

LINE #	LOC	CODE	LINE
2326	EF79	68	PLA
2327	EF7A	60	RTS
2329	EF7B		; CONVERT X INTO REAL ADDR FOR DISPLAY
2330	EF7B		; AND OUTPUT IT PB=DATA ; PA=W,CE ,A0 A1 (6520)
2331	EF7B	48	OUTDD1 PHA ; SAVE DATA
2332	EF7C	8A	TXA
2333	EF7D	48	PHA ; SAVE X
2334	EF7E	4A	LSR A ; DIVIDE X BY 4
2335	EF7F	4A	LSR A ; TO GET CHIP SELECT
2336	EF80	AA	TAX ; BACK TO X
2337	EF81	A9 04	LDA #4 ; FIRST CHIP SELECT
2338	EF83	E0 00	CFX #0 ; FIRST CHIP ?
2339	EF85	F0 04	BEQ OUTDD3
2340	EF87	0A	OUTDD2 ASL A
2341	EF88	CA	DEX
2342	EF89	D0 FC	BNE OUTDD2 ; BACK TILL RIGH CS
2343	EF8B	8D 28 A4	OUTDD3 STA STIY+1 ; SAVE CS TEMPORARILY
2344	EF8E	68	PLA ; GET X AGAIN FOR CHAR
2345	EF8F	29 03	AND ##03 ; IN THAT CHIP
2346	EF91	0D 28 A4	ORA STIY+1 ; OR IN CS AND CHAR
2347	EF94		; STORE ADDR AND DATA INTO DISPL
2348	EF94	49 FF	EDR ##FF ; W=1 , CE=0 & A1,A0
2349	EF96	8D 00 AC	STA RA
2350	EF99	AA	TAX ; SAVE A IN X
2351	EF9A	68	PLA ; GET DATA
2352	EF9B	48	PHA
2353	EF9C	8D 02 AC	STA RB
2354	EF9F	8A	TXA
2355	EFA0	49 80	EDR ##80 ; SET W=0
2356	EFA2	8D 00 AC	STA RA
2357	EFA5	EA	NOP
2358	EFA6	09 7C	ORA ##7C ; SET CE=1
2359	EFA8	8D 00 AC	STA RA
2360	EFA8	A9 FF	LDA ##FF ; SET W=1
2361	EFA9	8D 00 AC	STA RA
2362	EFB0	68	PLA ; RETURN DATA
2363	EFB1	60	RTS
2365	EFB2		*=\$EFF9
2366	EFF9	EA	.BYT \$EA
2367	EFFA		*=\$F000
2368	F000		;;
2369	F000		; OUTPUT ACC TO PRINTER SUBROUTINE
2370	F000		; PRINTS ON FIRST CHAR OR WHEN (CCR)
2371	F000		; IT WILL PUT IT ON BUFFER BUT WONT PRINT IF
2372	F000		; PRIFLG=0
2373	F000	48	OUTPRI PHA ; SAVE CHR TO BE OUTPUT
2374	F001	20 9E EB	JSR PHXY ; SAVE X
2375	F004	C9 0D	CMF ##0D ; SEE IF CR
2376	F006	F0 07	BEQ OUT01 ; YES SO PRINT THE BUFF
2377	F008	AE 16 A4	LDX CURPOS ; PTR TO NEXT POS IN BUFF
2378	F00B	E0 14	CPX #20 ; SEE IF BUFF FULL

LINE #	LOC	CODE	LINE
2379	F00D	D0 16	BNE OUT04 ; NOT FULL SO RETURN
2380	F00F		; CCR) SO FILL REST OF BUFFER WITH BLANKS
2381	F00F	48	OUT01 PHA
2382	F010	A9 00	LDA #0 ; CURPOS = 0
2383	F012	AE 16 A4	LDX CURPOS ; SEE IF ANYTHING IN BUFFER
2384	F015	8D 16 A4	STA CURPOS
2385	F018	20 38 F0	JSR OUTPR ; CLEAR PRIBUF TO THE RIGHT
2386	F01B		; BUFFER FILLED SO PRINT IT
2387	F01B	20 45 F0	JSR IPST ; START THE PRINT
2388	F01E	A2 00	LDX #0 ; STORE CHR IN BUFF (FIRST LOC)
2389	F020	68	PLA ; GET IT
2390	F021	C9 0D	CMF #CR ; DONT STORE IF (CCR)
2391	F023	F0 0E	BEQ OUT05
2392	F025	9D 60 A4	OUT04 STA IBUFH,X ; STORE CHR IN BUFF
2393	F028	EE 16 A4	INC CURPOS ; INCR BUFF PNTR
2394	F02B	E8	INX
2395	F02C	29 80	AND ##80
2396	F02E	D0 03	BNE OUT05 ; DONT CLR IF MSB=1
2397	F030	20 38 F0	JSR OUTPR ; CLEAR PRIBUFF TO THE RIGHT
2398	F033	20 AC EB	OUT05 JSR FLXY ; RESTORE REGS
2399	F036	68	PLA
2400	F037	60	RTS
2401	F038	A9 20	OUTPR LDA ##20 ; FILL REST OF BUFF WITH BLANKS
2402	F03A	E0 14	OUTPR1 CPX #20 ; SEE IF END OF BUFF
2403	F03C	F0 06	BEQ OUTPR2
2404	F03E	9D 60 A4	STA IBUFH,X ; NO SO STORE BLANK
2405	F041	EB	INX ; INCR BUFF PNTR
2406	F042	10 F6	BFL OUTPR1
2407	F044	60	OUTPR2 RTS
2409	F045		; SUB TO OUTPUT BUFFER, 70 DOTS (10 DOTS AT
2410	F045		; A TIME BY 7 ROWS) FOR EACH LINE OF PRINTING
2411	F045	2C 11 A4	IPST BIT PRIFLG ; PRINT FLG ON ?
2412	F048	10 2E	BFL IF04
2413	F04A	20 CB F0	IPSO JSR PINT ; INITIALIZE VALUES
2414	F04D	20 E3 F0	JSR IPSU ; SET UP FIRS OUTPUT PATTERN
2415	F050	A9 C1	IP00 LDA #PRST+SP12+MON ; TURN MOTOR ON
2416	F052	8D 0C AB	STA PCR
2417	F055	20 A0 FF	JSR PAT23 ; TIME OUT?
2418	F058	D0 0C	BNE IP02 ; NO, START SIGNAL RECEIVED
2419	F05A	20 A0 FF	JSR PAT23 ; YES, TRY AGAIN
2420	F05D	D0 07	BNE IP02 ; OK
2421	F05F	4C 79 F0	JMP FRIERR ; TWO TIME OUTS - ERROR
2422	F062	EA	NOP
2423	F063	EA	NOP
2424	F064	EA	NOP
2425	F065	EA	NOP
2426	F066	20 87 F0	IP02 JSR PRNDOT ; STRB F1=1 PRINT DOTS (1.7MSEC)
2427	F069	20 87 F0	JSR PRNDOT ; STRB F2=1 PRINT DOTS (1.7MSEC)
2428	F06C		; CHECK FOR 90, WHEN 70 PRNDOT WILL OUTPUT ZEROS
2429	F06C	AD 77 A4	LDA IDDT
2430	F06F	C9 5A	CMF #90
2431	F071	90 F3	BCC IP02 ; L.T. 90 THEN GO STROB F1
2432	F073	A9 E1	IP03 LDA #PRST+SP12+MOFF ; TURN MOTOR OFF
2433	F075	8D 0C AB	STA PCR

```

LINE # LOC      CODE      LINE
2434 F078 60          IP04 RTS
2436 F079 20 44 EB      PRIERR JSR CLR          ;CLEAR PRI PNTR
2437 F07C 20 B1 FE          JSR PATCH5          ;TURN PRI OFF
2438 F07F A0 3B          LDY #M12-M1
2439 F081 20 AF E7          JSR KFP
2440 F084 4C A1 E1          JMP COMIN          ;BACK WHERE SUBR WAS CALLED

2442 F087          ;SUBR TO INCR DOT COUNTER WHEN
2443 F087          ;NEG TRANS OUTPUT CHR FOR 1.7 MSEC
2444 F087          ;CLEAR & SET UP NEXT PATTERN
2445 F087 A9 00          PRNDOT LDA #0          ;CLR INTERRUPTS
2446 F089 8D 01 AB          STA DRAH
2447 F08C AD 0D AB          PRDOTO LDA IFR
2448 F08F 29 02          AND #MSP12          ;ANY STROBES ?
2449 F091 F0 F9          BEQ PRDOTO
2450 F093 AD 0C AB          LDA PCR
2451 F096 49 01          EOR #*01
2452 F098 8D 0C AB          STA PCR
2453 F09B EE 77 A4          INC IDOT
2454 F09E AD 79 A4          LDA IOUTU          ;2 LEFT ELEM
2455 FOA1 0D 00 AB          ORA DRB          ;DO NOT TURN TTY OUTPUT OFF
2456 FOA4 8D 00 AB          STA DRB
2457 FOA7 AD 78 A4          LDA IOUTL          ;7 RIGHT ELEM, CLR CA1 INTER FLB
2458 FOAA 8D 01 AB          STA DRAH
2459 FOAD A9 A4          LDA #CFRIME
2460 FOAF 8D 08 AB          STA T2L
2461 F0B2 A9 06          LDA #DPRIME          ;START T2 FOR 1.7 MSEC
2462 F0B4 8D 09 AB          STA T2H
2463 F0B7 20 E3 F0          JSR IPSU          ;SET NEXT PATTERN WHILE WAITING
2464 F0BA 20 1B EC          JSR DE2          ;WAIT TILL TIME OUT
2465 F0BD A9 00          LDA #0          ;THERMAL ELEM OFF
2466 F0BF 8D 01 AB          STA DRAH
2467 F0C2 AD 00 AB          LDA DRB          ;BUT DONT CHANGE TAPE CONTROLS
2468 F0C5 29 FC          AND #*FC
2469 F0C7 8D 00 AB          STA DRB
2470 FOCA 60          RTS

2472 F0CB          ; SUBROUTINE PINT --- INIT VARS FOR PRINTER
2473 F0CB A9 FF          PINT LDA #*FF
2474 F0CD 8D 74 A4          STA IDIR          ;DIRECTION (<= -
2475 F0D0 A9 05          LDA #5
2476 F0D2 8D 75 A4          STA ICOL          ;COLUMN (<= LEFTMOST +1
2477 F0D5 A9 01          LDA #1
2478 F0D7 8D 76 A4          STA IOFFST          ;OFFSET (<= LEFT CHARACTER
2479 F0DA 8D 7C A4          STA IMASK
2480 F0DD A9 00          LDA #0
2481 F0DF 8D 77 A4          STA IDOT          ;DOT COUNTER (<= 0
2482 F0E2 60          RTS

2484 F0E3          ;THE VARIABLES FOR THE PRINTER ARE AS FOLLOWS:
2485 F0E3
2486 F0E3          ;IDIR DIRECT HEAD IS CURRENTLY MOVING (0=+, *FF=-)
2487 F0E3          ;ICOL CLMN TO BE FRNTD NEXT (LEFTMOST=0,RIGHTMOST=4
2488 F0E3          ;IOFFST OFFSET N PRINT BUFF (0=LEFT CHR, 1=RIGHT CHR)

```

```

LINE # LOC      CODE      LINE
2489 F0E3          ;IDOT COUNT OF NUMBER OF DOTS PRINTED THUS FAR
2490 F0E3          ;IOUTL SOLENOID PATTERN (8 CHRS ON RIGHT)
2491 F0E3          ;IOUTU SOLENOID PATTERN (2 CHRS ON LEFT)
2492 F0E3          ;IBITL 1 BIT MSK USED IN SETTING NEXT SOLENOID VALUE
2493 F0E3          ;IBITU UPPER PART OF MASK
2494 F0E3          ;IBUFM START OF PRINT BUFFER (LEFTMOST CHR FIRST)
2495 F0E3          ;IMASK MASK FOR CURRENT ROW BEING PRINTED
2496 F0E3          ;JUMP ADDRESS OF TABLE FOR CURRENT COLUMN
2497 F0E3
2498 F0E3          ; THE DOT PATTERNS FOR THE CHRS ARE STORED SO THAT...
2499 F0E3          ;EACH BYTE CONTAINS THE DOTS FOR ONE COLUMN OF ONE...
2500 F0E3          ;CHR. SINCE EACH COLUMN CONTAINS SEVEN DOTS...
2501 F0E3          ;THIS MEANS THAT ONE BIT PER BYTE IS UNUSED.
2502 F0E3          ; THE PATTERNS ARE ORGANIZED INTO 5 TABLES OF 64...
2503 F0E3          ;BYTES WHERE EACH TABLE CONTAINS ALL THE DOT...
2504 F0E3          ;PATTERNS FOR A PARTICULAR COLUMN. THE BYTES IN EACH...
2505 F0E3          ;TABLE ARE ORDERED ACCORDING TO THE CHR CODE OF...
2506 F0E3          ;THE CHR BEING REFERENCED. THE CHR CODE CAN...
2507 F0E3          ;THUS BE USED TO DIRECTLY INDEX INTO THE TABLE.

2509 F0E3          ;SUBROUTINE IPSU -- SET UP OUTPUT PATTERN FOR PRINTER
2510 F0E3          ; THIS ROUTINE IS CALLED IN ORDER TO
2511 F0E3          ;SET UP THE NEXT GROUP OF SOLENOIDS TO
2512 F0E3          ;BE OUTPUT TO THE PRINTER.
2513 F0E3          ; ON ENTRY THE CONTENTS OF ALL REGISTERS
2514 F0E3          ;ARE ARBITRARY
2515 F0E3          ; ON EXIT THE CONTENTS OF A,X,Y ARE UNDEFINED
2516 F0E3 A2 00          IPSU LDX #0          ;X POINTS TO VAR BLOCK FOR PNTR
2517 F0E5 20 21 F1          JSR INCP          ;ADVANCE PTRS TO NXT DOT POSITION
2518 F0E8          ;X NOW CONTAINS INDEX INTO PRINT BUFFER
2519 F0E8 8D 60 A4          IPS1 LDA IBUFM,X    ;LOAD NEXT CHAR FROM BUFFER
2520 F0EB 29 3F          AND #*3F
2521 F0ED AB          TAY
2522 F0EE A9 7D          LDA #CJUMP          ;AC= DOT PATTERN FOR CHAR & COL
2523 F0F0 20 58 EB          JSR LDAY
2524 F0F3 2C 7C A4          BIT IMASK          ;SEE IF DOT IS SET
2525 F0F6 F0 16          BEQ IPS2          ;NO SO GO ON TO NEXT CHAR
2526 F0F8 AD 7A A4          LDA IBITL          ;DOT ON SO SET THE CURR SOLENOID
2527 F0FB F0 08          BEQ IPS3          ;LSB OF SOL MASK IS 0, DO MSB
2528 F0FD 0D 78 A4          ORA IOUTL          ;SET THE SOLENOID IN THE PATTERN
2529 F100 8D 78 A4          STA IOUTL
2530 F103 D0 09          BNE IPS2          ;BRANCH ALWAYS
2531 F105 AD 7B A4          LDA IBITU          ;SOLENOID IS ONE OF THE 2 MSD
2532 F108 0D 79 A4          ORA IOUTU          ;SET THE BIT IN THE PATTERN
2533 F10B 8D 79 A4          STA IOUTU
2534 F10E 0E 7A A4          IPS2 ASL IBITL          ;SHIFT MSK TO NXT CHR POSITION
2535 F111 2E 7B A4          ROL IBITU
2536 F114 CA          DEX          ;DECR PTR INTO BUFFER
2537 F115 CA          DEX
2538 F116 10 D0          BPL IPS1          ;NOT END YET
2539 F118          ;SOLENOID PATTERN IS SET UP IN IOUTU,IOUTL
2540 F118 AD 79 A4          LDA IOUTU          ;LEFTMOST 2
2541 F11B 29 03          AND #*03          ;DISABLE FOR SEGMENTS
2542 F11D 8D 79 A4          STA IOUTU
2543 F120 60          RTS

