

11) A Couple of Quick FLEX Tips

Here are a couple of quick tips to use in your FLEX related software that you may not have thought of.

a) QUICK FILE TEST

Ever want to test a file to see if it could be read without errors? If you have our FLEX Diagnostics package there is a utility for doing just that. There is a crude method of performing such a test using the standard FLEX "LIST" utility. By placing a line number on the LIST command after the filename, it is possible to suppress all listing until that line is hit. If we specify a very large number such as 9999, the LIST command will never hit that line, but will have read through the entire file looking for it. Thus we have tested the file. This technique can also be applied to binary type files since they probably won't have many carriage returns in them which is what LIST looks for as line terminators. For example, to test the COPY command for read errors enter the command:

```
+++LIST 0.COPY.CMD 9999
```

b) DISABLE TWO SPACES AFTER PERIODS IN TEXT PROCESSOR

Our Text Processing System program always attempts to ensure that there are two spaces after every sentence punctuation. In other words, if the processor sees a period, question mark, or exclamation point followed by a space, it assumes it is at the end of a sentence and that there should be two spaces following the punctuation character. If that is not the case in the input text, the processor will insert a space in the formatted output. There are times when this is not desired. For example, consider the name "Mr. Jones". When the text processor saw the period followed by a space it would think it was at the end of a sentence and add a space. This would probably not be desired. A simple way around the situation is to be sure the period does not look like a sentence punctuation character. This is most easily accomplished by making the space that follows the period a non-paddable space character by use of the backslash. For example: "Mr.® Jones". Now the period is not followed by a real space, and it will not look like punctuation to the text processor.

12) Free FLEX Utility!

We are giving you another free FLEX utility in this newsletter. It dumps a binary file in the Motorola S1/S9 ASCII hex format and is called "SIDUMP". As written, the output is routed to the terminal. If desired, the user can route the output to a printer with the "P" command or to a disk file with the "O" command. Instructions for use are given in the listing itself. This listing is in 6800 code, but may be easily converted to 6809 by changing the equates and the ORG statement.

```

* S1DUMP UTILITY FOR 6800
*
* COPYRIGHT (C) 1980 BY
* TECHNICAL SYSTEMS CONSULTANTS, INC.
* PO BOX 2570, WEST LAFAYETTE, IN 47906
*
* THIS UTILITY DUMPS A BINARY FILE IN THE MOTOROLA
* S1/S9 ASCII HEX FORMAT.  OUTPUT IS SENT TO TERMINAL
* BUT CAN BE ROUTED TO DISK VIA THE FLEX "O" COMMAND.
* A SINGLE LINE OF TITLE CAN BE SENT WITH THE DUMP BY
* PLACING IT ON THE COMMAND LINE AS SHOWN HERE:
*     +++S1DUMP FILENAME THIS IS A TITLE
* WHERE "THIS IS A TITLE" WOULD BE OUTPUT BEFORE THE
* DATA AS A TITLE.  THE FILENAME DEFAULTS TO ".BIN".
*

```

* EQUATES

```

AD03      WARMS   EQU    $AD03
A840      FCB     EQU    $A840
AD18      PUTCHR EQU    $AD18
AD24      PCRLF  EQU    $AD24
AD2D      GETFIL EQU    $AD2D
AD33      SETEXT EQU    $AD33
AD3F      RPTERR EQU    $AD3F
AC14      BUFPNT EQU    $AC14
B403      FMSCLS EQU    $B403
B406      FMS     EQU    $B406

```

* MAIN PROGRAM

```

A100                      ORG    $A100

A100 20 08      S1DUMP  BRA    SDMP1
A102 01          VN      FCB    1          VERSION NUMBER 1

