DECUS
PROGRAM LIBRARY

DECUS NO.	8-385
TITLE	MIXED ASCII FORMATTING AND OUTPUT TECHNIQUE
AUTHOR	Donald L. Scanlon
COMPANY	Romelan Industries Santa Clara, California
DATE	November 3, 1970
SOURCE LANGUAGE	PAL III

MIXED ASCII FORMATTING AND OUTPUT TECHNIQUE

DECUS Program Library Write-up DECUS NO. 8-385

This technique provides a reasonably good trade off between storage requirements and processing time in real-time systems when both fixed and variable content messages are to be output on a particular device such as the teletype. It allows the fixed portion of the message to be stored in packed (two characters per word) format and the variable portion to be stored in unpacked (one character per word) standard ASCII for output. This eliminates the necessity for using a great deal of memory for storing unpacked messages to be output with unpacked data and the necessity for packing data values to be output with packed messages. This method has little effect on interrupt processing time.

A teletype message such as "NUMBER PASSED - number" where 'number' is a variable data would be stored as follows using the mixed format:

LOCATION	CONTENT
n	NU
n+1	MB
n+2	ER
n+3	P
n+4	AS
n+5	SE
n+6	D
n+7	-
n+8	n
n+9	u
n+10	m
n+11	b
n+12	e
n+13	r

The same message in unpacked ASCII format would require 22 core locations. Utilizing packed ASCII would only require 11 locations; but, would require packing a six digit number. Utilizing the mixed formatting technique, all of the variable characters are stored in their standard ASCII code, one character per word. All fixed characters are stored in packed, modified ASCII (ASCII - 233_8) code, two characters per word.

A study of the ASCII code set will reveal that in any 12 bit unpacked character, none of the most significant four bits is ever set. In the modified ASCII code, one of the upper four bits of any six bit character is always set. This allows packing two characters into a word and always having one of the most significant four bits set. The output routine then detects the presence or absence of any of the most significant bits in any data word to determine whether the word consists of one unpacked character or two packed characters and outputs the data accordingly. All legal ASCII characters except RETURN, LINE FEED, BELL, /, [,] , ← , and ↑ may be used in either format. These exceptions must be handled as unpacked ASCII.

The message "NUMBER PASSED - 2Ø3175" (where 2Ø3175 is a variable) would be stored as follows:

LOCATION	ASCII CONTENT	OCTAL CONTENT
n	NU	6372
n+1	MB	6247
n+2	ER	5267
n+3	P	Ø565
n+4	AS	467Ø
n+5	SE	7Ø52
n+6	D	51Ø5
n+7	-	22Ø5
n+8	2	Ø262

LOCATION	ASCII CONTENT	ASCII CONTENT
n+9	Ø	Ø26Ø
n+10	3	Ø263
n+11	1	Ø261
n+12	7	Ø267
n+13	5	Ø265

The portion of the interrupt routine used to process this message format is reproduced below.

```
INTCN,    TSF                   /IS IT THE TTY?
          JMP       INT11       /NO
          TAD    I  TOUTXR      /GET THE NEXT WORD
          AND       TTYMSK      /CHECK FOR PACKED
          SNA       CLA
          JMP       INT1Ø3      /UNPACKED ASCII
          ISZ       TTYF        /PACKED - WHICH HALF WORD
          JMP       INT1Ø1      /UPPER HALF WORD
          TAD    I  TOUTXR      /GET CHARACTER
          ISZ       TOUTXR      /INCREMENT TO NEXT WORD
          JMP       INT1Ø2
INT1Ø1,   CMA                   /RESET UPPER/LOWER FLAG
          DCA       TTYF
          TAD    I  TOUTXR      /GET UPPER HALF WORD
          RTR
          RTR
          RTR
INT1Ø2,   AND       TYMSK1      /GET CHARACTER
          SNA                   /CHECK FOR END OF MSG
          JMP       INT1ØB      /YES
          TAD       TTY233      /MAKE INTO 8-BIT ASCII
          JMP       INT1Ø4      /GO OUTPUT
INT1Ø3,   TAD    I  TOUTXR      /UNPACKED ASCII
          SNA                   /CHECK FOR END OF MESSAGE
          JMP       INT1ØB      /YES
          ISZ       TOUTXR      /INCREMENT TO NEXT WORD
INT1Ø4,   TLS                   /OUTPUT
          JMP       INTXT       /EXIT INTERRUPT ROUTINE
/
INT1ØB,   Code required to pick up the next message in
          a chain.  TTYF must also be set to -1.
```

The following Page zero locations were used:

```
TOUTXR,   Ø                     /POINTS TO NEXT WORD TO OUTPUT
TTYMSK,   74ØØ                  /MASK FOR UPPER FOUR BITS
TTYF,     -1                    /UPPER/LOWER HALF WORD FLAG
TTYMSK1,  77                    /MASK FOR LOWER SIX BITS
```

```
TTY233,   233                  /CONSTANT TO CONVERT PACKED
                               /  CHARACTER TO UNPACKED CHAR
```

NOTE: In this example, the requirement was established that a message had to terminate with either the lower 6 bits or the entire last data word set to binary zero.

A list of the modified and unmodified ASCII codes used in this technique is attached.

PACKED AND UNPACKED ASCII CODES
(Packed = 8-bit ASCII - 233_8)

CHAR	UNPACKED	PACKED	CHAR	UNPACKED	PACKED
A	3Ø1	46	8	27Ø	35
B	3Ø2	47	9	271	36
C	3Ø3	5Ø	!	241	Ø6
D	3Ø4	51	"	242	Ø7
E	3Ø5	52	#	243	1Ø
F	3Ø6	53	$	244	11
G	3Ø7	54	%	245	12
H	31Ø	55	&	246	13
I	311	56	'	247	14
J	312	57	(	25Ø	15
K	313	6Ø	)	251	16
L	314	61	*	252	17
M	315	62	+	253	2Ø
N	316	63	,	254	21
O	317	64	-	255	22
P	32Ø	65	.	256	23
Q	321	66	/	257	24
R	322	67	:	272	37
S	323	7Ø	;	273	4Ø
T	324	71	<	274	41
U	325	72	=	275	42
V	326	73	>	276	43
W	327	74	?	277	44
X	33Ø	75	@	3ØØ	45
Y	331	76	[	333	--
Z	332	77	/	334	--
Ø	26Ø	25	]	335	--
1	261	26	↑	336	--
2	262	27	←	337	--
3	263	3Ø	BELL	2Ø7	--
4	264	31	LINE FEED	212	--
5	265	32	RETURN	215	--
6	266	33	SPACE	24Ø	Ø5
7	267	34			