```

LINE #	LOC	CODE	LINE
2545	F121		; SUBROUTINE INCP
2546	F121		; THIS SUBROUTINE IS USED TO UPDATE THE PRINTER VARIABLES
2547	F121		; TO POINT TO THE NEXT DOT POSITION TO BE PRINTED
2548	F121		; X REG IS USED TO POINT TO THE VARIABLE BLOCK OF
2549	F121		; BEING UPDATED
2550	F121		; ON EXIT X CONTAINS THE POINTER TO THE LAST CHARACTER IN
2551	F121		; THE PRINT BUFFER
2552	F121		; CONTENTS OF A,Y ON EXIT ARE ARBITRARY
2553	F121	BD 74 A4	INCP LDA IDIR,X ; EXAMINE DIRECTION(+ OR -)
2554	F124	10 1E	BPL OP03 ; DIRECTION = +
2555	F126		; **DIRECTION = -
2556	F126	BD 75 A4	LDA ICOL,X ; SEE WHAT THE COLUMN IS
2557	F129	F0 05	BEQ OP04 ; COLUMN = 0 SO END OF DIGIT
2558	F12B		; **COLUMN # 0 SO JUST DECREMENT COLUMN
2559	F12B	DE 75 A4	DEC ICOL,X
2560	F12E	10 33	BPL NEWCOL ; BRANCH ALWAYS
2561	F130		; **COLUMN = 0 SO SEE IF EVEN OR ODD DIGIT
2562	F130	BD 76 A4	OP04 LDA IOFFST,X
2563	F133	F0 0A	BEQ OP07 ; OFFSET = 0 SO DIRECTION CHANGE
2564	F135		; ***OFFSET = 1 SO MOVE TO RIGHT DIGIT
2565	F135	DE 76 A4	DEC IOFFST,X ; OFFSET (= 0 (LEFT CHARACTER)
2566	F138	A9 04	LDA #4 ; COLUMN (= 4
2567	F13A	9D 75 A4	STA ICOL,X
2568	F13D	10 24	BPL NEWCOL ; BRANCH ALWAYS
2569	F13F		; ***OFFSET = 0 SO CHANGE DIRECTION TO +
2570	F13F	FE 74 A4	OP07 INC IDIR,X ; DIRECTION (= \$00 (+)
2571	F142	10 1C	BPL NEWROW ; BRANCH ALWAYS
2572	F144		; *DIRECTION = +
2573	F144	BD 75 A4	OP03 LDA ICOL,X ; SEE IF LAST COLUMN IN DIGIT
2574	F147	C9 04	CMR #4
2575	F149	F0 05	BEQ OP05 ; COLUMN = 4 SO GO TO NEXT DIGIT
2576	F14B	FE 75 A4	INC ICOL,X ; JUST INCR COLUMN-NOT END OF DIGIT
2577	F14E	10 13	BPL NEWCOL ; BRANCH ALWAYS
2578	F150		; **AT COLUMN 4 -- SEE IF LEFT OR RIGHT DIGIT
2579	F150	BD 76 A4	OP05 LDA IOFFST,X
2580	F153	D0 08	BNE OP06 ; OFFSET # 0 SO RIGHT DIGIT
2581	F155	9D 75 A4	STA ICOL,X ; COLUMN (= 0
2582	F158	FE 76 A4	INC IOFFST,X ; OFFSET (= 1 (RIGHT CHARACTER)
2583	F15B	10 06	BPL NEWCOL ; BRANCH ALWAYS
2584	F15D		; ***OFFSET = 1 SO DIRECTION CHANGE
2585	F15D	DE 74 A4	OP06 DEC IDIR,X ; DIRECTION (= \$FF (-)
2587	F160		; START OF NEW PRINT ROW
2588	F160	1E 7C A4	NEWROW ASL IMASK,X ; UPDATE ROW MASK FOR DOT PATTERN
2589	F163		; START OF NEW PRINT COLUMN
2590	F163	A9 00	NEWCOL LDA #0 ; CLEAR OUTPUT PATTERN
2591	F165	9D 78 A4	STA IOUHL,X ; PATTERN FOR 8 RIGHT CHRS
2592	F168	9D 79 A4	STA IOUHU,X ; PATTERN FOR 2 LEFT SOLEN
2593	F16B	9D 7B A4	STA IBITU,X ; OUTPUT MSK FOR LEFTMOST SOLEN
2594	F16E	A9 01	LDA #1
2595	F170	9D 7A A4	STA IBITL,X ; OUTPUT MSK FOR RIGHTMOST SOLEN
2596	F173		; GET ADDRESS OF DOT PATTERN TABLE FOR NEXT COLUMN
2597	F173	BD 75 A4	LDA ICOL,X ; GET COLUMN NUMBER (0-4)
2598	F176	0A	ASL A ; *2, INDEX INTO TBL OF TBL ADDR
2599	F177	AB	TAY

LINE #	LOC	CODE	LINE
2600	F17B	B9 D7 F2	LDA MTBL,Y ; LSB OF ADDR OF TABLE
2601	F17B	9D 7D A4	STA JMP,X ; PTR TO TBL WITH DOT PATTERNS
2602	F17E	B9 D8 F2	LDA MTBL+1,Y ; MSB OF TABLE ADDRESS
2603	F181	9D 7E A4	STA JMP+1,X
2604	F184	A9 12	LDA #18 ; COMPUTE INDEX INTO PRNTR BUFFER
2605	F186	1D 76 A4	ORA IOFFST,X ; +1 IF RIGHT CHR
2606	F189	AA	TAX
2607	F18A	60	RTS
2609	F18B		;;
2610	F18B		; OUTPUT ACC TO TAPE BUFFER SUBROUTINE
2611	F18B		; & WHEN FULL OUTPUT BUFF TO TAPE
2612	F18B		; IF INFLG=OUTFLG= T USE TWO BUFFERS
2613	F18B		; OTHERWISE USE SAME BUFFER FOR INPUT
2614	F18B		; AND OUTPUT (MONIT BUFFER)
2615	F18B	20 9E EB	TORYTE JSR PHXY ; SAVE X
2616	F18E	AE 37 A4	LUX TAPTR2 ; TAPE BUFFER POINTER FOR OUTPUT
2617	F191	20 0F F2	JSR BKCK2 ; STORE IN BUFFER
2618	F194	E8	INX
2619	F195	8E 37 A4	STX TAPTR2 ; FOR NEXT
2620	F198	E0 50	CPX #80 ; BUFFER FULL?
2621	F19A	D0 32	BNE TABY3 ; NO, GO BACK
2622	F19C		; OUTPUT A BLOCK FROM BUFFER TO TAPE
2623	F19C	20 E7 F1	JSR BKCKSM ; COMPUTE BLOCK CHECKSUM
2624	F19F	20 1D F2	JSR TAOSST ; SET TAPE FOR OUTPUT
2625	F1A2	A9 23	LDA #'# ; CHAR FOR BEGINNING
2626	F1A4	20 4A F2	JSR OUTTAP ; OF BLOCK
2627	F1A7		; OUTPUT CHRS FROM ACTIVE BUFFER
2628	F1A7	20 D2 F1	TABY2 JSR CKBUFF ; LOAD CHR FROM ACTIVE BUFFER
2629	F1AA	20 4A F2	JSR OUTTAP ; FROM BUFFER
2630	F1AD	E8	INX
2631	F1AE	E0 53	CPX #83 ; 2 BLOCK CKSUM CHR + 1 EXTRA CHR
2632	F1B0	D0 F5	BNE TABY2 ; OTHERWISE ERROR
2633	F1B2	AD 00 AB	LDA DRB
2634	F1B5	29 CF	AND #*CF ; TURN TAPES OFF PB5,PB4
2635	F1B7	BD 00 AB	STA DRB
2636	F1BA	58	CLI ; ENABLE INTERRUPT
2637	F1BB	A9 00	LDA #0
2638	F1BD	BD 37 A4	STA TAPTR2 ; CLR TAPE BUFF PTR
2639	F1C0	A9 00	LDA #111 ; RESET FREE RUNNING TO 1 SHOT
2640	F1C2	BD 0B AB	STA ACR
2641	F1C5	20 9A FF	JSR PAT22 ; ADD 1 TO BLK COUNT & OUTPUT
2642	F1C8	AD 68 01	LDA BLKO ; PUT BLK CNT IN FIRST LOC (TABUFF)
2643	F1CB	20 8B F1	JSR TORYTE
2644	F1CE	20 AC EB	TABY3 JSR PLXY
2645	F1D1	60	RTS
2647	F1D2		; CHCK ACTIVE BUFFER AND LOAD A CHR
2648	F1D2		; CARRY=0 IF ONLY 1 BUFFER ,C=1 IF 2 BUFFERS
2649	F1D2	AD 12 A4	CKBUFF LDA INFLG
2650	F1D5	CD 13 A4	CMR OUTFLG
2651	F1D8	D0 08	BNE CRUFF1
2652	F1DA	C9 54	CMR #'T ; SEE IF INFLG=OUTFLG = T
2653	F1DC	D0 04	BNE CBUFF1

LINE #	LOC	CODE	LINE
2654	F1DE	39	SEC
2655	F1DF	B5 AD	LDA TABUF2,X
2656	F1E1	60	RTS
2657	F1E2	18	CRUFF1 CLC
2658	F1E3	BD 16 Q1	LDA TABUFF,X
2659	F1E6	60	RTS
2661	F1E7		; COMPUTE BLOCK CHECKSUM & PUT IT
2662	F1E7		; AT THE END OF ACTIVE BUFFER
2663	F1E7	A9 00	BKCKSM LDA #0
2664	F1E9	BD 66 01	STA TABUFF+80
2665	F1EC	BD 67 01	STA TABUFF+81
2666	F1EF	A2 4F	LIX #79
2667	F1F1	20 D2 F1	BKCK1 JSR CKBUFF
2668	F1F4	18	CLC
2669	F1F5	6D 66 01	ADC TABUFF+80
2670	F1FB	BD 66 01	STA TABUFF+80
2671	F1FB	90 03	BCC #+5
2672	F1FD	EE 67 01	INC TABUFF+B1
2673	F200	CA	DEX
2674	F201	10 EE	BPL BKCK1
2675	F203	A2 50	LIX #80
2676	F205	AD 66 01	LDA TABUFF+80
2677	F208	20 0F F2	JSR BKCK2
2678	F20B	EB	INX
2679	F20C	AD 67 01	LDA TABUFF+B1
2680	F20F	48	PHA
2681	F210	20 D2 F1	JSR CKBUFF
2682	F213	68	PLA
2683	F214	80 04	BCS BKCK3
2684	F216	9D 16 01	STA TABUFF,X
2685	F219	60	RTS
2686	F21A	95 AD	BKCK3 STA TABUF2,X
2687	F21C	60	RTS
2689	F21D		; SET TAPE (1 OR 2) FOR OUTPUT
2690	F21D	20 C0 F2	TAOS1 JSR SETSPD
2691	F220	AD 35 A4	LDA TAFOUT
2692	F223	20 1C EE	JSR T10SET
2693	F226	A9 EC	LDA #DATOUT+MOFF
2694	F228	BD 0C AB	STA PCR
2695	F22E	A9 C0	LDA #T1FR
2696	F22D	BD 0B AB	STA ACR
2697	F230	A9 00	LDA #00
2698	F232	BD 05 AB	STA T1CH
2699	F235	AE 07 A4	LIX GAP
2700	F238	A9 16	TAOS1 LDA #16
2701	F23A	20 4A F2	JSR OUTTAP
2702	F23D	20 4A F2	JSR OUTTAP
2703	F240	20 4A F2	JSR OUTTAP
2704	F243	20 4A F2	JSR OUTTAP
2705	F246	CA	DEX
2706	F247	10 EF	BNE TAOS1
2707	F249	60	RTS

LINE #	LOC	CODE	LINE
2709	F24A		; OUTPUT ACC TO TAPE
2710	F24A	BE 2D A4	OUTTAP STX CFIY+3
2711	F24D	A0 07	LIX #807
2712	F24F	8C 27 A4	STY STIY
2713	F252	AE 08 A4	LIX TSPEED
2714	F255	30 39	BMI OUTTA1
2715	F257	48	PHA
2716	F258	A0 02	TRY LIX #2
2717	F25A	BC 28 A4	STY STIY+1
2718	F25D	BE 0A A4	ZON LIX NPUL,Y
2719	F260	48	PHA
2720	F261	B9 0B A4	ZON1 LDA TIMG,Y
2721	F264	BD 06 AB	STA T1LL
2722	F267	A9 00	LDA #0
2723	F269	8D 07 AB	STA T1LH
2724	F26C	2C 0D AB	ZON2 BIT IFR
2725	F26F	50 FR	BVC ZON2
2726	F271	AD 04 AB	LDA T1L
2727	F274	CA	DEX
2728	F275	D0 EA	BNE ZON1
2729	F277	68	PLA
2730	F278	CE 28 A4	DEC STIY+1
2731	F27B	F0 05	REQ SETZ
2732	F27D	30 07	BMI ROUT
2733	F27F	4A	LSR A
2734	F280	90 DB	BCC ZON
2735	F282	A0 00	SETZ LIX #0
2736	F284	F0 D7	REQ ZON
2737	F286	CE 27 A4	ROUT DEC STIY
2738	F289	10 CD	BPL TRY
2739	F28B	68	ROUT1 PLA
2740	F28C	AE 2D A4	LIX CFIY+3
2741	F28F	60	RTS
2743	F290		; OUTPUT HALF PULSE FOR 0 (1200 HZ) &
2744	F290		; TWO HALF PULSES FOR 1 (2400 HZ) (00 TSPEED)
2745	F290	48	OUTTA1 PHA
2746	F291	BD 28 A4	STA STIY+1
2747	F294	A2 02	LIX #2
2748	F296	A9 D0	LDA #D0
2749	F298	BD 06 AB	STA T1LL
2750	F29B	A9 00	LDA #00
2751	F29D	BD 07 AB	STA T1LH
2752	F2A0	20 BC FF	JSR PATC25
2753	F2A3	4E 28 A4	LSR STIY+1
2754	F2A6	B0 0A	BCS OUTTA3
2755	F2AB	A9 0A	LDA #A0
2756	F2AA	BD 06 AB	STA T1LL
2757	F2AD	A9 01	LDA #A01
2758	F2AF	BD 07 AB	STA T1LH
2759	F2B2	20 BC FF	OUTTA3 JSR PATC25
2760	F2B5	CA	DEX
2761	F2B6	10 FA	BPL OUTTA3
2762	F2BB	88	DEY
2763	F2B9	10 D9	BPL OUTTA2

LINE #	LOC	CODE	LINE
2764	F2BB	4C 8B F2	JMP ROUT1 ;RESTORE REGS
2765	F2BE	EA	NOP
2766	F2BF	EA	NOP
2768	F2C0		;SET SPEED FROM NORMAL TO 3 TIMES NORMAL
2769	F2C0	AD 08 A4	SETSPD LDA TSPEED ;SPEED FLG
2770	F2C3	6A	ROR A ;NORMAL OR 3* NORM
2771	F2C4	A9 0C	LDA #12
2772	F2C6	90 02	BCC SETSP1
2773	F2C8	A9 04	LDA #4
2774	F2CA	8D 0A A4	SETSP1 STA NPUL
2775	F2CD	A9 12	LDA #18
2776	F2CF	90 02	BCC SETSP2
2777	F2D1	A9 06	LDA #6
2778	F2D3	8D 0C A4	SETSP2 STA TIMG+1
2779	F2D6	60	RTS
2780	F2D7		.FILE A3/2