```

* TAPE OUTPUT CHARACTER ROUTINE (CAN BE PATCHED)

```

A103 7E AD 18  TOUCH   JMP     PUTCHR

```

* TEMPORARY STORAGE

```

A106      LENGTH  RMB    1
A107      ADDR    RMB    2
A109      CHKSUM  RMB    1

```

```

A10A CE A8 40  SDMP1   LDX    #FCB
A10D BD AD 2D          JSR    GETFIL      GET FILENAME FROM COMMAND
A110 24 03          BCC    SDMP2
A112 7E A1 F9          JMP    ERROR
A115 4F          SDMP2   CLR    A          DEFAULT EXTENSION IS .BIN
A116 BD AD 33          JSR    SETEXT
A119 CE A8 40          LDX    #FCB
A11C 86 01          LDA    A          #1          OPEN FOR READ
A11E A7 00          STA    A          0,X

```

A120	BD	B4	06		JSR	FMS	
A123	27	03			BEQ	SDMP3	
A125	7E	A1	F9		JMP	ERROR	
A128	86	FF		SDMP3	LDA A	#\$FF	
A12A	A7	3B			STA A	59,X	TURN OFF SPACE COMPRESSION
A12C	BD	AD	24		JSR	PCRLF	
A12F	FE	AC	14		LDX	BUFNT	POINT TO TITLE IN BUFFER
A132	BD	A1	ED		JSR	PDATA2	SEND IT TO TAPE
A135	BD	AD	24		JSR	PCRLF	
A138	BD	A1	C5	SDMP4	JSR	READ	GET A CHARACTER
A13B	81	16			CMP A	#\$16	A TRANSFER ADDRESS?
A13D	26	08			BNE	SDMP5	
A13F	BD	A1	C5		JSR	READ	IF SO, EAT UP ADDRESS
A142	BD	A1	C5		JSR	READ	
A145	20	F1			BRA	SDMP4	
A147	81	02		SDMP5	CMP A	#\$02	A BINARY RECORD HEADER?
A149	26	ED			BNE	SDMP4	IGNORE IF NOT
A14B	BD	A1	C5		JSR	READ	GET LOAD ADDRESS
A14E	B7	A1	07		STA A	ADDR	
A151	BD	A1	C5		JSR	READ	
A154	B7	A1	08		STA A	ADDR+1	
A157	BD	A1	C5		JSR	READ	GET RECORD LENGTH
A15A	B7	A1	06		STA A	LENGTH	
A15D	B6	A1	06	SDMP6	LDA A	LENGTH	
A160	27	D6			BEQ	SDMP4	NEXT RECORD IF FINISHED
A162	16				TAB		
A163	C1	10			CMP B	#16	IS LENGTH > 16?
A165	23	02			BLS	SDMP7	SKIP IF NOT
A167	C6	10			LDA B	#16	ELSE, LIMIT TO 16
A169	10			SDMP7	SBA		
A16A	B7	A1	06		STA A	LENGTH	SET NEW LENGTH
A16D	8D	02			BSR	PUNREC	PUNCH ONE RECORD
A16F	20	EC			BRA	SDMP6	GO FINISH RECORD

* ROUTINE TO PUNCH ONE RECORD WITH "B" EQUAL TO NUMBER
 * OF DATA BYTES TO BE PUNCHED.

A171	BD	AD	24	PUNREC	JSR	PCRLF	
A174	CE	A2	02		LDX	#S1	PRINT HEADER
A177	BD	A1	E0		JSR	PDATA	
A17A	7F	A1	09		CLR	CHKSUM	
A17D	17				TBA		
A17E	8B	03			ADD A	#3	
A180	8D	20			BSR	OUTHEX	OUTPUT RECORD LENGTH
A182	B6	A1	07		LDA A	ADDR	
A185	8D	1B			BSR	OUTHEX	OUTPUT LOAD ADDRESS
A187	B6	A1	08		LDA A	ADDR+1	
A18A	8D	16			BSR	OUTHEX	
A18C	BD	A1	C5	PUNRC1	JSR	READ	GET A DATA BYTE
A18F	8D	11			BSR	OUTHEX	OUTPUT IT
A191	FE	A1	07		LDX	ADDR	
A194	08				INX		BUMP ADDRESS POINTER

```

A195 FF A1 07      STX   ADDR
A198 5A           DEC   B      FINISHED WITH RECORD?
A199 26 F1       BNE   PUNRC1  LOOP IF NOT

A19B B6 A1 09      LDA   A      CHKSUM  OUTPUT CHECKSUM
A19E 43          COM   A
A19F 8D 01       BSR   OUTHEX
A1A1 39          RTS

