceding figure and the sector numbers discussed earlier is shown in the following table.

Region B Sector No.	RL01 Sector
octal 0	25
octal 1	27
octal 2	29
octal 3	31
octal 4	33
octal 5	35
octal 6	37
octal 7	39

As you see, every RL01 sector number has either a Region A sector number or a Region B sector number. Sectors whose RL01 sector number is an odd number 25 or above are in Region B; all other sectors in each cylinder are in Region A.

Interleave

The preceding tables also illustrate that, to read sectors with consecutive serial numbers, WPS does not read sequential RL01 sectors. Instead, it skips one each time. For instance, to read consecutive sectors, WPS might read a cylinder's sector 25, then 27, then 29. By the time WPS has processed the data from sector 25, sector 27 should be about to pass under the disk's heads. Thus WPS spends minimum time waiting for the disk's rotation.

Volume Header

The volume header is written in 8–bit mode. This means it consists of 256 8–bit characters. Copies of the volume header are found in sectors 00001, 00002, and 00003. The volume header contains the following information:

Byte	Meaning
0	Series number of WPS software: always 200.
1	Disk layout version: 2.
2–3	Version number and modification level, respectively, of the WPS software which initialized this disk. For WPS Version 4.3, word 2 contains 4 and word 3 contains 3.
4	Month of initialization.
5	Day of initialization.
6	Last two digits of year of initialization.
7	Mode in which the file structure information is recorded: 12 (12–bit mode).
8	Pack type. 1 means this is a Document Disk; 2 means this is a System Disk.
10–11	Sector number of the section table (see below). Byte 10 contains the low 8 bits; byte 11 contains the high 7 bits.
183–255	Volume name, in ASCII. Terminated by a zero byte. Excluding the zero byte, the name can be up to 72 characters.

Other words contain pointers to internal tables on the disk.

Section Table

The volume header points to the section table which is recorded in 12–bit mode. The section table consists of up to fifteen entries; each entry is two 12–bit words.

The words	contain the sector address of the start of the section
0 and 1	SECT 1
2 and 3	SECT 2
4 and 5	SECT 3
6 and 7	SECT 4
8 and 9	SECT 5
10 and 11	SECT 6
12 and 13	SECT 7
14 and 15	SECT 8
16 and 17	SECT 9
18 and 19	SECT10
20 and 21	SECT11
22 and 23	SECT12
24 and 25	SECT13 (Document Disks only)
26 and 27	SECT14 (Document Disks only)
28 and 29	SECT15 (Document Disks only)

Of the two words, the first word contains the low 12 bits of the sector address. The low 3 bits of the second word are the high 3 bits of the sector address.

Entries in the section table which are unused (because the disk does not have that many sections) are set to octal 7777.

Sections

Each entry in the section table points to the first of 1264 sectors of a section. All these sectors are written in 12–bit mode. Two consecutive sectors, containing a total of 256 12–bit words, make one block. For instance, the first two sectors of a section form block 0 of that section.

There are a total of 632 blocks. Each block has exactly the same format as the corresponding block on a WPS–8 diskette (see Chapter 2).

WPS–11M File Formats

4

This chapter describes how WPS documents are stored on PDP–11 computers using RSX–11M and RSTS.

Filenames

Each WPS document is stored in a different file.[1] If the file is created by WPS–11M, the filename is DOCnnn.W11, where "nnn" is the document number. Document numbers range from 001 through 200. WPS–11M associates an RSX–11M UFD with each area; it stores the file DOCnnn.W11 in the UFD corresponding to the area in which the user created the document.

If the file is created by DX/RSTS or DX/11M, the filename and the UFD or account in which the file is stored are specified by the DX user.

In every UFD used by WPS–11M to store documents, a file named BITMAP.W11 shows the values of "nnn" for which files DOCnnn.W11 exist. BITMAP.W11 also contains some header information from each document, to make the Index operation go faster. The format of BITMAP.W11 is given later in this chapter.

File Organization

The first block of a file (characters 0 through 511) is the *file header.* Remaining blocks contain the document text. Invisible characters, highlighting, and rulers are encoded as described in Chapter 1.