LINE #	LOC	CODE	LINE
2782	F2D7		; ADDRESS TABLE FOR EACH PRINT COLUMN.
2783	F2D7		; EACH TBL CONTAINS DOT PATTERNS FOR 1 OF THE 5 COLUMNS.
2784	F2D7		; DATA ARE STORED WITH EACH BYTE DEFINING ONE COLUMN...
2785	F2D7		; OF A CHARACTER, WITH THE TOP DOT CORRESPONDING TO THE...
2786	F2D7		; LSB IN THE BYTE
2787	F2D7	E1 F2	MTBL .WOR COL0,COL1,COL2,COL3,COL4
2787	F2D9	21 F3	
2787	F2DB	61 F3	
2787	F2DD	A1 F3	
2787	F2DF	E1 F3	
2789	F2E1		;DOT PATTERNS FOR COLUMN ZERO (LEFTMOST COLUMN)
2790	F2E1	3E	COL0 .BYT \$3E,\$7E,\$7F,\$3E,\$7F,\$7F,\$7F,\$3E @ -- G
2790	F2E2	7E	
2790	F2E3	7F	
2790	F2E4	3E	
2790	F2E5	7F	
2790	F2E6	7F	
2790	F2E7	7F	
2790	F2E8	3E	
2791	F2E9	7F	.BYT \$7F,\$00,\$20,\$7F,\$7F,\$7F,\$7F,\$3E H -- 0
2791	F2EA	00	
2791	F2EB	20	
2791	F2EC	7F	
2791	F2ED	7F	
2791	F2EE	7F	
2791	F2EF	7F	
2791	F2F0	3E	
2792	F2F1	7F	.BYT \$7F,\$3E,\$7F,\$46,\$01,\$3F,\$07,\$7F F -- W
2792	F2F2	3E	
2792	F2F3	7F	
2792	F2F4	46	
2792	F2F5	01	
2792	F2F6	3F	
2792	F2F7	07	
2792	F2F8	7F	
2793	F2F9	63	.BYT \$63,\$07,\$61,\$7F,\$03,\$00,\$02,\$40 X -- C
2793	F2FA	07	
2793	F2FB	61	
2793	F2FC	7F	
2793	F2FD	03	
2793	F2FE	00	
2793	F2FF	02	
2793	F300	40	
2794	F301	00	.BYT \$00,\$00,\$00,\$14,\$24,\$63,\$60,\$00 -- /
2794	F302	00	
2794	F303	00	
2794	F304	14	
2794	F305	24	
2794	F306	63	
2794	F307	60	
2794	F308	00	
2795	F309	00	.BYT \$00,\$00,\$14,\$08,\$40,\$08,\$40,\$60 ( -- /
2795	F30A	00	
2795	F30B	14	

## DOT PATTERNS

PA00-J001A.....PAGE 0058

LINE #	LOC	CODE	LINE
2795	F30C	08	
2795	F30D	40	
2795	F30E	08	
2795	F30F	40	
2795	F310	60	
2796	F311	3E	.BYT \$3E,\$44,\$62,\$41,\$18,\$27,\$3C,\$01 0 -- 7
2796	F312	44	
2796	F313	62	
2796	F314	41	
2796	F315	18	
2796	F316	27	
2796	F317	3C	
2796	F318	01	
2797	F319	36	.BYT \$36,\$46,\$00,\$40,\$08,\$14,\$41,\$02 8 -- 7
2797	F31A	46	
2797	F31B	00	
2797	F31C	40	
2797	F31D	08	
2797	F31E	14	
2797	F31F	41	
2797	F320	02	
2799	F321		.DOT PATTERNS FOR COLUMN 1
2800	F321	41	COL1 .BYT \$41,\$09,\$49,\$41,\$41,\$49,\$09,\$41 e -- G
2800	F322	09	
2800	F323	49	
2800	F324	41	
2800	F325	41	
2800	F326	49	
2800	F327	09	
2800	F328	41	
2801	F329	08	.BYT \$08,\$41,\$40,\$08,\$40,\$02,\$06,\$41 H -- D
2801	F32A	41	
2801	F32B	40	
2801	F32C	08	
2801	F32D	40	
2801	F32E	02	
2801	F32F	06	
2801	F330	41	
2802	F331	09	.BYT \$09,\$41,\$09,\$49,\$01,\$40,\$18,\$20 P -- W
2802	F332	41	
2802	F333	09	
2802	F334	49	
2802	F335	01	
2802	F336	40	
2802	F337	18	
2802	F338	20	
2803	F339	14	.BYT \$14,\$08,\$51,\$41,\$04,\$00,\$01,\$40 X -- C
2803	F33A	08	
2803	F33B	51	
2803	F33C	41	
2803	F33D	04	
2803	F33E	00	
2803	F33F	01	
2803	F340	40	

## DOT PATTERNS

PA00-J001A.....PAGE 0059

LINE #	LOC	CODE	LINE
2804	F341	00	.BYT \$00,\$00,\$07,\$7F,\$2A,\$13,\$4E,\$04 -- /
2804	F342	00	
2804	F343	07	
2804	F344	7F	
2804	F345	2A	
2804	F346	13	
2804	F347	4E	
2804	F348	04	
2805	F349	1C	.BYT \$1C,\$41,\$08,\$08,\$30,\$08,\$00,\$10 ( -- /
2805	F34A	41	
2805	F34B	08	
2805	F34C	08	
2805	F34D	30	
2805	F34E	08	
2805	F34F	00	
2805	F350	10	
2806	F351	51	.BYT \$51,\$42,\$51,\$41,\$14,\$45,\$4A,\$71 0 -- 7
2806	F352	42	
2806	F353	51	
2806	F354	41	
2806	F355	14	
2806	F356	45	
2806	F357	4A	
2806	F358	71	
2807	F359	49	.BYT \$49,\$49,\$00,\$34,\$14,\$14,\$41,\$01 8 -- 7
2807	F35A	49	
2807	F35B	00	
2807	F35C	34	
2807	F35D	14	
2807	F35E	14	
2807	F35F	41	
2807	F360	01	
2809	F361		.DOT PATTERNS FOR COLUMN 2
2810	F361	5D	COL2 .BYT \$5D,\$09,\$49,\$41,\$41,\$49,\$09,\$41 e -- G
2810	F362	09	
2810	F363	49	
2810	F364	41	
2810	F365	41	
2810	F366	49	
2810	F367	09	
2810	F368	41	
2811	F369	08	.BYT \$08,\$7F,\$41,\$14,\$40,\$0C,\$08,\$41 H -- D
2811	F36A	7F	
2811	F36B	41	
2811	F36C	14	
2811	F36D	40	
2811	F36E	0C	
2811	F36F	08	
2811	F370	41	
2812	F371	09	.BYT \$09,\$51,\$19,\$49,\$7F,\$40,\$60,\$18 P -- W
2812	F372	51	
2812	F373	19	
2812	F374	49	
2812	F375	7F	

LINE #	LOC	CODE	LINE
2812	F376	40	
2812	F377	60	
2812	F378	18	
2813	F379	08	.BYT \$0B,\$7B,\$49,\$41,\$0B,\$41,\$01,\$40 X -- C
2813	F37A	78	
2813	F37B	49	
2813	F37C	41	
2813	F37D	0B	
2813	F37E	41	
2813	F37F	01	
2813	F380	40	
2814	F381	00	.BYT \$00,\$4F,\$00,\$14,\$7F,\$0B,\$59,\$02 --- /
2814	F382	4F	
2814	F383	00	
2814	F384	14	
2814	F385	7F	
2814	F386	0B	
2814	F387	59	
2814	F388	02	
2815	F389	22	.BYT \$22,\$22,\$3E,\$3E,\$00,\$0B,\$00,\$0B ( -- /
2815	F38A	22	
2815	F38B	3E	
2815	F38C	3E	
2815	F38D	00	
2815	F38E	0B	
2815	F38F	00	
2815	F390	0B	
2816	F391	49	.BYT \$49,\$7F,\$51,\$49,\$12,\$45,\$49,\$09 0 -- 7
2816	F392	7F	
2816	F393	51	
2816	F394	49	
2816	F395	12	
2816	F396	45	
2816	F397	49	
2816	F398	09	
2817	F399	49	.BYT \$49,\$49,\$44,\$00,\$22,\$14,\$22,\$51 8 -- ?
2817	F39A	49	
2817	F39B	44	
2817	F39C	00	
2817	F39D	22	
2817	F39E	14	
2817	F39F	22	
2817	F3A0	51	
2819	F3A1		; DOT PATTERNS FOR COLUMN 3
2820	F3A1	55	COL3 .BYT \$55,\$09,\$49,\$41,\$22,\$49,\$09,\$49 @ -- 6
2820	F3A2	09	
2820	F3A3	49	
2820	F3A4	41	
2820	F3A5	22	
2820	F3A6	49	
2820	F3A7	09	
2820	F3A8	49	
2821	F3A9	0B	.BYT \$0B,\$41,\$3F,\$22,\$40,\$02,\$30,\$41 H -- 0
2821	F3AA	41	

LINE #	LOC	CODE	LINE
2821	F3AB	3F	
2821	F3AC	22	
2821	F3AD	40	
2821	F3AE	02	
2821	F3AF	30	
2821	F3B0	41	
2822	F3B1	09	.BYT \$09,\$21,\$29,\$49,\$01,\$40,\$18,\$20 P -- W
2822	F3B2	21	
2822	F3B3	29	
2822	F3B4	49	
2822	F3B5	01	
2822	F3B6	40	
2822	F3B7	18	
2822	F3B8	20	
2823	F3B9	14	.BYT \$14,\$0B,\$45,\$00,\$10,\$41,\$01,\$40 X -- C
2823	F3BA	0B	
2823	F3BB	45	
2823	F3BC	00	
2823	F3BD	10	
2823	F3BE	41	
2823	F3BF	01	
2823	F3C0	40	
2824	F3C1	00	.BYT \$00,\$00,\$07,\$7F,\$2A,\$64,\$26,\$01 --- /
2824	F3C2	00	
2824	F3C3	07	
2824	F3C4	7F	
2824	F3C5	2A	
2824	F3C6	64	
2824	F3C7	26	
2824	F3C8	01	
2825	F3C9	41	.BYT \$41,\$1C,\$0B,\$0B,\$00,\$0B,\$00,\$04 ( -- /
2825	F3CA	1C	
2825	F3CB	0B	
2825	F3CC	0B	
2825	F3CD	00	
2825	F3CE	0B	
2825	F3CF	00	
2825	F3D0	04	
2826	F3D1	45	.BYT \$45,\$40,\$49,\$55,\$7F,\$45,\$49,\$05 0 -- 7
2826	F3D2	40	
2826	F3D3	49	
2826	F3D4	55	
2826	F3D5	7F	
2826	F3D6	45	
2826	F3D7	49	
2826	F3D8	05	
2827	F3D9	49	.BYT \$49,\$29,\$00,\$00,\$41,\$14,\$14,\$09 8 -- ?
2827	F3DA	29	
2827	F3DB	00	
2827	F3DC	00	
2827	F3DD	41	
2827	F3DE	14	
2827	F3DF	14	
2827	F3E0	09	
2828	F3E1		; DOT PATTERNS FOR COLUMN 4

## DOT PATTERNS

PA00-J001A.....PAGE 0062

LINE #	LOC	CODE	LINE
2829	F3E1	1E	COL4 .BYT \$1E,\$7E,\$36,\$22,\$1C,\$41,\$01,\$7A @ -- G
2829	F3E2	7E	
2829	F3E3	36	
2829	F3E4	22	
2829	F3E5	1C	
2829	F3E6	41	
2829	F3E7	01	
2829	F3E8	7A	
2830	F3E9	7F	.BYT \$7F,\$00,\$01,\$41,\$40,\$7F,\$7F,\$3E H -- 0
2830	F3EA	00	
2830	F3EB	01	
2830	F3EC	41	
2830	F3ED	40	
2830	F3EE	7F	
2830	F3EF	7F	
2830	F3F0	3E	
2831	F3F1	06	.BYT \$06,\$5E,\$46,\$31,\$01,\$3F,\$07,\$7F P -- W
2831	F3F2	5E	
2831	F3F3	46	
2831	F3F4	31	
2831	F3F5	01	
2831	F3F6	3F	
2831	F3F7	07	
2831	F3F8	7F	
2832	F3F9	63	.BYT \$63,\$07,\$43,\$00,\$60,\$7F,\$02,\$40 X -- C
2832	F3FA	07	
2832	F3FB	43	
2832	F3FC	00	
2832	F3FD	60	
2832	F3FE	7F	
2832	F3FF	02	
2832	F400	40	
2833	F401	00	.BYT \$00,\$00,\$00,\$14,\$12,\$63,\$50,\$00 -- /
2833	F402	00	
2833	F403	00	
2833	F404	14	
2833	F405	12	
2833	F406	63	
2833	F407	50	
2833	F408	00	
2834	F409	00	.BYT \$00,\$00,\$14,\$08,\$00,\$08,\$00,\$03 < -- /
2834	F40A	00	
2834	F40B	14	
2834	F40C	08	
2834	F40D	00	
2834	F40E	08	
2834	F40F	00	
2834	F410	03	
2835	F411	3E	.BYT \$3E,\$40,\$46,\$22,\$10,\$39,\$31,\$03 0 -- 7
2835	F412	40	
2835	F413	46	
2835	F414	22	
2835	F415	10	
2835	F416	39	
2835	F417	31	

## DOT PATTERNS

PA00-J001A.....PAGE 0063

LINE #	LOC	CODE	LINE
2835	F418	03	
2836	F419	36	
2836	F41A	1E	.BYT \$36,\$1E,\$00,\$00,\$41,\$14,\$08,\$06 B -- ?
2836	F41B	00	
2836	F41C	00	
2836	F41D	41	
2836	F41E	14	
2836	F41F	08	
2836	F420	06	
2838	F421		ASCII CHARACTERS FOR KB
2839	F421	20	ROW1 .BYT \$20,\$0B,\$00,\$0D,\$00,\$00,\$00,\$00
2839	F422	08	
2839	F423	00	
2839	F424	0D	
2839	F425	00	
2839	F426	00	
2839	F427	00	
2839	F428	00	
2840	F429	00	ROW2 .BYT \$00,\$60,'\'', \$00,\$00,\$00,\$7F,\$00
2840	F42A	60	
2840	F42B	5C	
2840	F42C	00	
2840	F42D	00	
2840	F42E	00	
2840	F42F	7F	
2840	F430	00	
2841	F431	2E 4C	ROW3 .BYT 'LP=0;/'
2842	F439	4D 4A	ROW4 .BYT 'MJ109BK,'
2843	F441	42 47	ROW5 .BYT 'BGYU76HN'
2844	F449	43 44	ROW6 .BYT 'CDRT54FV'
2845	F451	5A 41	ROW7 .BYT 'ZANE32SX'
2846	F459	00	ROW8 .BYT \$00,\$00,\$1B,'01',\$5E,'JL'
2846	F45A	00	
2846	F45B	1B	
2846	F45C	51 31	
2846	F45E	5E	
2846	F45F	5D 5B	



LINE #	LOC	CODE	LINE
2848	F461		;DISASSEMBLE INSTRUCTION AND SHOW REGS IS REGF SET
2849	F461	AD 0E A4	REGQ LDA REGF ;GET FLAG
2850	F464	F0 06	BEQ DISASM
2851	F466	20 32 E2	JSR REG1 ;SHOW THE SIX REGS
2852	F469	20 24 EA	JSR CRCK ;CCR)
2854	F46C	20 45 F5	DISASM JSR PRBL2
2855	F46F	20 3C F5	JSR PRFC ;OUTPUT PRG COUNTR
2856	F472	A0 00	LDY #0
2857	F474	20 56 EB	JSR PCLLD
2858	F477	A8	TAY
2859	F478	4A	LSR A
2860	F479	90 08	BCC IEVEN
2861	F47B	4A	LSR A
2862	F47C	B0 17	BCS ERR
2863	F47E	C9 22	CHP ##22
2864	F480	F0 13	BEQ ERR
2865	F482	29 07	AND #7
2866	F484	09 80	ORA ##80
2867	F486	4A	IEVEN LSR A
2868	F487	AA	TAX
2869	F48B	B0 5B F5	LDA MODE,X
2870	F48B	B0 04	BCS RTMODE
2871	F48D	4A	LSR A
2872	F48E	4A	LSR A
2873	F48F	4A	LSR A
2874	F490	4A	LSR A
2875	F491	29 0F	RTMODE AND ##F
2876	F493	D0 04	BNE GETFMT
2877	F495	A0 80	ERR LDY ##80
2878	F497	A9 00	LDA #0
2879	F499	AA	GETFMT TAX
2880	F49A	B0 9F F5	LDA MODE2,X
2881	F49D	B0 16 01	STA FORMA
2882	F4A0	29 03	AND #3
2883	F4A2	85 EA	STA LENGTH
2884	F4A4	98	TYA ;OPCODE
2885	F4A5	29 8F	AND ##8F
2886	F4A7	AA	TAX
2887	F4A8	98	TYA ;OPCODE IN A AGAIN
2888	F4A9	A0 03	LDY #3
2889	F4AB	E0 8A	CPX ##8A
2890	F4AD	F0 0B	BEQ MNNDX3
2891	F4AF	4A	MNNDX1 LSR A
2892	F4B0	90 0B	BCC MNNDX3
2893	F4B2	4A	LSR A
2894	F4B3	4A	MNNDX2 LSR A
2895	F4B4	09 20	ORA ##20
2896	F4B6	B8	DEY
2897	F4B7	D0 FA	BNE MNNIX2
2898	F4B9	CB	INY
2899	F4BA	B8	MNNIX3 DEY
2900	F4BB	D0 F2	BNE MNNIX1
2901	F4BD	48	PHA ;SAVE MNEMONIC TABLE INDEX
2902	F4BE	20 56 EB	JSR PCLLD