```

* OUTPUT BYTE IN "A" AS TWO ASCII HEX DIGITS

```

A1A2 36          OUTHEX PSH   A
A1A3 44          LSR   A
A1A4 44          LSR   A
A1A5 44          LSR   A
A1A6 44          LSR   A
A1A7 8B 90       ADD   A    #$90
A1A9 19          DAA
A1AA 89 40       ADC   A    #$40
A1AC 19          DAA
A1AD BD A1 03    JSR   TOUCH
A1B0 32          PUL   A
A1B1 36          PSH   A
A1B2 84 0F       AND   A    #$0F
A1B4 8B 90       ADD   A    #$90
A1B6 19          DAA
A1B7 89 40       ADC   A    #$40
A1B9 19          DAA
A1BA BD A1 03    JSR   TOUCH
A1BD 32          PUL   A
A1BE BB A1 09    ADD   A    CHKSUM
A1C1 B7 A1 09    STA   A    CHKSUM
A1C4 39          RTS

```

* READ ONE CHARACTER FROM FILE

```

A1C5 CE A8 40    READ   LDX   #FCB
A1C8 BD B4 06    JSR   FMS
A1CB 26 01       BNE   READ1
A1CD 39          RTS           RETURN IF GOOD
A1CE 32          READ1  PUL   A           FIX STACK
A1CF 32          PUL   A
A1D0 A6 01       LDA   A    1,X    GET ERROR NUMBER
A1D2 81 08       CMP   A    #8     END OF FILE ERROR?
A1D4 26 23       BNE   ERROR
A1D6 BD AD 24    JSR   PCRLF
A1D9 CE A2 05    LDX   #S9       PRINT END OF FILE MARKER
A1DC 8D 02       BSR   PDATA
A1DE 20 1C       BRA   EXIT     BACK TO FLEX

```

* PRINT STRING TO TAPE

```

A1E0 A6 00      PDATA  LDA   A    0,X
A1E2 81 04      CMP   A    #4

```

A1E4	27	06		BEQ	PDATA1
A1E6	BD	A1	03	JSR	TOUCH
A1E9	08			INX	
A1EA	20	F4		BRA	PDATA
A1EC	39		PDATA1	RTS	
A1ED	A6	00	PDATA2	LDA	A 0,X
A1EF	81	0D		CMP	A #\$0D
A1F1	27	F9		BEQ	PDATA1
A1F3	BD	A1	03	JSR	TOUCH
A1F6	08			INX	
A1F7	20	F4		BRA	PDATA2

* ERROR ROUTINE

A1F9	BD	AD	3F	ERROR	JSR	RPTERR
A1FC	BD	B4	03	EXIT	JSR	FMSCLS
A1FF	7E	AD	03		JMP	WARMS

* PRINT STRINGS

A202	53	S1	FCB	'S, '1, 4
A205	53	S9	FCB	'S, '9, 4

END S1DUMP

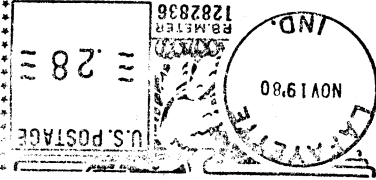
NO ERROR(S) DETECTED

Dear Subscriber:
The number in the upper right-hand corner of the mailing label designates the number of the last issue you will receive. If you wish to continue to receive this newsletter be sure to renew your subscription when the current issue number matches that on the label.

David C. Wiens
2261 East 14th Ave.
Vancouver, B.C.
CANADA V5N 1Z7

06

PRIORITY MAIL



technical systems
consultants, inc.
Box 2570, West Lafayette, IN 47906