---

[1] IAS and RSX programs accessing document files *must* use the following attributes, described in Appendix A of the *IAS/RSX I/O Operations Reference Manual, AA-2515D-TC:*

F.RTYP = R.FIX
F.RATT = FD.BLK
F.RSIZ = 512.

File Header

The file header contains the same information as the header block of a WPS–8 diskette: the document's print settings, version number, information about when it was created and last edited, as well as other control information.

Each piece of information in the file header is a 12–bit number. Every word from the header block of a WPS–8 diskette is encoded into two successive characters of the WPS–11 file header, using the scheme introduced in Section 1.6.

The following table lists the numeric value of each entry in the header block (after decoding the two characters). Differences from WPS–8 are noted in *italics*. Characters not mentioned are reserved by DIGITAL for future expansion and typically contain zero.

Chars.	Meaning
0–1	Always contains –255 (octal 7401).
2–3	Bits 2–5 contain the version number of the WPS software that created the document. Bits 6–8 contain 1, identifying this as a header block. (*Unlike WPS–8, bit 1 is never set because documents are not edited in place.*)
4–5 6–7	*Both numbers are* always 0 because WPS–11M never needs "*additional header blocks.*"
8–9	Always contains 40.
10–11	Number of blocks allocated to this file.
12–13 14–15	Date the document was created. Character 12 contains the day; character 13 contains the month; characters 14 and 15 contain the last two digits of the year.
16–17 18–19	Date the document was last edited, in the same format as characters 12 through 15.
20–21	Version number; the number of times the document has been edited.
22–23	Document number.
24–25	The time the document was last edited. Character 24 contains the hour; character 25 contains the minute.
26–27	Editor Menu setting CT for this document.
28–29	Elapsed time, in minutes, of the last edit.
30–31	Total elapsed time while this document has been edited.
32–33	*Read access codes, showing whether users may access this file even if it is not in one of their areas. The number is octal 5701 (encoded as 'N ') if nobody but the creator can read it; 5001 ('G ') means anyone in the creator's group can read it; 4601 ('E ') means anyone can read it.*

Characters 38 through 87 contain settings from the document's Print Menu.

Chars.	Meaning
38–39	Contains 0 if the Print Menu has never been used. If so, the rest of this section is undefined.
40–41	CP, number of copies;
42–43	PM, print margin;
44–45	EX, extra half-lines;
46–47	TM, top margin;
48–49	BM, bottom margin;
50–51	PS, page size;
52–53	PI, pitch;
54–55	FR, from page number;
56–57	TO, to page number;
58–59	IP, initial page number;
62–63	AP (0 = AP NO; 1 = AP YES);
68–69	SE (0 = SE NO; 1 = SE YES; 2 = SE FIRST);
72–73	DA (0 = DA NORMAL; m1 = DA DARK);
74–75	TW (0 = TW NO; 1 = TW YES);
76–77	DD (0 = DD LQP; 1 = DD DP; 2 = DD HOST; *3 = DD LP*);
78–79	CM;
80–81	R1;
82–83	R2;

Format of BITMAP.W11

The document BITMAP.W11 shows which of the 200 legal document numbers are in use in the same UFD (area) and contains some header information from each document.

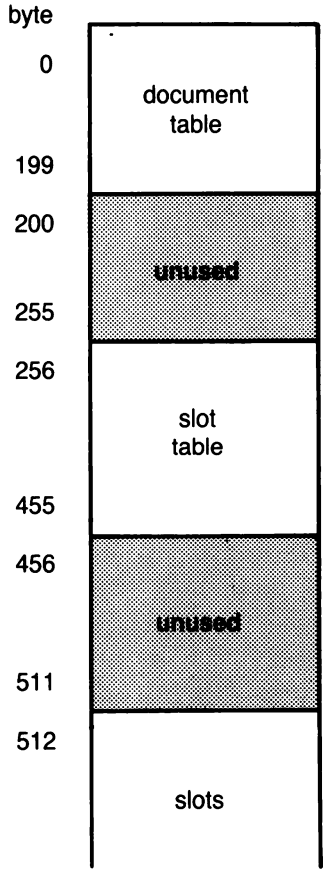
The header information is stored in a "slot." BITMAP.W11 can contain up to 200 slots. Slots, like documents, are numbered from 1 through 200. But the header information from document number 50, for instance, does not have to reside in slot number 50. BITMAP.W11 contains two tables, one which links document numbers to slot numbers, and another which links slot numbers to document numbers. Thus, if the first document created is document number 200, its header information can be stored in slot 1 of BITMAP.W11. At this time, BITMAP.W11 need only be large enough to contain one slot. The file could be lengthened when other slots must be used.