LINE #	LOC	CODE	LINE
2903	F4C1	20 46 EA	JSR NUMA
2904	F4C4	20 45 F5	JSR PRBL2 ;PRINT LAST BLANK
2905	F4C7	68	PLA
2906	F4C8	AB	TAY
2907	F4C9	B9 B9 F5	LDA MNEML,Y
2908	F4CC	B0 17 01	STA LMNEM
2909	F4CF	B9 F9 F5	LDA MNEMR,Y
2910	F4D2	B0 18 01	STA RMNEM
2911	F4D5	A2 03	LDX #3 ;MUST BE
2912	F4D7	A9 00	PRMN1 LDA #0
2913	F4D9	A0 05	LDY #5
2914	F4DB	0E 18 01	PRMN2 ASL RMNEM
2915	F4DE	2E 17 01	ROL LMNEM
2916	F4E1	2A	ROL A
2917	F4E2	88	DEY
2918	F4E3	D0 F6	BNE PRMN2
2919	F4E5	69 BF	ADC ##BF ;ADD '?' OFFSET
2920	F4E7	20 BC E9	JSR OUTALL
2921	F4EA	CA	DEX
2922	F4EB	D0 EA	BNE PRMN1
2923	F4ED	20 45 F5	JSR PRBL2
2924	F4F0	A2 06	LDX #6
2925	F4F2	A9 00	LDA #0
2926	F4F4	B0 29 A4	STA STIY+2 ;FLAG
2927	F4F7	E0 03	PRADR1 CPX #3
2928	F4F9	D0 1E	BNE PRADR3 ;IF X=3 PRINT ADDR VALUE
2929	F4FB	A4 EA	LDY LENGTH
2930	F4FD	F0 1A	BEQ PRADR3 ;1 BYTE INSTR
2931	F4FF	AD 16 01	PRADR2 LDA FORMA
2932	F502	C9 E8	CMF ##E8 ;RELATIVE ADDRESSING
2933	F504	20 56 EB	JSR PCLLD
2934	F507	B0 27	BCS RELADR
2935	F509		;SEE IF SYMBOL
2936	F509	48	PHA
2937	F50A	AD 29 A4	LDA STIY+2
2938	F50D	D0 03	BNE MR11A
2939	F50F	EE 29 A4	INC STIY+2 ;SHOW WE WERE HERE
2941	F512	68	MR11A PLA
2942	F513	20 46 EA	JSR NUMA
2943	F516	88	DEY
2944	F517	D0 E6	BNE PRADR2
2945	F519	0E 16 01	PRADR3 ASL FORMA
2946	F51C	90 0E	BCC PRADR4
2947	F51E	B0 AC F5	LDA CHAR1-1,X
2948	F521	20 BC E9	JSR OUTALL
2949	F524	B0 B2 F5	LDA CHAR2-1,X
2950	F527	F0 03	BEQ PRADR4
2951	F529	20 BC E9	JSR OUTALL
2952	F52C	CA	PRADR4 DEX
2953	F52D	D0 C8	BNE PRADR1
2954	F52F	60	RTS
2955	F530	20 4D F5	RELADR JSR PCADJ3
2956	F533	AA	TAX
2957	F534	EB	INX

LINE #	LOC	CODE	LINE
2958	F535	D0 01	BNE PRNTYX
2959	F537	C8	INY
2960	F538	98	PRNTYX TYA
2961	F539	4C 42 EA	JMP WRAX ;PRINT A &X
2962	F53C	AD 26 A4	PRPC LDA SAVPC+1 ;PRINT PC
2963	F53F	AE 25 A4	LIX SAVPC
2964	F542	20 42 EA	JSR WRAX
2965	F545	A9 20	PRBL2 LDA #*20
2966	F547	4C BC E9	JMP OUTALL
2967	F54A	A5 EA	LDA LENGTH
2968	F54C	38	SEC
2969	F54D	AC 26 A4	PCAIJ3 LDY SAVPC+1 ;PRG CNTR HIGH
2970	F550	AA	TAX
2971	F551	10 01	RPL PCAIJ4
2972	F553	88	DEY
2973	F554	6D 25 A4	PCAIJ4 ADC SAVPC ;PRG CNTR LOW
2974	F557	90 01	BCC RTS1
2975	F559	C8	INY
2976	F55A	60	RTS1 RTS
2978	F55B	40	MODE .BYT \$40,2,\$45,3,\$D0,8,\$40,9
2978	F55C	02	
2978	F55D	45	
2978	F55E	03	
2978	F55F	D0	
2978	F560	08	
2978	F561	40	
2978	F562	09	
2979	F563	30	.BYT \$30,\$22,\$45,\$33,\$D0,8,\$40,9
2979	F564	22	
2979	F565	45	
2979	F566	33	
2979	F567	D0	
2979	F568	08	
2979	F569	40	
2979	F56A	09	
2980	F56B	40	.BYT \$40,2,\$45,\$33,\$D0,8,\$40,9
2980	F56C	02	
2980	F56D	45	
2980	F56E	33	
2980	F56F	D0	
2980	F570	08	
2980	F571	40	
2980	F572	09	
2981	F573	40	.BYT \$40,2,\$45,\$B3,\$D0,8,\$40,9
2981	F574	02	
2981	F575	45	
2981	F576	B3	
2981	F577	D0	
2981	F578	08	
2981	F579	40	
2981	F57A	09	
2982	F57B	00	.BYT 0,\$22,\$44,\$33,\$D0,\$BC,\$44,0
2982	F57C	22	

LINE #	LOC	CODE	LINE
2982	F57D	44	
2982	F57E	33	
2982	F57F	D0	
2982	F580	8C	
2982	F581	44	
2982	F582	00	
2983	F583	11	.BYT \$11,\$22,\$44,\$33,\$D0,\$BC,\$44,\$9A
2983	F584	22	
2983	F585	44	
2983	F586	33	
2983	F587	D0	
2983	F588	8C	
2983	F589	44	
2983	F58A	9A	
2984	F58B	10	.BYT \$10,\$22,\$44,\$33
2984	F58C	22	
2984	F58D	44	
2984	F58E	33	
2985	F58F	D0	.BYT \$D0,8,\$40,9
2985	F590	08	
2985	F591	40	
2985	F592	09	
2986	F593	10	.BYT \$10,\$22,\$44,\$33,\$D0,8,\$40,9
2986	F594	22	
2986	F595	44	
2986	F596	33	
2986	F597	D0	
2986	F598	08	
2986	F599	40	
2986	F59A	09	
2987	F59B	62	.BYT \$62,\$13,\$78,\$A9
2987	F59C	13	
2987	F59D	78	
2987	F59E	A9	
2989	F59F	00	MODE2 .BYT 0,\$21,1,2,0,\$80,\$59,\$4D
2989	F5A0	21	
2989	F5A1	01	
2989	F5A2	02	
2989	F5A3	00	
2989	F5A4	80	
2989	F5A5	59	
2989	F5A6	4D	
2990	F5A7	11	.BYT \$11,\$12,6,\$4A,5,\$1D
2990	F5A8	12	
2990	F5A9	06	
2990	F5AA	4A	
2990	F5AB	05	
2990	F5AC	1D	
2992	F5AD	2C	CHAR1 .BYT ',',\$29,',\$(',' ,'
2992	F5AE	29	
2992	F5AF	2C 23 28	
2992	F5B2	2E	
2993	F5B3	59	CHAR2 .BYT 'Y',0,'X',0,0,'A'

LINE #	LOC	CODE	LINE
2993	F5B4	00	
2993	F5B5	5B	
2993	F5B6	00	
2993	F5B7	00	
2993	F5B8	41	
2995	F5B9	1C	MNEML .BYT \$1C, \$8A, \$1C, \$23, \$5D, \$8B, \$1B
2995	F5BA	8A	
2995	F5BB	1C	
2995	F5BC	23	
2995	F5BD	5D	
2995	F5BE	8B	
2995	F5BF	1B	
2996	F5C0	A1	.BYT \$A1
2997	F5C1	9D	.BYT \$9D, \$8A, \$1D, \$23, \$9D, \$8B, \$1D, \$A1
2997	F5C2	8A	
2997	F5C3	1D	
2997	F5C4	23	
2997	F5C5	9D	
2997	F5C6	8B	
2997	F5C7	1D	
2997	F5C8	A1	
2998	F5C9	00	.BYT 0, \$29, \$19, \$AE, \$69, \$AB, \$19, \$23
2998	F5CA	29	
2998	F5CB	19	
2998	F5CC	AE	
2998	F5CD	69	
2998	F5CE	AB	
2998	F5CF	19	
2998	F5D0	23	
2999	F5D1	24	.BYT \$24, \$53, \$1B, \$23, \$24, \$53, \$19, \$A1
2999	F5D2	53	
2999	F5D3	1B	
2999	F5D4	23	
2999	F5D5	24	
2999	F5D6	53	
2999	F5D7	19	
2999	F5D8	A1	
3000	F5D9	00	.BYT 0, \$1A, \$5B, \$5B, \$A5, \$69, \$24, \$24
3000	F5DA	1A	
3000	F5DB	5B	
3000	F5DC	5B	
3000	F5DD	A5	
3000	F5DE	69	
3000	F5DF	24	
3000	F5E0	24	
3001	F5E1	AE	.BYT \$AE, \$AE, \$AB, \$AD, \$29, 0, \$7C, 0
3001	F5E2	AE	
3001	F5E3	AB	
3001	F5E4	AD	
3001	F5E5	29	
3001	F5E6	00	
3001	F5E7	7C	
3001	F5E8	00	
3002	F5E9	15	.BYT \$15, \$9C, \$6D, \$9C, \$A5, \$69, \$29, \$53

LINE #	LOC	CODE	LINE
3002	F5EA	9C	
3002	F5EB	6D	
3002	F5EC	9C	
3002	F5ED	A5	
3002	F5EE	69	
3002	F5EF	29	
3002	F5F0	53	
3003	F5F1	84	.BYT \$84, \$13, \$34, \$11, \$A5, \$69, \$23, \$A0
3003	F5F2	13	
3003	F5F3	34	
3003	F5F4	11	
3003	F5F5	A5	
3003	F5F6	69	
3003	F5F7	23	
3003	F5F8	A0	
3005	F5F9	D8	MNEMR .BYT \$D8, \$62, \$5A, \$4B, \$26, \$62, \$94
3005	F5FA	62	
3005	F5FB	5A	
3005	F5FC	4B	
3005	F5FD	26	
3005	F5FE	62	
3005	F5FF	94	
3006	F600	8B	.BYT \$8B
3007	F601	54	.BYT \$54, \$44, \$CB, \$54, \$68, \$44, \$EB, \$94
3007	F602	44	
3007	F603	CB	
3007	F604	54	
3007	F605	68	
3007	F606	44	
3007	F607	EB	
3007	F608	94	
3008	F609	00	.BYT 0, \$B4, B, \$B4, \$74, \$B4, \$2B, \$6E
3008	F60A	B4	
3008	F60B	0B	
3008	F60C	84	
3008	F60D	74	
3008	F60E	B4	
3008	F60F	2B	
3008	F610	6E	
3009	F611	74	.BYT \$74, \$F4, \$CC, \$4A, \$72, \$F2, \$A4, \$8A
3009	F612	F4	
3009	F613	CC	
3009	F614	4A	
3009	F615	72	
3009	F616	F2	
3009	F617	A4	
3009	F618	8A	
3010	F619	00	.BYT 0, \$AA, \$A2, \$A2, \$74, \$74, \$74, \$72
3010	F61A	AA	
3010	F61B	A2	
3010	F61C	A2	
3010	F61D	74	
3010	F61E	74	
3010	F61F	74	

LINE #	LOC	CODE	LINE
3010	F620	72	
3011	F621	44	
3011	F622	68	.BYT \$44,\$68,\$B2,\$32,\$B2,0,\$22,0
3011	F623	B2	
3011	F624	32	
3011	F625	B2	
3011	F626	00	
3011	F627	22	
3011	F628	00	
3012	F629	1A	.BYT \$1A,\$1A,\$26,\$26,\$72,\$72,\$88,\$C8
3012	F62A	1A	
3012	F62B	26	
3012	F62C	26	
3012	F62D	72	
3012	F62E	72	
3012	F62F	88	
3012	F630	C8	
3013	F631	C4	.BYT \$C4,\$CA,\$26,\$4B,\$44,\$44,\$A2,\$C8
3013	F632	CA	
3013	F633	26	
3013	F634	4B	
3013	F635	44	
3013	F636	44	
3013	F637	A2	
3013	F638	C8	

LINE #	LOC	CODE	LINE
3015	F639		;*****
3016	F639		;*** AIM TEXT EDITOR ***
3017	F639		;*** 05/01/78 ***
3018	F639		;*****
3020	F639		; R=READ FROM ANY INPUT DEVICE
3021	F639		; I=INSERT A LINE FROM INPUT DEV
3022	F639		; K=DELETE A LINE
3023	F639		; U=GO UP ONE LINE
3024	F639		; D=GO DOWN ONE LINE
3025	F639		; L=LIST LINES TO OUTPUT DEV
3026	F639		; T=GO TO TOP OF TEXT
3027	F639		; B=GO TO BOTTOM OF TEXT
3028	F639		; F=FIND STRING
3029	F639		; C=CHANGE STRING TO NEW STRING
3030	F639		; Q=QUIT EDITOR
3031	F639		; (SPACE)=DISPLAY CURRENT LINE
3033	F639		;***** E COMMAND-EDITOR ENTRY (FROM MONITOR) *****
3034	F639	20 13 EA	EDIT JSR CRLOW
3035	F63C	A0 6C	LDY #EMSG1-M1
3036	F63E	20 AF E7	JSR KEP ;START UP MSG
3037	F641	20 13 EA	JSR CRLOW
3038	F644	20 A3 E7	EDIO JSR FROM
3039	F647	B0 FB	BCS EDIO
3040	F649	AD 1E A4	LDA CKSUM ;IS CLR IF ADDR WAS INPUTTED
3041	F64C	F0 03	BEQ #+5
3042	F64E	20 DB E2	JSR WRITAZ ;OUTPUT DEFAULT ADDR (0200)
3043	F651	A2 01	LDX #1
3044	F653	BD 1C A4	EDI1 LDA ADDR,X
3045	F656	95 E3	STA TEXT,X
3046	F658	95 E1	STA BOTLN,X
3047	F65A	9D 1A A4	STA S1,X ;FOR MEMORY TEST
3048	F65D	CA	DEX
3049	F65E	10 F3	BPL EDI1
3050	F660	20 3B E8	JSR BLANK2
3051	F663	20 A7 E7	EDI2 JSR TO ;END
3052	F666	B0 FB	BCS EDI2
3053	F668	20 BC FB	JSR TOPNO ;TRANSF TEXT TO ADDR FOR RAM CHECK
3054	F66B	AD 1E A4	LDA CKSUM ;IS CLR IF ADDR WAS INPUTTED
3055	F66E	F0 10	BEQ EDI4 ;BRNCH IF NOT DEFAULT VALUE
3056	F670	20 34 F9	JSR SAVNOW
3057	F673	20 B6 F6	EDI3 JSR EDI ;CARRY IS SET IF NO RAM THERE
3058	F676	90 FB	BCC EDI3
3059	F678	A9 00	LDA #0 ;SET UPPER LIMIT TO BEGINNING...
3060	F67A	BD 1C A4	STA ADDR ;OF PAGE
3061	F67D	20 DB E2	JSR WRITAZ ;OUTPUT DEFAULT VALUE ,UPPER LIMIT
3062	F680	AD 1C A4	EDI4 LDA ADDR
3063	F683	85 E5	STA END
3064	F685	AD 1D A4	LDA ADDR+1
3065	F688	85 E6	STA END+1
3066	F68A	20 34 F9	JSR SAVNDW
3067	F68D		;NOW SEE IF MEMORY IS THERE

