

FDD3:48	407	PHA		
FDD4:A9 BD	408	LDA	#\$BD	;PRINT '-', THEN RESULT
FDD6:20 ED FD	409	JSR	COUT	
FDD9:68	410	PLA		
FDDA:48	411	PRBYTE	PHA	;PRINT BYTE AS 2 HEX DIGITS
FDDB:4A	412	LSR	A	; (DESTROYS A-REG)
FDDC:4A	413	LSR	A	
FDDD:4A	414	LSR	A	
FDDE:4A	415	LSR	A	
FDDF:20 E5 FD	416	JSR	PRHEXZ	
FDE2:68	417	PLA		
FDE3:29 0F	418	PRHEX	AND	;\$OF ;PRINT HEX DIGIT IN A-REG
FDE5:09 80	419	PRHEXZ	ORA	;\$B0 ;LSBITS ONLY.
FDE7:C9 BA	420	CMP	;\$BA	
FDE9:90 02	421	BCC	COUT	
FDEB:69 06	422	ADC	;\$06	
FDED:	423 *			
FDED:6C 36 00	424	COUT	JMP (CSWL)	;VECTOR TO USER OUTPUT ROUTINE
FDFO:	425 *			
FDFO:48	426	COUT1	PHA	;save original character
FDFl:C9 AO	427	CMP	;\$AO	;is it a control?
FDf3:4C 95 FC	428	JMP	DOCOUT1	;=>mask if not; return to COUTZ1
FDf6:	429 *			
FDf6:48	430	COUTZ	PHA	;save original character
FDf7:84 35	431	COUTZ1	STY YSAV1	;save Y
FDf9:A8	432	TAY		;save masked character
FDfA:68	433	PLA		;get original char
FDfB:4C 46 FC	434	JMP	NEWVW	;new entry to vidwait
FDfE:EA	435	NOP		
FDff:EA	436	NOP		
FE00:	437 *			
FE00:C6 34	438	BL1	DEC YSAV	
FE02:FO 9F	439	BEQ	XAM8	
FE04:CA	440	BLANK	DEX	;BLANK TO MON
FE05:DO 16	441	BNE	SETMDZ	;AFTER BLANK
FE07:C9 BA	442	CMP	;\$BA	;DATA STORE MODE?
FE09:DO BB	443	BNE	XAMPM	; NO; XAM, ADD, OR SUBTRACT.
FE0B:85 31	444	STOR	STA MODE	;KEEP IN STORE MODE
FE0D:A5 3E	445	LDA	A2L	
FE0F:91 40	446	STA	(A3L),Y	;STORE AS LOW BYTE AT (A3)
FE11:E6 40	447	INC	A3L	
FE13:DO 02	448	BNE	RTS5	
FE15:E6 41	449	INC	A3H	;INCR A3, RETURN.
FE17:60	450	RTS5	RTS	
FE18:	451 *			
FE18:A4 34	452	SETMODE	LDY YSAV	;SAVE CONVERTED ':', '+',
FE1A:B9 FF 01	453	LDA	IN-1,Y	; '-', '.' AS MODE
FE1D:85 31	454	SETMDZ	STA MODE	
FE1F:60	455	RTS		
FE20:	456 *			
FE20:A2 01	457	LT	LDX \$S01	
FE22:85 3E	458	LT2	LDA A2L,X	;COPY A2 (2 BYTES) TO
FE24:95 42	459	STA	A4L,X	; A4 AND A5
FE26:95 44	460	STA	A5L,X	

FE28:CA	461	DEX		
FE29:10 F7	462	BPL	LT2	
FE2B:60	463	RTS		
FE2C:	464 *			
FE2C:B1 3C	465	MOVE	LDA (A1L),Y	;MOVE (A1) THRU (A2) TO (A4)
FE2E:91 42	466	STA	(A4L),Y	
FE30:20 B4 FC	467	JSR	NXTA4	
FE33:90 F7	468	BCC	MOVE	
FE35:60	469	RTS		
FE36:	470 *			
FE36:B1 3C	471	VFY	LDA (A1L),Y	;VERIFY (A1) THRU (A2)
FE38:D1 42	472	CMP	(A4L),Y	; WITH (A4)
FE3A:FO 1C	473	BEQ	VFYOK	
FE3C:20 92 FD	474	JSR	PRAI	
FE3F:B1 3C	475	LDA	(A1L),Y	
FE41:20 DA FD	476	JSR	PRBYTE	
FE44:A9 AO	477	LDA	;\$AO	
FE46:20 ED FD	478	JSR	COUT	
FE49:A9 A8	479	LDA	;\$A8	
FE4B:20 ED FD	480	JSR	COUT	
FE4E:B1 42	481	LDA	(A4L),Y	
FE50:20 DA FD	482	JSR	PRBYTE	
FE53:A9 A9	483	LDA	;\$A9	
FE55:20 ED FD	484	JSR	COUT	
FE58:20 B4 FC	485	VFYOK	JSR NXTA4	
FE5B:90 D9	486	BCC	VFY	
FE5D:60	487	RTS		
FE5E:	488 *			
FE5E:20 75 FE	489	LIST	JSR A1PC	;MOVE A1 (2 BYTES) TO
FE61:A9 14	490	LDA	;\$14	; PC IF SPEC'D AND
FE63:48	491	LIST2	PHA	; DISASSEMBLE 20 INSTRUCTIONS.
FE64:20 DO F8	492	JSR	INSTDSP	
FE67:20 53 F9	493	JSR	PCADJ	;ADJUST PC AFTER EACH INSTRUCTION.
FE6A:85 3A	494	STA	PCL	
FE6C:84 3B	495	STY	PCH	
FE6E:68	496	PLA		
FE6F:38	497	SEC		
FE70:E9 01	498	SBC	;\$01	;NEXT OF 20 INSTRUCTIONS
FE72:DO EF	499	BNE	LIST2	
FE74:60	500	RTS		
FE75:	501 *			
FE75:8A	502	A1PC	TXA	;IF USER SPECIFIED AN ADDRESS,
FE76:FO 07	503	BEQ	A1PCRTS	; COPY IT FROM A1 TO PC.
FE78:B5 3C	504	A1PCLP	LDA A1L,X	;YEP, SO COPY IT.
FE7A:95 3A	505	STA	PCL,X	
FE7C:CA	506	DEX		
FE7D:10 F9	507	BPL	A1PCLP	
FE7F:60	508	A1PCRTS	RTS	
FE80:	509 *			
FE80:A0 3F	510	SETINV	LDY \$S3F	;SET FOR INVERSE VID
FE82:DO 02	511	BNE	SETIFLG	; VIA COUT1
FE84:A0 FF	512	SETNORM	LDY \$SFF	;SET FOR NORMAL VID
FE86:84 32	513	SETIFLG	STY INVFLG	
FE88:60	514	RTS		