The first block of BITMAP.W11 contains these two tables. The remainder of BITMAP.W11 contains the slots.

The first half of the first block is the *document table.* Only the first 200 bytes are used. If these bytes are numbered 1 through 200, then each byte shows whether a document with that number exists. For instance, if byte 18 is zero, then DOC018.W11 does not exist in the same UFD (area). If byte 18 is nonzero, then its value is the slot number in which that document's header information is stored.

The second half of the first block is the *slot table.* Again, only the first 200 bytes are used. Each byte shows whether a slot in BITMAP.W11 is used. For instance, if byte 41 is zero, then the 41st slot in BITMAP.W11 is unused. If byte 41 is nonzero, then its value is the document number whose header information is stored in the 41st slot.

The slots begin in the second block. Each block is divided into 16 slots. Each slot is 32 characters long and contains characters 0 through 31 of the document file DOCnnn.W11. If the 200th slot were in use, the size of this area would be 13 blocks. The maximum size of BITMAP.W11 is thus 14 blocks.



Example Suppose the WPS user creates document number 47. Checking through the slot table, WPS finds that the first free slot is slot number 22. In the document table, WPS changes byte 47 from 0 to 22. In the slot table, WPS changes byte 22 from 0 to 47. Then WPS copies the first 32 bytes of the new document's header into the 22nd slot.

Tools RSX–11M users can use the DMP utility to examine actual documents. DMP is described in Chapter 15 of the *RSX–11 Utilities Manual*, order number AA-H268A-TC. The DMP switches /HD and /AS are especially useful when examining documents.

wps

DX/AX Line Protocol 5

This chapter contains the *line protocol*, or set of rules, that the DX (Document Transfer) and AX (Automatic Transfer) packages use to send and receive WPS documents with another WPS unit. This protocol is also used by other DIGITAL operating systems when they are communicating with a WPS unit using DX Products.

Note

The CX (Character Transfer) package does not use a protocol. It simply sends characters one at a time to the other unit. This means that CX cannot automatically detect transmission errors nor represent the non-text parts of a document (such as highlighting).

Full-Duplex Serial Interface

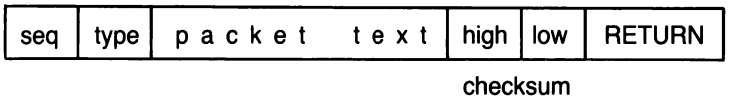
DX and AX engage in full-duplex communication with another WPS unit. This means a unit can send and receive characters at the same time. In practice, however, a unit never sends a document and receives a document simultaneously; only one major operation is in progress at any time.

When one unit sends information to another unit, it sends 8–bit characters serially as defined in the ANSI Standard. The eighth (high-order) bit may be used as a parity bit.

The rate of transmission is typically 300 Baud. The rate may be changed by setting both WPS units to communicate at the different rate. (Depending on the model, this may require adjustments by DIGITAL Field Service. Some communication hardware imposes an upper limit on the data rate.) Even on models where the data rate is software-controllable, neither DX nor AX automatically changes speeds to match the other unit.