LINE #	LOC	CODE	LINE
3068	F68D	20 B6 F6	EDI5 JSR EDI
3069	F690	90 FB	BCC EDI5
3070	F692	A5 E6	LDA ENDR+1 ;CMP WITH END
3071	F694	CD 1D A4	CMP ADDR+1
3072	F697	F0 11	BEQ EDI7
3073	F699	80 13	BCS EDI8
3074	F69B	20 BC FB	EDI6 JSR TOPND ;RESTORE NOWLN
3075	F69E	A9 00	LDA #0
3076	F6A0	91 DF	STA (NOWLN)Y ;END OF TEXT MARKER
3077	F6A2	20 13 EA	JSR CRLOW
3078	F6A5	A9 52	LDA #'R ;FORCE READ COMMAND
3079	F6A7	4C 8D FA	JMP ENTRY
3080	F6AA	A5 E5	EDI7 LDA END ;IF ZERO MEM IS OK
3081	F6AC	F0 ED	BEQ EDI6
3082	F6AE	A9 00	EDI8 LDA #0
3083	F6B0	8D 1C A4	STA ADDR
3084	F6B3	4C 33 EB	JMP MEMERR ;NO MEMORY FOR THOSE LIMITS
3086	F6B6	A0 00	EDI LDY #0 ;CHK IF MEMORY WRITES
3087	F6B8	20 B7 FE	JSR PATCH6 ;GET BYTE ADDR BY ADDR,ADDR+1
3088	F6BB	4B	PHA ;SAVE IT
3089	F6BC	A9 AA	LDA #'AA ;SET THIS PATTERN
3090	F6BE	20 78 EB	JSR SADDR ;CHK IT
3091	F6C1	D0 09	BNE EDI2B
3092	F6C3	68	PLA
3093	F6C4	20 78 EB	JSR SADDR ;RESTORE CHR
3094	F6C7	EE 1D A4	INC ADDR+1 ;NEXT PAG
3095	F6CA	18	CLC ;IT WROTE
3096	F6CB	60	RTS
3097	F6CC	38	EDI2B SEC ;DIDNT WRITE
3098	F6CD	68	PLA
3099	F6CE	60	RTS
3101	F6CF		***** T COMMAND-REENTRY EDITOR *****
3102	F6CF		;RE-ENTRY POINT, TEXT ALREADY THERE
3103	F6CF	20 24 EA	REENTR JSR CRCK ;(CR) IF PRI ON
3104	F6D2	20 BC FB	TF JSR TOPND ;GO TO TOP
3105	F6D5	4C B9 F7	JMP IN03A ;DISPLAY LINE
3107	F6D8		***** U COMMAND-UP LINE *****
3108	F6D8		;GO UP ONE LINE BUT...
3109	F6D8		;DOWN IN ADDRESSING MEMORY
3110	F6D8	20 DB F8	DNNO JSR ATTOP ;THIS RTN DOESNT PRINT
3111	F6DB	90 06	BCC DOW1 ;NOT TOP
3112	F6DD	20 27 F7	JSR PLNE ;ARE AT TOP
3113	F6E0	4C 78 FA	JMP ERRO
3114	F6E3	A0 00	DOW1 LDY #0
3115	F6E5	20 1D F9	JSR SUB ;DECREMENT NOWLN PAST (CR)
3116	F6E8	20 1D F9	JSR SUB
3117	F6EB	20 DB F8	JSR ATTOP
3118	F6EE	80 30	BCS UP4
3119	F6F0	B1 DF	LDA (NOWLN)Y
3120	F6F2	C9 0D	CMP #CR
3121	F6F4	D0 F2	BKE DOW2
3122	F6F6	4C 28 F9	JMP AD1

LINE #	LOC	CODE	LINE
3124	F6F9		***** D COMMAND-DOWN LINE *****
3125	F6F9		;GO DOWN ONE LINE BUT...
3126	F6F9		;UP IN ADDRESSING MEMORY
3127	F6F9	20 09 F7	UP JSR UPND
3128	F6FC	20 27 F7	JSR PLNE ;DISPLAY LINE & CHCK BOTTOM
3129	F6FF	20 E9 FB	JSR ATBOT
3130	F702	90 1C	BCC UP4
3131	F704	AQ 72	LDY #EMSG2-M1 ;PRINT 'END'
3132	F706	4C AF E7	JMP KEP
3133	F709	A0 00	UPND LDY #0
3134	F70B	20 E9 FB	JSR ATBOT
3135	F70E	90 03	BCC UP1
3136	F710	4C 5C FA	JMP ENDERR
3137	F713	B1 DF	UP1 LDA (NOWLN)Y
3138	F715	F0 09	BEQ UP4
3139	F717	C8	INY
3140	F718	C9 0D	CMP #CR
3141	F71A	D0 F7	BNE UP1
3142	F71C	9B	TYA
3143	F71D	20 2A F9	JSR ADDA ;ADD LENGTH TO CURRENT LINE
3144	F720	60	UP4 RTS
3146	F721		***** B COMMAND-GO TO BOTTOM *****
3147	F721	20 C5 F8	BT JSR SETBOT
3148	F724		;START U-COMMAND HERE
3149	F724	20 DB F6	DOWN JSR DNNO ;U COMMAND
3151	F727		***** (SPACE) COMMAND-DISPLAY CURRENT LINE *****
3152	F727	A0 00	PLNE LDY #0 ;PRINT CURRENT LINE
3153	F729	B1 DF	F02 LDA (NOWLN)Y
3154	F72B	F0 0E	BEQ F01 ;FAST END ?
3155	F72D	C9 0D	CMP #CR ;DONE?
3156	F72F	F0 0A	BEQ F01
3157	F731	20 BC E9	JSR OUTALL ;PUT IT SOMEWHERE
3158	F734	99 3B A4	STA DIRUFF,Y
3159	F737	C8	INY
3160	F738	4C 29 F7	JMP F02
3161	F73B	84 EA	P01 STY LENGTH
3162	F73D	84 E9	STY OLDLEN
3163	F73F	AC 13 A4	P03 LDY OUTFLG ;ONE MORE (CR) FOR TAPE
3164	F742	C0 0D	CPY #CR
3165	F744	F0 03	BEQ F00
3166	F746	4C F0 E9	JMP CRLF ;TO OUTPUT DEV
3167	F749	4C 24 EA	P00 JMP CRCK ;(CR), & DONT CLR DISPL
3169	F74C		***** K COMMAND-KILL LINE *****
3170	F74C		;DELETE CURRENT LINE
3171	F74C	20 B6 FB	DLNE JSR KIFLG ;CLR K OR I COMM FLG
3172	F74F	EA	NOP
3173	F750	EA	NOP
3174	F751	EA	NOP
3175	F752	20 27 F7	JSR PLNE
3176	F755	20 E9 FB	JSR ATBOT
3177	F758	80 CD	BCS PLNE ;AT END OF TEXT
3178	F75A	A0 00	LDY #0

LINE #	LOC	CODE	LINE
3179	F75C	B4 EA	STY LENGTH
3180	F75E	20 3F F9	JSR REPLAC ; KILL LINE
3181	F761	4C 27 F7	JMP PLNE
!			
3183	F764		;***** I COMMAND-INSERT LINE *****
3184	F764	20 6D F7	IN JSR INL
3185	F767	20 F9 F6	JSR UP ; DISPLAY NEXT LINE DOWN
3186	F76A	4C 78 FA	JMP ERRO ; IF AT BOTTOM PRINT "END"
3187	F76D	20 B6 F8	INL JSR KIFLG ; CLR K OR I COMM FLG
3188	F770	A0 00	LDY #0 ; GET LINE INTO DIRUFF
3189	F772	B4 E9	STY OLDLEN
3190	F774	20 BD E7	JSR FROMPT
3191	F777	20 44 EB	JSR CLR
3192	F77A	20 93 E9	IN02 JSR INALL
3193	F77D	20 FB FE	JSR PATC12 ; CLR, SO WE CAN OUTPUT TO PRI
3194	F780	C9 7F	CMF #*7F ; RUB
3195	F782	4C 2A FF	JMP PATC17 ; NO ZEROS IN CASE OF PAPER TAPE
3196	F785	C9 0A	IN02A CMF #*LF
3197	F787	F0 F1	BEQ IN02
3198	F789	C9 0D	CMF #*CR
3199	F78B	F0 1B	BEQ IN03
3200	F78D	C0 3C	CPY #60 ; DO NOT INCR Y IF 60
3201	F78F	B0 08	BCC IN03B
3202	F791	99 38 A4	STA DIRUFF,Y
3203	F794	C8	INY
3204	F795	C0 3C	CPY #60
3205	F797	D0 E1	BNE IN02 ; CONTIN, DISP WONT ALLOW > 60 CH
3206	F799	A0 3C	IN03B LDY #60 ; SET Y TO MAX OF 60
3207	F79B	A9 01	LDA #*01
3208	F79D	0D 11 A4	ORA PRIFLG ; DO NOT OUTPUT TO PRI ANY MORE
3209	F7A0	8D 11 A4	STA PRIFLG ; OTHERWISE CLOBBERS THE BUFFER
3210	F7A3	8C 15 A4	STY CURP02
3211	F7A6	D0 D2	BNE IN02 ; GO BACK
3212	F7A8	84 EA	IN03 STY LENGTH
3213	F7AA	C0 00	CPY #0 ; FIRST CHAR?
3214	F7AC	D0 17	BNE IN05
3215	F7AE	AD 19 A4	LDA COUNT ; K OR I COMM FLG ?
3216	F7B1	D0 12	BNE IN05 ; BRANCH IF C COMMAND
3217	F7B3	20 24 EA	JSR CRCK ; (CR) IF PRI PNTR DIFF FROM 0
3218	F7B6	20 03 FF	JSR PATC13 ; TURN ON TAPES & SET DEFAULT DEV
3219	F7B9	20 27 F7	IN03A JSR PLNE ; DISPLAY NEXT LINE DOWN
3220	F7BC	20 09 F7	JSR UPNO ; PRINT "END" IF BOTTOM
3221	F7B8	20 D8 F6	JSR DNNO
3222	F7C2	4C 78 FA	JMP ERRO
3223	F7C5	20 3F F9	IN05 JSR REPLAC ; INSERT THE LINE
3224	F7C8	4C 24 EA	JMP CRCK ; (CR) IF PRI PTR NOT 0
3226	F7CB		;***** R COMMAND-READ LINE *****
3227	F7CB		; READ TEXT FROM ANY INPUT DEVICE UNTIL
3228	F7CB		; TWO CONSECUTIVE (CR) ARE ENCOUNTER.
3229	F7CB	20 48 EB	INPU JSR WHEREI
3230	F7CE	AC 12 A4	LDY INFLG ; IF TAPE DO NOT ERRASE BUFFER
3231	F7D1	C0 54	CPY #*T
3232	F7D3	F0 03	BEQ INPU1
3233	F7D5	20 13 EA	JSR CRLOW

LINE #	LOC	CODE	LINE
3234	F7D8	20 6D F7	INPU1 JSR INL
3235	F7DB	20 09 F7	JSR UPNO ; NEXT LINE
3236	F7DE	4C DB F7	JMP INPU1
3238	F7E1		;***** L COMMAND-LIST LINES *****
3239	F7E1		; PRINT FROM HERE N LINES TO ACTIVE OUTPUT DEV
3240	F7E1	20 37 E8	LST JSR PSL1 ; PRINT "/"
3241	F7E4	20 85 E7	JSR GCNT ; GET LINES COUNT
3242	F7E7	20 13 EA	JSR CRLOW
3243	F7EA	20 71 E8	JSR WHEREO ; WHERE TO
3244	F7ED	4C F8 F7	JMP LST02 ; ONE MORE LINE
3245	F7F0	20 07 E9	LST01 JSR RCHEK
3246	F7F3	20 90 E7	JSR DONE
3247	F7F6	F0 0B	BEQ LST3
3248	F7F8	20 27 F7	LST02 JSR PLNE
3249	F7FB	20 09 F7	JSR UPNO ; NEXT LINE
3250	F7FE	20 E9 F8	JSR ATBOT
3251	F801	90 ED	BCC LST01 ; NO
3252	F803	20 3F F7	LST3 JSR P03 ; ONE MORE CRLF FOR TAPE
3253	F806	20 0D FF	JSR PATC14 ; CLOSE TAPE IF NEEDED
3254	F809	4C 5C FA	JMP ENDERR
3256	F80C		;***** F COMMAND-FIND STRING *****
3257	F80C		; FIND STRING AND PRINT LINE TO TERMINAL
3258	F80C	20 1E F8	FCHAR JSR FCH
3259	F80F	AD 15 A4	FCHA1 LDA CURP02 ; SAVE BUFFER PNTR
3260	F812	4B	PHA
3261	F813	20 44 EB	JSR CLR ; CLEAR DISP PNTR
3262	F816	20 27 F7	JSR PLNE
3263	F819	68	PLA
3264	F81A	8D 15 A4	STA CURP02
3265	F81D	60	RTS
3266	F81E		; FIND A CHARACTER STRING
3267	F81E	A0 00	FCH LDY #0
3268	F820	20 BD E7	JSR PROMPT
3269	F823	20 5F E9	FC1 JSR RDRUB ; GET THE CHARACTER
3270	F826	C9 0D	CMF #*D ; REUSE OLD ARGUMENT??
3271	F828	D0 0A	BNE FC3
3272	F82A	C0 00	CPY #0 ; FIRST CHAR?
3273	F82C	D0 06	BNE FC3
3274	F82E	20 09 F7	FC2 JSR UPNO ; NEXT LINE DOWN
3275	F831	4C 49 F8	JMP FC5
3276	F834	C9 0B	FC3 CMF #*CR ; DONE
3277	F836	F0 0B	BEQ FC4
3278	F838	99 EB 00	STA STRING,Y
3279	F83B	C8	INY
3280	F83C	D0 14	CPY #20 ; MAX LENGTH
3281	F83E	D0 E3	BNE FC1
3282	F840	4C 72 FA	JMP ERROR
3283	F843	20 24 EA	FC4 JSR CRCK ; CLEAR DISPLAY
3284	F846	8C 29 A4	STY STIY+2 ; COUNT OF CHARACTERS
3285	F849	A0 00	FC5 LDY #0
3286	F84B	8C 15 A4	STY CURP02 ; START AT BEGINNING OF LINENTR IS
3287	F84E	AC 15 A4	FC6 LDY CURP02 ; CLOBBER
3288	F851	A2 00	LDX #0

LINE #	LOC	CODE	LINE	LINE
3289	F853	B1 DF	FC7	LDA (NOWLN)Y ;GET THE CHARACTER
3290	F855	D0 03		BNE FC8 ;NOT AT END
3291	F857	4C 5C FA		JMP ENDERR
3292	F85A	C9 0D	FC8	CMP #CR ;END OF LINE
3293	F85C	F0 10		BEQ FC2
3294	F85E	D5 E8		CMP STRING.X
3295	F860	F0 06		BEQ FC9
3296	F862	EE 15 A4		INC CURPO2
3297	F865	4C 4E F8		JMP FC6
3298	F868	C8	FC9	INY
3299	F869	E8		INX
3300	F86A	EC 29 A4		CPX STIY+2 ;DONE?
3301	F86D	D0 E4		BNE FC7
3302	F86F	60		RTS
3304	F870			***** Q COMMAND-EXIT EDITOR *****
3305	F870			; EXIT THE TEXT EDITOR NEATLY
3306	F870	20 13 EA		STOP JSR CROW
3307	F873	4C A1 E1		JMP COMIN
3309	F876			***** C COMMAND-CHANGE STRING *****
3310	F876			;CHANGE STRING TO ANOTHER STRING IN A LINE
3311	F876	20 B2 F8	CHNG	JSR CFLG ;SET C COMMAND FLG
3312	F879	20 0C F8		JSR FCHAR ;FIND CORRECT LINE
3313	F87C	20 3C E9	CHN1	JSR READ ;IS (CR) IF OK
3314	F87F	C9 0D		CMP #CR
3315	F881	F0 09		BEQ CHN2
3316	F883	20 2E F8		JSR FC2 ;TRY NEXT ONE
3317	F886	20 0F F8		JSR FCHA1 ; SHOW LINE
3318	F889	4C 7C F8		JMP CHN1
3319	F88C	AD 29 A4	CHN2	LDA STIY+2 ;GET CHAR COUNT
3320	F88F	85 E9		STA OLDLEN ;GET READY FOR REPLAC
3321	F891	AD 15 A4		LDX CURPO2 ;PNTR TO BEGINNING OF STRING
3322	F894	48		PHA ;SAVE IT
3323	F895	20 2A F9		JSR ADDA ;ADD TO NOWLN (LINE PNTR)
3324	F898	20 44 EB		JSR CLR ;CLEAR DISP
3325	F89B	A0 05		LDY #M3-M1 ;PRINT 'D'
3326	F89D	20 AF E7		JSR KEP
3327	F8A0	A0 00		LDY #0
3328	F8A2	20 7A F7		JSR INO2 ;GET NEW STRING & REPLAC
3329	F8A5	68		FLA
3330	F8A6	AA		TAX
3331	F8A7	F0 06		BEQ CHN4
3332	F8A9	20 1D F9	CHN3	JSR SUB ;RESTORE NOWLN WHERE IT WAS
3333	F8AC	CA		DEX
3334	F8AD	D0 FA		BNE CHN3
3335	F8AF	4C 27 F7	CHN4	JMP PLNE ;DISPLAY THE CHANGED LINE
3337	F8B2			;THE FOLLOWING ARE SUBROUTINES USED BY COMMANDS
3338	F8B2	A9 01	CFLG	LDA #1 ;SET FLG FOR C COMMAND
3339	F8B4	D0 02		BNE KI2
3340	F8B6	A9 00	KIFLG	LDA #0 ;CLR K OR I COMMAND FLG