Packets

All information is sent across the communication line in *packets*. A packet is a group of characters which identifies its own function and sequence and contains a checksum so that the receiver can detect transmission errors. There are nearly two dozen types of packets. Most are designated *high-level packets* and take the following form:



LEGEND: seq: the sequence number of the packet. This is a number in the range 00–77, encoded.[1]

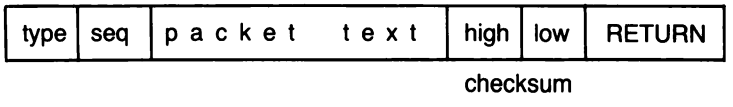
type: the type (or function) of the packet. Each type of packet is discussed and the packet type codes are presented later in this chapter.

packet text: up to 64 characters of text. Its function depends on the packet type. On some packet types, the text may be omitted.

checksum: the arithmetic sum of the characters sent before it, namely, the sequence number, packet type, and packet text. The checksum is a 12–bit number encoded[1] into two characters. (If the actual sum of the characters sent is greater than 4095, then only the low 12 bits are used as the checksum.)

RETURN: the carriage return, octal code 015. This the only ASCII control code which ever appears in DX/AX communication. The presence of the RETURN defines the length of the packet.

Five types of packets (NAK, ACK, INIT, INIT–ACK, and QUIT) are known as *low-level packets*. They follow the same format except that the packet type precedes the sequence number. But no code which could appear as a sequence number is used as a packet type.



The receiver of any high-level packet should promptly return the low-level packet ACK to the sender if the checksum and sequence number it received was what it expected. It should return NAK otherwise. ACK only says that the checksum was good; some packets require an additional high-level response.

---

[1] Anywhere this chapter talks about an *encoding* of binary data, it refers to the scheme introduced in Section 1.6.

Types of Packets

Summary of Packet Types

Type Code	Packet Name	Low–Level	Requires Response?
140	OK		
141	DOCUMENT OPTIONS		ANSWER
142	MESSAGE		
143	BYE		
144	START OF DOCUMENT		OK or NO
145	TEXT		
146	END OF DOCUMENT		
147	IDENTIFICATION		
151	SEND		
152	RECEIVE		
153	OPTIONS		
160	NAK	*	
164	NO		
170	ACK	*	
171	QUIT	*	
172	ANSWER TO PROMPT		
173	INIT–ACK	*	
174	INIT	*	
175	PROMPT WITH ECHO		ANSWER
176	PROMPT WITHOUT ECHO		ANSWER

Description of Packet Types

140 (OK).
Affirmative response to a packet which requires such an answer, such as START OF DOCUMENT.

141 (DOCUMENT OPTIONS).
AX sends this packet to DX when DX asks it to receive a document and gives the document a duplicate name. The packet text contains one character which tells DX which options to display to the user:

040 None
041 Top (T)
042 Bottom (A)
043 Top (T) and bottom (A)
044 Overwrite (O)
045 Overwrite (O) and top (T)
046 Overwrite (O) and bottom (A)
047 Overwrite (O), top (T), and bottom (A)

WPS currently uses only codes 043 and 047. DX must respond with an ANSWER TO PROMPT packet whose packet text shows which option the DX user chose:

040 User pressed Gold MENU to select different name
041 User selected O; supersedes old document contents
042 User selected T; add new text to top of document
043 User selected A; add new text to bottom

142 (MESSAGE).
The receiver is to display the packet text on its screen.

143 (BYE).
Same as MESSAGE but also aborts any transfer in progress.

144 (START OF DOCUMENT).
This is the first packet sent in each document. The packet text contains the present size (in blocks) of the document on the sender's storage device, followed by the document's 25 printer settings. Each of these 26 numbers is encoded into two text characters.

The receiver must reply with an OK or NO packet depending on whether it has enough space to store the document on the requested device. (Take into account the fact that WPS–11 blocks can store more text than WPS–8 blocks.) A document sent from WPS–11M to a 200 Series unit might require more blocks there than it did on WPS–11M.

145 (TEXT).
The packet text contains 64 characters of document text, using the WPS–11 format (see Chapter 1).

146 (END OF DOCUMENT).
Same as TEXT but indicates the last packet in the document. The packet text may contain fewer than 64 characters.

147 (IDENTIFICATION).
The first character of the packet text identifies the software: DX sends 040; AX sends 041. The remainder of the packet text is a descriptive message the receiver displays on its screen, such as "with a WS100 in DX."

151 (SEND).
The unit which sends this packet is ready to receive a document. The packet text contains the name of the document. This name has different implications when this packet is sent to AX. This is discussed shortly.

152 (RECEIVE).
The unit which sends this packet is ready to send a document. The packet text contains the name of that document. Transfer does not begin until the other unit sends a SEND packet.

153 (OPTIONS).
Sent to DX to tell it what major operations it should let its user select. For instance, if DX is talking to AX and the DX user has typed a "send-only" password, AX tells DX not to make the R(eceive) option available. The packet text contains one character:

040 Allow nothing
041 Allow the B(ye) option
042 Allow the B(ye) and M(essage) options
043 Allow B(ye), M(essage), and S(end)
044 Allow B(ye), M(essage), S(end), and R(eceive)

160 (NAK).
A response that indicates the sender received a packet with an improper checksum. The sequence number is that of the last packet received correctly.

164 (NO).
Negative response to packets which require such an answer, such as START OF DOCUMENT.

170 (ACK).
A positive acknowledgment. The sequence number is that of a packet received. ACK asserts that the named packet, and any previous ones since the last ACK or NAK, have been received successfully.

171 (QUIT).
Sent by a unit whose user has pressed Gold MENU to recall the Main Menu. Ends communications.

172 (ANSWER TO PROMPT).
Solicited by the DOCUMENT OPTIONS, PROMPT WITH ECHO, and PROMPT WITHOUT ECHO packets. The packet text represents information the sender's user typed.

173 (INIT–ACK).
A positive acknowledgment of INIT, with the same format.

174 (INIT).
Sent initially by DX until receipt of INIT–ACK establishes communication. The first character of the packet text is 040 to indicate the DX/AX protocol. The second character is the version number of the protocol in effect at the sending unit.

The third character is the number of NULs (octal 000) the sender of the packet wants the receiver to send after the RETURN character in all future packets. (Version 3.0 does not include a third character in this packet.)

175 (PROMPT WITH ECHO).
This packet asks the receiver to display the packet text on its screen, allow keyboard input, and send the response back in an ANSWER TO PROMPT packet.

176 (PROMPT WITHOUT ECHO).
Same as PROMPT WITH ECHO, but also requests that the receiver not display the user's response as it is typed. Used when the response contains secret information, such as a password.

Operation

For communication to be established between two WPS units, users at each unit must invoke the DX or AX packages from the Main Menu.

Initialization

DX sends INIT every 5 seconds until it receives either INIT or INIT–ACK. AX does not send anything, but rather waits for an INIT packet.

When either DX or AX receives INIT, it sends INIT–ACK. This is the signal to both the sender and receiver to synchronize their counts of sequence numbers. Each unit will assign sequence number 1 to the next packet it sends.

This packet should be an IDENTIFICATION packet. Each unit identifies itself to the other. If they are both in DX, they exchange OPTION packets and initialization ends.

If one unit is in AX, it may ask for information such as a password before sending an OPTION packet. DX never sends an OPTION packet to AX because AX runs unattended; there are no options to choose from.

Sending a Document

When a DX user invokes the S(end) option and answers all questions, DX sends a RECEIVE packet to the other unit. If the other unit is in AX, the packet text contains the name AX should give to the document. AX can respond in three ways:

- A NO packet means that AX cannot create the document. The packet text tells why; DX displays this error message on the screen.
- A SEND packet means that AX is creating a new document. AX invites DX to send the document now.
- A DOCUMENT OPTIONS packet means that the named document already exists. The DX user must specify what to do with the existing text. The packet text tells DX what choices to give its user. DX sends back an ANSWER TO PROMPT with the selection. Then AX sends the SEND packet.

If the other unit is in DX, the packet text of RECEIVE instead contains the document's current name on the sender's storage device.

The user of the receiving unit must specify the document's name on that unit. When that user has answered all questions, the receiver sends a SEND packet to the sender. This packet's text is the name of the document chosen to receive the text.

If the user of the other unit invoked the R(eceive) option and answered all questions, that unit could send a SEND request to the other unit before that unit got a RECEIVE packet.

If a DX user invokes the R(eceive) option to fetch a document from a unit running AX, DX sends a SEND packet containing the name of the document requested. AX responds with either a RECEIVE packet or a NO packet indicating that the document doesn't exist.