LINE #	LOC	CODE	LINE	LINE
3341	F8B8	8D 19 A4	KI2	STA COUNT
3342	F8BB	60		RTS
3344	F8BC	A5 E3	TOPND	LDA TEXT ;SET CURRENT LINE TO TOP
3345	F8BE	A6 E4		LDX TEXT+1
3346	F8C0	85 DF	TP01	STA NOWLN
3347	F8C2	86 E0		STX NOWLN+1
3348	F8C4	60		RTS
3350	F8C5	A5 E1	SETBOT	LDA BOTLN ;SET CURRENT LINE TO BOTTOM
3351	F8C7	A6 E2		LDX BOTLN+1
3352	F8C9	85 E7		STA SAVE
3353	F8CB	86 E8		STX SAVE+1
3354	F8CD	4C C0 F8		JMP TP01
3356	F8D0	AD 1C A4	RESNOW	LDA ADDR ;RESTORE CURRENT LINE ADDRESS
3357	F8D3	85 DF		STA NOWLN
3358	F8D5	AD 1D A4		LDA ADDR+1
3359	F8D8	85 E0		STA NOWLN+1
3360	F8DA	60		RTS
3362	F8DB			; SEE IF CURRENT LINE AT TOP (C SET IF SO)
3363	F8DB	A5 DF	ATTOP	LDA NOWLN
3364	F8DD	C5 E3		CMP TEXT
3365	F8DF	D0 16		BNE AT01
3366	F8E1	A5 E0		LDA NOWLN+1
3367	F8E3	C5 E4		CMP TEXT+1
3368	F8E5	D0 10		BNE AT01
3369	F8E7	38		SEC
3370	F8E8	60		RTS
3372	F8E9			; SEE IF CURRENT LINE AT BOTTOM (C SET IF SO)
3373	F8E9	A5 DF	ATBOT	LDA NOWLN
3374	F8EB	A6 E0		LDX NOWLN+1
3375	F8ED	C5 E1		CMP BOTLN
3376	F8EF	D0 06		BNE AT01
3377	F8F1	E4 E2		CPX BOTLN+1
3378	F8F3	D0 02		BNE AT01
3379	F8F5	38	AT02	SEC
3380	F8F6	60		RTS
3381	F8F7	18	AT01	CLC
3382	F8F8	60		RTS
3384	F8F9			;SEE IF WE RAN PAST END OF BUFFER LIMIT
3385	F8F9	A5 E1	ATEND	LDA BOTLN
3386	F8FB	A6 E2		LDX BOTLN+1
3387	F8FD	E4 E6		CPX END+1 ;HIGH BYTE > OR = ?
3388	F8FF	90 F6		BCC AT01
3389	F901	D0 F2		BNE AT02
3390	F903	C5 E5		CMP END ;LOW BYTE > OR = ?
3391	F905	90 F0		BCC AT01
3392	F907	80 EC		BCS AT02
3394	F909			; SAVE CURRENT LINE (NOWLN) IN S1
3395	F909	A5 DF	NOWS1	LDA NOWLN

```

LINE # LOC      CODE      LINE
3396 F90B A6 E0          LDX NOWLN+1
3397 F90D 4C 16 F9          JMP ADDS1A

3399 F910          ; MOVE ADDR INTO S1
3400 F910 AD 1C A4  ADDR$1 LDA ADDR
3401 F913 AE 1D A4  LDX ADDR+1
3402 F916 8D 1A A4  ADDS1A STA S1
3403 F919 8E 1B A4  STX S1+1
3404 F91C 60          RTS

3406 F91D          ; SUBTRACT ONE FROM CURRENT LINE (NOWLN)
3407 F91D C6 DF  SUB  DEC NOWLN
3408 F91F A5 DF  LDA NOWLN
3409 F921 C9 FF  CMP #$FF
3410 F923 D0 02  BNE SUB1
3411 F925 C6 E0  DEC NOWLN+1
3412 F927 60  SUB1  RTS

3414 F928          ; ADD ACC TO CURRENT LINE (NOWLN)
3415 F928 A9 01  AD1  LDA #1
3416 F92A 18  ADDA  CLC
3417 F92B 65 DF  ADC NOWLN
3418 F92D 85 DF  STA NOWLN
3419 F92F 90 02  RCC ADDA1
3420 F931 E6 E0  INC NOWLN+1
3421 F933 60  ADDA1  RTS

3423 F934 A5 DF  SAVNOW LDA NOWLN          ; SAVE CURRENT LINE INTO ADDR
3424 F936 8D 1C A4  STA ADDR
3425 F939 A5 E0  LDA NOWLN+1
3426 F93B 8D 1D A4  STA ADDR+1
3427 F93E 60  REF2  RTS
    
```

```

LINE # LOC      CODE      LINE
3429 F93F          ; MOVE CURRENT TEXT AROUND TO HAVE
3430 F93F          ; SPACE TO PUT IN THE NEW BUFFER
3431 F93F A4 EA  REPLAC LDY LENGTH
3432 F941 C4 E9  CPY OLDLEN          ; COMPARE OLD AND NEW LENGTHS
3433 F943 D0 1A  BNE R2W          ; BRANCH IF DIFF
3434 F945 F0 07  BEQ R87          ; LENGTHS ARE EQUAL. JUST REPLACE
3435 F947 A9 0D  R8  LDA #CR
3436 F949 91 DF  STA (NOWLN)Y
3437 F94B 20 4A FA  JSR GOGO

3439 F94E          ; LENGTH = OLDLEN
3440 F94E 88  R87  DEY
3441 F94F C0 FF  CPY #$FF
3442 F951 F0 EB  BEQ REP2
3443 F953 B9 38 A4  R88  LDA DIRUFF,Y
3444 F956 91 DF  STA (NOWLN)Y
3445 F958 20 4A FA  JSR GOGO
3446 F95B 88  DEY
3447 F95C 10 F5  BPL R88
3448 F95E 60  RTS
3449 F95F B0 6E  R2W  RCS R100          ; LENGTH > OLDLEN

3451 F961          ; LENGTH < OLDLEN
3452 F961 20 34 F9  JSR SAVNOW          ; PUT NOWLN INTO ADDR
3453 F964 20 10 F9  JSR ADDR$1          ; PUT IT IN S1 ALSO
3454 F967 A5 E9  LDA OLDLEN
3455 F969 38  SEC
3456 F96A E5 EA  SBC LENGTH          ; GET DIFFERENCE IN LENGTHS
3457 F96C A4 EA  LDY LENGTH
3458 F96E D0 07  BNE RQP
3459 F970 AE 19 A4  LDX COUNT          ; C-COMM ?
3460 F973 D0 02  BNE RQP          ; YES, JUMP
3461 F975 69 00  ADC #0          ; INCLUDE (CR)
3462 F977 48  RQP  PHA
3463 F97B 18  CLC
3464 F979 6D 1A A4  ADC S1
3465 F97C 8D 1A A4  STA S1
3466 F97F 90 03  BCC R6
3467 F981 EE 1B A4  INC S1+1
3468 F984 A9 1A  R6  LDA #(S1
3469 F986 20 58 EB  JSR LDAY
3470 F989 91 DF  STA (NOWLN)Y          ; ... AND MOVE IT UP (DOWN IN ADDR)
3471 F98B 20 4A FA  JSR GOGO
3472 F98E AA  TAX
3473 F98F AD 1A A4  LDA S1
3474 F992 C5 E1  CMP BOTLN          ; DONE ??
3475 F994 D0 07  BNE R5
3476 F996 AD 1B A4  LDA S1+1
3477 F999 C5 E2  CMP BOTLN+1
3478 F99B F0 0E  BEQ R7
3479 F99D 20 2B F9  R5  JSR AD1
3480 F9A0 EE 1A A4  INC S1
3481 F9A3 D0 03  BNE R55
3482 F9A5 EE 1B A4  INC S1+1
3483 F9A8 4C B4 F9  R55  JMP R6
    
```



LINE #	LOC	CODE	LINE	LINE
3484	F9AE	20 D0 F8	R7	JSR RESNOW ;RESTORE NOWLN
3485	F9AE	68		PLA ;RESTORE DIFFERENCE
3486	F9AF	8D 2A A4		STA CFIY ;SAVE IT
3487	F9B2	A5 E1		LDA BOTLN
3488	F9B4	38		SEC
3489	F9B5	E1 2A A4		SBC CFIY ;AND SUBTRACT IT FROM BOTTOM
3490	F9B8	85 E1		STA BOTLN
3491	F9BA	B0 02		BCS R9
3492	F9BC	C6 E2		DEC BOTLN+1
3493	F9BE	AD 19 A4	R9	LDA COUNT ;C COMM OR K , I COMM ?
3494	F9C1	D0 04		BNE R10
3495	F9C3	A4 EA		LDY LENGTH
3496	F9C5	D0 05		BNE R11
3497	F9C7	A4 EA	R10	LDY LENGTH
3498	F9C9	D0 83		BNE R87
3499	F9CB	60		RTS
3500	F9CC	4C 47 F9	R11	JMP R8
3502	F9CF			;LENGTH > OLDLEN
3503	F9CF	A5 EA	R100	LDA LENGTH ;NEW LINE IS LONGER
3504	F9D1	38		SEC
3505	F9D2	E5 E9		SBC OLDLEN
3506	F9D4	A4 E9		LDY OLDLEN
3507	F9D6	D0 02		BNE R101 ;ALREADY HAVE ROOM FOR CR
3508	F9D8	69 00		ADC #0 ;ADD ONE TO DIFFERENCE
3509	F9DA	48	R101	PHA
3510	F9DB	20 34 F9		JSR SAVNOW ;NOWLN INTO S1
3511	F9DE	20 C5 F8		JSR SETBOT
3512	F9E1	A0 00		LDY #0
3513	F9E3	B1 DF	R102	LDA (NOWLN)Y
3514	F9E5	C9 00		CMP #0
3515	F9E7	F0 06		BEQ R108
3516	F9E9	20 28 F9		JSR AD1
3517	F9EC	4C E3 F9		JMP R102
3518	F9EF	68	R108	PLA
3519	F9F0	48		PHA
3520	F9F1	18		CLC
3521	F9F2	65 E1		ADC BOTLN ;ADD DIFFERENCE TO END
3522	F9F4	85 E1		STA BOTLN ;STORE NEW END
3523	F9F6	90 02		BCC R103
3524	F9F8	E6 E7		INC BOTLN+1
3525	F9FA	20 F9 FB	R103	JSR ATEND
3526	F9FD	90 0B		BCC R107
3527	F9FF	A5 E7		LDA SAVE ;RESTORE OLD BOTTOM
3528	FA01	85 E1		STA BOTLN
3529	FA03	A5 EB		LDA SAVE+1
3530	FA05	85 E2		STA BOTLN+1
3531	FA07	4C 5C FA		JMP ENDERR ;RAN PAST BUFFER END
3532	FA0A	20 09 F9	R107	JSR NOWS1 ;SAVE CURRENT END
3533	FA0D	68		PLA
3534	FA0E	18		CLC
3535	FA0F	65 DF		ADC NOWLN
3536	FA11	85 DF		STA NOWLN
3537	FA13	90 02		BCC R104
3538	FA15	E6 E0		INC NOWLN+1

LINE #	LOC	CODE	LINE	LINE
339	FA17	A9 1A	R104	LDA #CS1
340	FA19	20 58 EB		JSR LIAY
341	FA1C	91 DF		STA (NOWLN)Y
342	FA1E	20 4A FA		JSR GOGO
343	FA21	AD 1A A4		LDA S1
344	FA24	CD 1C A4		CMP ADDR
345	FA27	D0 08		BNE R105
346	FA29	AD 1B A4		LDA S1+1
347	FA2C	CD 1D A4		CMP ADDR+1
348	FA2F	F0 13		BEQ R106 ;BACK WHERE WE STARTED ??
349	FA31	20 1D F9	R105	JSR SUB ;BRANCH IF DONE
350	FA34	CE 1A A4		DEC S1
351	FA37	AD 1A A4		LDA S1
352	FA3A	C9 FF		CMP #FF
353	FA3C	D0 03		BNE R1051
354	FA3E	CE 1B A4		DEC S1+1
355	FA41	4C 17 FA	R1051	JMP R104
356	FA44	20 D0 F8	R106	JSR RESNOW
357	FA47	4C BE F9		JMP R9
359	FA4A			;SEE IF IT WROTE INTO MEMORY
360	FA4A	D1 DF	GOGO	CMP (NOWLN)Y
361	FA4C	F0 0D		BEQ GOGO1
362	FA4E			;MOVE ADDRESS
363	FA4E	A5 DF		LDA NOWLN
364	FA50	8D 1C A4		STA ADDR
365	FA53	A5 E0		LDA NOWLN+1
366	FA55	8D 1D A4		STA ADDR+1
367	FA58	4C 33 EB		JMP MEMERR
368	FA5B	60	GOGO1	RTS ;OK

## ERROR HANDLERS

PA00-J001A.....PAGE 0082

LINE #	LOC	CODE	LINE
3570	FA5C	20 44 E8	ENDERR JSR CLR ;CLEAR PNTR
3571	FA5F	A0 72	LDY #MSG2-M1 ;PRINT "END"
3572	FA61	20 AF E7	JSR KEF
3573	FA64	20 B8 F6	JSR DNNO ;BACK UP TO LAST LINE
3574	FA67	20 42 E8	JSR TTYTST ;IF TTY (CR)
3575	FA6A	D0 03	BNE ENDE2
3576	FA6C	20 13 EA	JSR CRLW
3577	FA6F	4C 78 FA	ENDE2 JMP ERRO
3578	FA72	20 FE E8	ERROR JSR LL
3579	FA75	20 D4 E7	JSR QM
3580	FA78	20 44 E8	ERRO JSR CLR
3581	FA7B	A2 FF	LDX ##FF
3582	FA7D		COM=ERRO
3583	FA7D	9A	TXS
3584	FA7E	20 FE E8	JSR LL ;I/O TO TERMINAL (KB,D/P OR TTY)
3585	FA81	B8	CLD
3586	FA82	20 88 FA	JSR COMM
3587	FA85	4C 78 FA	JMP COM
3589	FA88		;GET EDITOR COMMANDS & DECODE
3590	FA88	A2 00	COMM LDX #0
3591	FA8A	20 BC FE	JSR PATCH8 ;READ A CHAR WITH "=C >"
3592	FA8D	A2 0B	ENTRY LDX #COMCN1
3593	FA8F	D0 AC FA	CD02 CMP COMTRL,X ;COMPARE WITH ALLOWABLE COMMANDS
3594	FA92	F0 0C	BEQ CFND1 ;MATCH ,SO PROCESS COMMAND
3595	FA94	CA	DEX
3596	FA95	10 FB	BPL CD02
3597	FA97	20 D4 E7	JSR QM ;NOT IN LIST ,SO NOT LEGAL COMMAND
3598	FA9A	20 24 EA	JSR CRCK
3599	FA9D	4C 78 FA	JMP ERRO
3600	FAA0	20 17 FF	CFND1 JSR PATC15 ;(CR) & START DECODING COMMAND
3601	FAA3	8D B9 FA	LDA JTBL+1,X
3602	FAA6	8D 1B A4	STA S1+1
3603	FAA9	6C 1A A4	JMP (S1)
3605	FAAC		COMCN1=11
3606	FAAC		;COMMAND TABLE
3607	FAAC	4B 20	COMTRL .BYT 'K RIUULTBFQC'
3608	FA88	4C F7	JTBL .WOR DLNE,PLNE,INPU,IN,DOWN,UP
3608	FA8A	27 F7	
3608	FA8C	CB F7	
3608	FA8E	64 F7	
3608	FAC0	24 F7	
3608	FAC2	F9 F6	
3609	FAC4	E1 F7	.WOR LST,TP,BT,FCHAR,STOP,CHNG
3609	FAC6	D2 F6	
3609	FACB	21 F7	
3609	FACA	0C F8	
3609	FACC	70 F8	
3609	FACE	76 F8	
3611	FAD0		;READ FROM MEMORY FOR ASSEMBLER
3612	FAD0	98	MREAD TYA
3613	FAD1	48	PHA

## ERROR HANDLERS

PA00-J001A.....PAGE 0083

LINE #	LOC	CODE	LINE
3614	FAD2	A0 00	LDY #0
3615	FAD4	B1 DF	LDA (NDWLN)Y
3616	FAD6	8D 2A A4	STA CPIY
3617	FAD9	20 28 F9	JSR AD1
3618	FADC	68	PLA
3619	FADD	AB	TAY
3620	FADE	AD 2A A4	LDA CPIY
3621	FAE1	60	RTS

LINE #	LOC	CODE	LINE
3623	FAE2		; THIS PROGRAM CONVERTS MNEMONIC INSTRUCTIONS INTO MACHIN
3624	FAE2		; CODE AND STORES IT IN THE DESIGNATED MEMORY AREA
3626	FAE2		; FROM TABLE LOCATIONS:
3627	FAE2	00	TYPTR1 .BYT 00,02,00,0B,\$F2,\$FF,\$B0,01
3627	FAE3	02	
3627	FAE4	00	
3627	FAE5	0B	
3627	FAE6	F2	
3627	FAE7	FF	
3627	FAE8	80	
3627	FAE9	01	
3628	FAEA	C0	.BYT \$C0,\$E2,\$C0,\$C0,\$FF,00,00
3628	FAEB	E2	
3628	FAEC	C0	
3628	FAED	C0	
3628	FAEE	FF	
3628	FAEF	00	
3628	FAF0	00	
3629	FAF1	0B	TYPTR2 .BYT 0B,00,\$10,\$B0,\$40,\$C0,00,\$C0
3629	FAF2	00	
3629	FAF3	10	
3629	FAF4	80	
3629	FAF5	40	
3629	FAF6	C0	
3629	FAF7	00	
3629	FAF8	C0	
3630	FAF9	00	.BYT \$00,\$40,00,00,\$E4,\$20,\$B0
3630	FAFA	40	
3630	FAFB	00	
3630	FAFC	00	
3630	FAFD	E4	
3630	FAFE	20	
3630	FAFF	80	
3631	FB00	00	CORR .BYT 00,\$FC,00,0B,0B,\$F8,\$FC,\$F4
3631	FB01	FC	
3631	FB02	00	
3631	FB03	0B	
3631	FB04	0B	
3631	FB05	FB	
3631	FB06	FC	
3631	FB07	F4	
3632	FB08	0C	.BYT \$0C,\$10,04,\$F4,00,\$20,\$10
3632	FB09	10	
3632	FB0A	04	
3632	FB0B	F4	
3632	FB0C	00	
3632	FB0D	20	
3632	FB0E	10	
3633	FB0F	00	! SIZEH .BYT 00,00,\$0F,01,01,01,\$11,\$11
3633	FB10	00	
3633	FB11	0F	
3633	FB12	01	
3633	FB13	01	
3633	FB14	01	

LINE #	LOC	CODE	LINE
433	FB15	11	
433	FB16	11	
434	FB17	02	.BYT 02,02,\$11,\$11,02,\$12,00
434	FB18	02	
434	FB19	11	
434	FB1A	11	
434	FB1B	02	
434	FB1C	12	
434	FB1D	00	
436	FB1E	00	STCODE .BYT \$00,\$0B,\$10,\$1B,\$20,\$2B,\$30,\$3B
436	FB1F	0B	
436	FB20	10	
436	FB21	1B	
436	FB22	20	
436	FB23	2B	
436	FB24	30	
436	FB25	3B	
437	FB26	40	.BYT \$40,\$4B,\$50,\$5B,\$60,\$6B,\$70,\$7B
437	FB27	4B	
437	FB28	50	
437	FB29	5B	
437	FB2A	60	
437	FB2B	6B	
437	FB2C	70	
437	FB2D	7B	
438	FB2E	80	.BYT \$80,\$8B,\$90,\$9B,\$AC,\$AB,\$B0,\$BB
438	FB2F	8B	
438	FB30	90	
438	FB31	9B	
438	FB32	AC	
438	FB33	AB	
438	FB34	BO	
438	FB35	BB	
439	FB36	CC	.BYT \$CC,\$CB,\$D0,\$DB,\$EC,\$EB,\$F0,\$FB
439	FB37	CB	
439	FB38	DO	
439	FB39	DB	
439	FB3A	EC	
439	FB3B	EB	
439	FB3C	FO	
439	FB3D	FB	
440	FB3E	0C	.BYT \$0C,\$2C,\$4C,\$4C,\$8C,\$AC,\$CC,\$EC
440	FB3F	2C	
440	FB40	4C	
440	FB41	4C	
440	FB42	8C	
440	FB43	AC	
440	FB44	CC	
440	FB45	EC	
441	FB46	8A	.BYT \$8A,\$9A,\$AA,\$BA,\$CA,\$DA,\$EA,\$FA
441	FB47	9A	
441	FB48	AA	
441	FB49	BA	
441	FB4A	CA	

LINE #	LOC	CODE	LINE
3641	FB4B	DA	
3641	FB4C	EA	
3641	FB4D	FA	
3642	FB4E	OE	.BYT \$0E,\$2E,\$4E,\$6E,\$8E,\$AE,\$CE,\$EE
3642	FB4F	2E	
3642	FB50	4E	
3642	FB51	6E	
3642	FB52	8E	
3642	FB53	AE	
3642	FB54	CE	
3642	FB55	EE	
3643	FB56	0D	.BYT \$0D,\$2D,\$4D,\$6D,\$8D,\$AD,\$CD,\$ED
3643	FB57	2D	
3643	FB58	4D	
3643	FB59	6D	
3643	FB5A	8D	
3643	FB5B	AD	
3643	FB5C	CD	
3643	FB5D	ED	
3644	FB5E	0D	TYPTB .BYT 13,13,12,13,14,13,12,13
3644	FB5F	0D	
3644	FB60	0C	
3644	FB61	0D	
3644	FB62	0E	
3644	FB63	0D	
3644	FB64	0C	
3644	FB65	0D	
3645	FB66	0D	.BYT 13,13,12,13,13,13,12,13
3645	FB67	0D	
3645	FB68	0C	
3645	FB69	0D	
3645	FB6A	0D	
3645	FB6B	0D	
3645	FB6C	0C	
3645	FB6D	0D	
3646	FB6E	0F	.BYT 15,13,12,13,9,13,12,13
3646	FB6F	0D	
3646	FB70	0C	
3646	FB71	0D	
3646	FB72	09	
3646	FB73	0D	
3646	FB74	0C	
3646	FB75	0D	
3647	FB76	0B	.BYT 8,13,12,13,8,13,12,13
3647	FB77	0D	
3647	FB78	0C	
3647	FB79	0D	
3647	FB7A	0B	
3647	FB7B	0D	
3647	FB7C	0C	
3647	FB7D	0D	
3648	FB7E	0F	.BYT 15,6,11,11,4,10,8,8
3648	FB7F	06	
3648	FB80	0B	
3648	FB81	0B	

LINE #	LOC	CODE	LINE
3648	FB82	04	
3648	FB83	0A	
3648	FB84	0B	
3648	FB85	0B	
3649	FB86	0D	.BYT 13,13,13,13,13,15,13,15
3649	FB87	0D	
3649	FB88	0D	
3649	FB89	0D	
3649	FB8A	0D	
3649	FB8B	0F	
3649	FB8C	0D	
3649	FB8D	0F	
3650	FB8E	07	.BYT 7,7,7,7,5,9,3,3
3650	FB8F	07	
3650	FB90	07	
3650	FB91	07	
3650	FB92	05	
3650	FB93	09	
3650	FB94	03	
3650	FB95	03	
3651	FB96	01	.BYT 1,1,1,1,2,1,1,1
3651	FB97	01	
3651	FB98	01	
3651	FB99	01	
3651	FB9A	02	
3651	FB9B	01	
3651	FB9C	01	
3651	FB9D	01	
3653	FB9E		;PROGRAM STARTS HERE:
3654	FB9E	AD 25 A4	MNEENT LDA SAVFC ;TRANSF PC TO ADDR
3655	FBA1	8D 1C A4	STA ADDR
3656	FBA4	AD 26 A4	LDA SAVFC+1
3657	FBA7	8D 1D A4	STA ADDR+1
3658	FBA8	20 24 EA	STARTM JSR CRCK ;(CR) IF PRI PTR DIFF FROM 0
3659	FBA9	A9 00	LDA #0
3660	FBAF	8D 37 A4	STA CODFLG
3661	FB82	20 3E EB	JSR BLANK
3662	FB85	20 DB E2	JSR WRITAZ ;WRITE ADDRESS
3663	FB88	20 38 EB	JSR BLANK2
3664	FB8B	20 38 EB	JSR BLANK2
3665	FB8E	4C 06 FE	JMP MNEM ;JUMP TO INPUT MNEMONIC OPCODE
3666	FBC1	A9 00	LDA #00 ;SET UP TO FORM MODE MATCH
3667	FBC3	8D 26 01	STA TMASK1
3668	FBC6	8D 27 01	STA TMASK2
3669	FBC9	20 3E EB	JSR BLANK
3670	FBCA	AC 2E 01	LDY TYPE
3671	FBCF	38	SEC
3672	FBD0	6E 26 01	PNTLUP ROR TMASK1 ;SHIFT POINTER TO INSTRUCTION TYPE
3673	FBD3	6E 27 01	ROR TMASK2
3674	FBD6	8B	DEY
3675	FBD7	10 F7	BNE PNTLUP
3677	FBD9		;TEST FOR ONE BYTE INSTRUCTION
3678	FBD9	AC 2E 01	LDY TYPE

LINE #	LOC	CODE	LINE
3679	FBIC	C0 0D	CPY ##0D
3680	FBIE	D0 05	BNE RDADDR
3681	FBE0	A2 00	LIX #00
3682	FBE2	4C CB FC	JMP OPCOMP
3684	FBES		;INPUT ADDRESS FIELD
3685	FBES	A0 06	RDADDR LDY #06 ;CLEAR ADDRESS FIELD (NON HEX)
3686	FBE7	A9 51	LDA #'Q'
3687	FBE9	99 32 01	CLRLUP STA ADFLD-1, Y
3688	FBEC	88	DEY
3689	FBED	D0 FA	BNE CLRLUP ;(LEAVES Y = 0 FOR NEXT PHASE)
3690	FBF7	20 5F E9	JSR RDRUB ;WITH RUBOUT
3691	FBF2	C9 20	CMF ##20 ;IGNORE SPACE CHARACTERS
3692	FBF4	F0 EF	BEQ RDADDR
3693	FBF6	99 33 01	STORCH STA ADFLD, Y ;STORE ADDRESS CHARACTER
3694	FBF9	C8	INY
3695	FBFA	C0 07	CPY #07
3696	FBFC	R0 5C	BCC TRY56
3697	FBFE	20 5F E9	JSR RDRUB ;READ REMAINDER OF ADDRESS CHARS
3698	FC01	C9 20	CMF ##20 ;THRU WHEN (SPACE) OR (CR)
3699	FC03	D0 05	BNE STOR1
3700	FC05	EE 37 A4	INC CDBFLG ;SET CODE FLG
3701	FC08	D0 04	BNE EVAL
3702	FC0A	C9 0D	STOR1 CMP ##0D ;CHECK FOR (CR)
3703	FC0C	D0 EB	BNE STORCH
3705	FC0E		;SEPARATE ADDRESSING MODE FROM ADDRESS FIELD
3706	FC0E	8C 31 A4	EVAL STY TEMPX ;TEMPX NOW HAS NUMBER OF CHAR
3707	FC11	AD 33 01	LDA ADFLD ;CHECK FIRST CHAR FOR # OR (
3708	FC14	C9 23	CMF #'#'
3709	FC16	F0 25	BEQ HATCJ
3710	FC18	C9 28	CMF #'('
3711	FC1A	F0 5A	BEQ PAREN
3712	FC1C	AD 31 A4	LDA TEMPX ;CHECK FOR ACCUMULATOR MODE
3713	FC1F	C9 01	CMF #01
3714	FC21	D0 05	BNE TRYZP
3715	FC23	A2 01	ACCUM LDX #01
3716	FC25	4C CB FC	JMP OPCOMP
3717	FC28	C9 02	TRYZP CMF #02 ;CHECK FOR ZERO PAGE MODE
3718	FC2A	D0 14	BNE TRY34
3719	FC2C	AD 2E 01	LDA TYPE ;CHK FOR BRNCH WITH RELATIVE ADDR
3720	FC2F	C9 0C	CMF ##0C
3721	FC31	D0 05	BNE ZPAGE
3722	FC33	A2 02	LIX #02
3723	FC35	4C CB FC	JMP OPCOMP
3724	FC38	A2 05	ZPAGE LDX #05
3725	FC3A	4C CB FC	JMP OPCOMP
3726	FC3B	4C B6 FC	HATCJ JMP HATCH
3727	FC40	A9 04	TRY34 LDA #04 ;CHECK FOR ABSOLUTE OR ZP, X OR ZP,
3728	FC42	D0 31 A4	CMF TEMPX
3729	FC45	90 15	BCC ABSIND
3730	FC47	A2 02	LIX #02
3731	FC49	20 F1 FD	JSR XORYZ ;CC = X, CS = Y, NE = ABSOLUTE
3732	FC4C	D0 58	BNE ABSOL
3733	FC4E	90 05	BCC ZPX