Once a SEND and a RECEIVE packet have been exchanged, the transfer begins. The sender sends a START OF DOCUMENT packet. This contains the number of blocks the document currently occupies on the sender's storage device. The receiver must compute whether it can fit that much text on its storage device, and must reply using either an OK or a NO packet.

If the receiver is in DX, its user may abort a transfer using the B(ye) option. DX sends a BYE packet to the other unit. If the receiver is in AX, it may abort a transfer by sending a NO packet. BYE and NO packets may be sent at any point in the transfer.

Time Limits

In several cases where the other unit does not respond promptly, WPS assumes the communication line has failed and repeats the initialization procedure described above.

You can send a packet before you have received ACK on packets sent previously. Depending on the situation, WPS can send up to five packets before it waits for ACK. (When sending a document, WPS will allow only two packets to be unacknowledged.) When the maximum number of unacknowledged packets is reached, WPS waits five seconds for ACK. If it does not receive ACK, it re-sends all outstanding packets and again waits for ACK. If the 5–second time-out is reached four consecutive times, the sender reinitalizes.

A unit sending a document may not let 30 seconds elapse between packets or the receiver will reinitialize.

Finally, if AX asks DX to obtain information from the DX user, DX must respond within 3 minutes or AX will reinitialize.

Errors

The checksum attached to each packet lets the receiver verify that it received the packet correctly. The checksum is the arithmetic sum of all previous characters in the packet. The receiver simply computes the sum of the characters it received. If the checksum matches, the receiver can send ACK; if the checksum reveals an error, the receiver should send NAK.

In both the ACK and NAK packets, a WPS unit uses as the sequence number the sequence number of the last packet it received correctly.

Thus, NAK can serve as a positive acknowledgment of some packets. NAK, however, asks the other unit to resend any packets that have not yet been acknowledged.

A unit could typically do the same thing by just not sending ACK or NAK. This would invoke the 5–second time-out. Sending NAK simply speeds things up. To minimize line traffic, however, WPS does not send more NAKs if the next packet(s) should also be bad.

When a unit receives a NAK while sending a packet, it aborts that packet by sending RESET (octal 175), then RETURN. RESET has no other meaning and is outside the range of characters which encode binary data. It therefore ensures that the checksum received will be faulty.

XON/XOFF To prevent characters from arriving faster than a unit can dispose of them, every unit immediately ceases to send characters if it receives XOFF (octal 023), and resumes sending where it left off when it receives XON (octal 021).

Reader's Comments

Your comments and suggestions are welcome. They will help us in our continuous effort to improve the quality and usefulness of our publications.

1. How would you rate this manual for:

	good	fair	poor
completeness of information	_____	_____	_____
accuracy of information	_____	_____	_____
ease of use (clarity, organization)	_____	_____	_____

2. Did you find errors in this manual? Please specify by page and paragraph, or mark corrections on pages and attach copies to this form.

	page	paragraph
a) Incorrect information:	_____	_____
b) Information left out:	_____	_____
c) Hard to understand:	_____	_____

3. What suggestions do you have for improving this manual?

_____

_____

_____

4. What parts of the manual were especially good at helping you understand things?

_____

_____

_____

5. Please check the boxes that apply to you.

- ☐ Technical
- ☐ Nontechnical
- ☐ Management
- ☐ Nonmanagement
- ☐ Other (explain)_____
- ☐ Used other word processors
- ☐ Used computers
- ☐ Use the word processing unit yourself

Name_____ Title_____

Company_____

Street_____ City_____

State_____ Zip_____ Date_____

—Do Not Tear - Fold Here and Tape—

digital

No Postage Necessary if Mailed in the United States

BUSINESS REPLY MAIL
FIRST CLASS PERMIT NO.33 MAYNARD MASS.

POSTAGE WILL BE PAID BY ADDRESSEE

Documentation Manager MK1-1/J14
Word Processing Computer Systems
DIGITAL EQUIPMENT CORPORATION
CONTINENTAL BOULEVARD
MERRIMACK, NH 03054

Do Not Tear - Fold Here and Tape

Cut Along Dotted Line


wps