LINE #	LOC	CODE	LINE
3734	FC50	A2 03	ZPY LIX #03 ;CARRY SET SO ZP, Y MODE
3735	FC52	4C CB FC	JMP OPCOMP
3736	FC55	A2 04	ZPX LIX #04 ;CARRY CLEAR SO ZP, X MODE
3737	FC57	4C CB FC	JMP OPCOMP
3738	FC5A	B0 69	TRY56 BCS ERRORM
3739	FC5C	20 EF FD	ABSIND JSR XORY ;CC=ABS, X CS=ABS, Y NE=ERROR
3740	FC5F	D0 64	BNE ERRORM
3741	FC61	90 0F	BCC ABSX
3742	FC63	A9 09	ABSX LDA #09
3743	FC65	CD 2E 01	CMF TYPE
3744	FC68	D0 04	BNE ABSY1
3745	FC6A	A2 0E	LIX ##0E
3746	FC6C	D0 5D	BNE OPCOMP
3747	FC6E	A2 08	ABSX1 LDX ##08
3748	FC70	D0 59	BNE OPCOMP
3749	FC72	A2 09	ABSX LIX #09 ;CARRY CLEAR SO ABS, X MODE
3750	FC74	D0 55	BNE OPCOMP
3751	FC76	AD 36 01	PAREN LDA ADFLD+3 ;SEE IF (HH, X), (HH)Y OR (HHHH)
3752	FC79	C9 2C	CMF #'/' ;(HHX) (HH), Y ARE OK TOO
3753	FC7B	F0 04	BEQ INDX ;COMMA IN 4TH POSITION = (HH, X)
3754	FC7D	C9 58	CMF #'X'
3755	FC7F	D0 04	BNE TRYINY ;X IN 4TH POSITION = (HHX)
3756	FC81	A2 0B	INDX LIX ##0B
3757	FC83	D0 46	BNE OPCOMP
3758	FC85	C9 29	TRYINY CMF #'/' ;'' IN 4TH POS = (HH)Y OR (HH), Y
3759	FC87	D0 0B	BNE TRYJMP
3760	FC89	20 EF FD	JSR XORY ;CHK TO SEE IF Y INDEX REG DESIRE
3761	FC8C	D0 37	BNE ERRORM
3762	FC8E	90 35	BCC ERRORM
3763	FC90	A2 0A	LIX ##0A
3764	FC92	D0 37	BNE OPCOMP
3765	FC94	AD 38 01	TRYJMP LDA ADFLD+5 ;CHECK FOR FINAL PAREN
3766	FC97	C9 29	CMF #'/'
3767	FC99	D0 2A	BNE ERRORM
3768	FC9B	AD 2E 01	LDA TYPE ;CONFIRM CORRECT ADDRESS TYPE
3769	FC9E	C9 0B	CMF ##0B
3770	FC9A	D0 23	BNE ERRORM
3771	FCA2	A2 0D	LIX ##0D ;OK, FORM IS JMP (HHHH)
3772	FCA4	D0 25	BNE OPCOMP
3773	FCA6	AD 2E 01	ABSOL LDA TYPE ;CHECK FOR BRANCH TO ABSOLUTE LOC
3774	FCA9	C9 0C	CMF ##0C
3775	FCAB	D0 05	BNE ABSOL1
3776	FCAD	A2 02	LIX #02
3777	FCAF	4C CB FC	JMP OPCOMP
3778	FCB2	A2 0C	ABSOL1 LDX ##0C
3779	FCB4	D0 15	BNE OPCOMP
3780	FCB6		;SELECT IMMEDIATE ADDRESSING TYPE
3781	FCB6	AD 2E 01	HATCH LDA TYPE
3782	FCB9	C9 01	CMF #01
3783	FCBB	F0 04	BEQ IMMEDI
3784	FCBD	A2 07	LIX #07
3785	FCBF	D0 0A	BNE OPCOMP
3786	FCC1	A2 06	IMMEDI LIX #06
3787	FCC3	D0 06	BNE OPCOMP
3788	FCC5	20 94 E3	ERRORM JSR CKEROO ;OUTPUT ERROR MESSAGE

```

LINE # LOC      CODE      LINE
3789 FCCB 4C AA FB          JMP STARTM

3791 FCCB          ; COMPUTE FINAL OP CODE FOR DEFINED ADDRESSING MODE
3792 FCCB B0 E2 FA      OPCOMP LDA TYPTR1,X ;MATCH TYPE MASK WITH VALID MODE
3793 FCCB F0 05          BEQ OPCMP1 ; PATTERNS & SKIP 1ST WORD TEST IF
3794 FCD0 2D 26 01      AND TMASK1 ;ALREADY ZERO
3795 FCD3 D0 08          BNE VALID
3796 FCD5 B0 F1 FA      OPCMP1 LDA TYPTR2,X ; TEST 2ND PART
3797 FCD8 2D 27 01      AND TMASK2
3798 FCD9 F0 EB          BEQ ERRORM ;INST DOES NOT HAVE SPECIFIED MOD
3799 FCD0 18          VALID CLC ;FORM FINAL OP CODE
3800 FCDE B0 00 FB      LDA CORR,X
3801 FCE1 6D 34 A4      ADC OPCODE
3802 FCE4 8D 34 A4      STA OPCODE

3804 FCE7          ;PROCESS ADDRESSES TO FINAL FORMAT
3805 FCE7 B0 0F FB      LDA SIZEM,X ; OBTAIN ADDRESS FORMAT FROM TABLE
3806 FCEA C9 00          CMP #00
3807 FCEC F0 50          BEQ ONEBYT
3808 FCEE C9 0F          CMP #0F ; NEED BRANCH COMPUTATION?
3809 FCF0 F0 1D          BEQ BRNCHC
3810 FCF2 8D 33 A4      STA TEMPA ; SAVE START POINT & CHAR COUNT
3811 FCF5 29 0F          AND #0F ; SEPARATE CHARACTER COUNT
3812 FCF7 A8          TAY ; LOAD ADDR BYTES INTO Y (0,1,OR 2)
3813 FCF8 8D 2F A4      STA BYTESM ; SAVE IN BYTES
3814 FCFB EE 2F A4      INC BYTESM ; TO INSTR LENGTH (1,2,OR 3 BYTES)
3815 FCFE AD 33 A4      LDA TEMPA ; SEPARATE STARTING POINT
3816 FD01 29 F0          AND #0F0
3817 FD03 4A          LSR A
3818 FD04 4A          LSR A
3819 FD05 4A          LSR A
3820 FD06 4A          LSR A
3821 FD07 AA          TAX ; AND PUT IT IN X
3822 FD08 20 12 FD      JSR CONVRT ; CONVERT ASCII ADDRESS TO HEX
3823 FD0B B0 B8          BCS ERRORM ; SKIP OUT IF ERROR IN INPUT
3824 FD0D 90 1D          BCC STASH
3825 FD0F 4C 86 FD      BRNCHC JMP BRCOMP

3827 FD12          ; ***** SUBROUTINE *****
3828 FD12          ; CONVERT FORMATTED ADDRESS INTO PROPER HEX ADDRESS
3829 FD12 B0 33 01      CONVRT LDA ADFLD,X ; PICK UP 1ST ADDR CHARACTER
3830 FD15 20 7D EA      JSR HEX ; CONVERT TO MOST SIG HEX
3831 FD18 B0 11          BCS ERRFLG
3832 FD1A EB          INX ; GET NEXT ASCII CHARACTER
3833 FD1B B0 33 01      LDA ADFLD,X
3834 FD1E EB          INX ; POINT TO NEXT CHARACTER, IF ANY
3835 FD1F 20 84 EA      JSR PACK
3836 FD22 B0 07          BCS ERRFLG
3837 FD24 99 34 A4      STA OPCODE,Y ; SAVE IN MOST SIG. BYTE LOCATION
3838 FD27 B8          DEY ; SET UP FOR NEXT ADDR BYTE, IF ANY
3839 FD28 B0 EB          BNE CONVRT ; IF NECESSARY, FORM NEXT ADDR BYTE
3840 FD2A 18          CLC
3841 FD2B 60          BRNCHC RTS ; NON HEX CLEARED CARRY
3842 FD2C          ; *****

```

```

LINE # LOC      CODE      LINE
3844 FD2C AC 2F A4      STASH LDY BYTESM ; SET UP TO STORE COMMAND
3845 FD2F B8          DEY
3846 FD30 B9 34 A4      STSHLP LDA OPCODE,Y
3847 FD33 20 78 EB      JSR SADDR ; STORE ONE BYTE OF COMMAND
3848 FD36 C0 00          CFY #00
3849 FD38 F0 0B          BEQ FORMD5
3850 FD3A B8          DEY
3851 FD3B B8          CLV
3852 FD3C 50 F2          BVC STSHLP ; REPEAT TILL THRU

3854 FD3E A9 01          ONEBYT LDA #01 ; SET BYTES = 1
3855 FD40 8D 2F A4      STA BYTESM
3856 FD43 D0 E7          BNE STASH

3858 FD45          ; FORMAT FOR SYSTEM 65 DISPLAY (REFORMAT FOR AIM)
3859 FD45 20 44 EB      FORMD5 JSR CLR
3860 FD48 20 D0 E5      JSR CGPC1 ; ADDR TO SAVPC FOR DISASSEMBLY
3861 FD4B 20 42 E8      JSR TTYTST ; IF TTY DO NOT GO TO DISASS
3862 FD4E D0 08          BNE FORMD1
3863 FD50 20 3B E8      JSR BLANK2 ; IT IS TTY
3864 FD53 20 3B E8      JSR BLANK2
3865 FD56 D0 11          BNE FORMD2 ; OUTPUT OPCODE
3866 FD58 20 6C F4      FORMD1 JSR DISASM
3867 FD5B 20 24 EA      JSR CRCK ; <CR> IF PRI PTR DIFF FROM 0
3868 FD5E AD 37 A4      LDA CODFLG ; SEE IF HE WANTS CODE ALSO
3869 FD61 F0 1A          BEQ FORM1
3870 FD63 20 3E E8      JSR BLANK
3871 FD66 20 3C F5      JSR PRFC ; PROG CNTR
3872 FD69          ; OUTPUT OPCODE
3873 FD69 AE 2F A4      FORMD2 LDY BYTESM
3874 FD6C A0 00          LDY #00
3875 FD6E A9 1C          DISPLY LDA #(ADDR ; DO LDA (ADDR),Y ,WHITOUT PAG 0
3876 FD70 20 58 EB      JSR LIAY
3877 FD73 20 46 EA      JSR NUMA
3878 FD76 20 3E E8      JSR BLANK
3879 FD79 C8          INY
3880 FD7A CA          DEX
3881 FD7B D0 F1          BNE DISPLY

3883 FD7D          ; POINT TO NEXT INSTRUCTION LOCATION
3884 FD7D AC 2F A4      FORM1 LDY BYTESM ; ADD BYTESM TO ADDR
3885 FD80 20 CD E2      JSR NXTADD
3886 FD83 4C 24 FF      JMP PATC16 ; UPDATE PC

3888 FD86          ; RELATIVE BRANCH ADDRESS COMPUTATION
3889 FD86 AD 31 A4      BRCOMP LDA TEMPX
3890 FD89 C9 02          CMP #02 ; IF REL BRANCH INPUT, USE IT
3891 FD8B D0 11          BNE COMPF8
3892 FD8D A2 00          LDX #00
3893 FD8F A0 01          LDY #01
3894 FD91 20 12 FD      JSR CONVRT
3895 FD94 B0 40          BCS ERRJMP
3896 FD96 A9 02          LDA #02
3897 FD98 B0 2F A4      STA BYTESM ; SET PROPER BYTES
3898 FD9B 4C 2C FD      JMP STASH

```

LINE #	LOC	CODE	LINE
3899	F09E	A2 00	COMPBR LDX #00
3900	FDA0	A0 02	LDY #02
3901	FDA2	20 12 FD	JSR CONVRT
3902	FDA5	B0 2F	BCS ERRJMP
3903	FDA7	AD 1D A4	LDA ADDR+1 ;ADD BRANCH OFFSET
3904	FDA8	BD 27 01	STA MOVAD+1
3905	FDA1	AD 1C A4	LDA ADDR
3906	FDB0	18	CLC
3907	FDB1	69 02	ADC #02
3908	FDB3	8D 26 01	STA MOVAD
3909	FDB6	90 03	BCC CMPBR1
3910	FDB8	EE 27 01	INC MOVAD+1
3911	FDBB	38	CMPBR1 SEC ;COMPUTE BRANCH RELATIVE ADDRESS
3912	FDBC	AD 35 A4	LDA OFCODE+1
3913	FDBF	ED 26 01	SRC MOVAD
3914	FDC2	8D 35 A4	STA OFCODE+1
3915	FDC5	AD 36 A4	LDA OFCODE+2
3916	FDC8	ED 27 01	SRC MOVAD+1
3917	FDCB	8D 36 A4	STA OFCODE+2
3918	FDC E	C9 00	CMP #00
3919	FDD0	F0 0E	BEQ FORWRD
3920	FDD2	C9 FF	CMP #FF
3921	FDD4	F0 03	BEQ BACKWD
3922	FDD6	4C C5 FC	ERRJMP JMP ERRORM
3923	FDD9	AD 35 A4	BACKWD LDA OFCODE+1 ;CHECK IN RANGE
3924	FDDC	30 09	BMI OK
3925	FDD E	10 F6	BPL ERRJMP
3926	FDE0	AD 35 A4	FORWRD LDA OFCODE+1
3927	FDE3	10 02	BPL OK
3928	FDE5	30 EF	BMI ERRJMP
3929	FDE7	A9 02	OK LDA #02 ;SET UP FOR STASH
3930	FDE9	8D 2F A4	STA BYTESM
3931	FDEC	4C 2C FD	JMP STASH
3933	FDEF		;***** SUBROUTINE *****
3934	FDEF		;SUBROUTINE FOR DETERMINING X OR Y OR NEITHER
3935	FDEF	A2 04	XORY LDX #04
3936	FDF1	BD 33 01	XORYZ LDA ADFLD,X
3937	FDF4	C9 2C	CMP #','
3938	FDF6	D0 04	BNE XORY1
3939	FDF8	E8	INX
3940	FDF9	BD 33 01	LDA ADFLD,X
3941	FDFC	C9 58	XORY1 CMP #'X'
3942	FDFE	F0 03	BEQ ISX
3943	FE00	C9 59	CMP #'Y'
3944	FE02		XORYRT
3945	FE02	60	RTS; NOT ZERO IS NOT X OR NOT Y
3946	FE03	18	CLC ;CARRY SET IS Y
3947	FE04	90 FC	BCC XORYRT ; CARRY CLEAR IS X
3948	FE06		;***** END OF SUB *****
3950	FE06		; INPUT FOR MNEMONIC CODE
3951	FE06	A0 00	MNEM LDY #00
3952	FE08	8C 34 A4	STY OFCODE
3953	FE0B	8C 35 A4	STY OFCODE+1

LINE #	LOC	CODE	LINE
3954	FE0E	8C 36 A4	STY OFCODE+2 ;CLEARS OPCODE FOR NEW INPUT
3955	FE11	8C 26 01	STY MOVAD ;CLEARS UNUSED BIT IN FINAL FORMAT
3956	FE14	20 5F E9	JSR RDLUP
3957	FE17	C9 2A	CMP #'*'
3958	FE19	F0 58	BEQ STLOAD ;GO TO SET CURRENT ADDRESS POINTER
3959	FE1B	C9 20	CMP #*20 ;IGNORE SPACE BAR INPUT
3960	FE1D	F0 F5	BEQ RDLUP
3961	FE1F	29 1F	AND #*1F ;MASK OFF UPPER 3 BITS
3962	FE21	99 30 01	STA CH,Y
3963	FE24	98	TYA
3964	FE25	AA	TAX ;Y----> X
3965	FE26	FE 30 01	INC CH,X ;FORMAT TO MATCH DISASSEMBLER TAB
3966	FE29	CB	INY
3967	FE2A	C0 03	CPY #03 ;REPEAT FOR EACH OF 3 CHARACTERS
3968	FE2C	D0 E6	BNE RDLUP
3970	FE2E		;COMPRESS 3 FORMATTED CHARACTERS TO MOVAD & MOVAD+1
3971	FE2E	A0 03	LDY #03 ;SET UP OUTER LOOP
3972	FE30	B9 2F 01	OUTLUP LDA CH-1,Y ;COMPRESS 3 CHARACTERS
3973	FE33	A2 05	LDX #05 ;SET UP INNER LOOP
3974	FE35	4A	INLUP LSK A ;SHIFT 5 BITS ACC TO MOVAD,MOVAD+1
3975	FE36	6E 26 01	ROR MOVAD
3976	FE39	6E 27 01	ROR MOVAD+1
3977	FE3C	CA	DEX
3978	FE3D	D0 F6	BNE INLUP
3979	FE3F	88	DEY
3980	FE40	D0 EE	BNE OUTLUP
3982	FE42		;SEARCH FOR MATCHING COMPRESSED CODE
3983	FE42	A2 40	LDX #*40
3984	FE44	AD 26 01	SRCHLP LDA MOVAD
3985	FE47	DD B8 F5	SRCHM CMP MNEML-1,X ;MATCH LEFT HALF
3986	FE4A	F0 05	BEQ MATCH
3987	FE4C	CA	DEX
3988	FE4D	D0 FB	BNE SRCHM ;IF NO - TRY AGAIN
3989	FE4F	F0 0B	BEQ MATCH1
3990	FE51	AD 27 01	MATCH LDA MOVAD+1 ;ALSO MATCH RIGHT HALF
3991	FE54	DD FB F5	CMP MNEMR-1,X
3992	FE57	F0 06	BEQ GOTIT
3993	FE59	CA	DEX
3994	FE5A	D0 EB	BNE SRCHLP
3995	FE5C	4C C5 FC	MATCH1 JMP ERRORM
3997	FE5F		;GET INSTRUCTION TYPE FROM TYPE TABLE
3998	FE5F	BD 5D FB	GOTIT LDA TYPTB-1,X
3999	FE62	BD 2E 01	STA TYPE
4001	FE65		;GET OPCODE FROM OP CODE UE
4002	FE65	BD 1D FB	LDA STCODE-1,X
4003	FE68	BD 34 A4	STA OPCODE
4004	FE6B	4C C1 FB	JMP MODEM
4006	FE6E		;THIS SECTION SETS THE CURRENT ADDRESS POINTER
4007	FE6E	A9 2A	STLO LDA #'*

LINE #	LOC	CODE	LINE	
4008	FE70	20 7A E9		JSR OUTPUT
4009	FE73	20 AE EA	STLOAD	JSR ADDIN ;GET ADDR
4010	FE76	B0 F6		BCS STLO ;IN CASE OF ERROR
4011	FE78	4C 24 FF		JMP PATC16 ;ADDR TO PC THEN TO STARTM
4013	FE7B			;PATCHES TO CORRECT PROBLEMS WITHOUT
4014	FE7B			;CHANGING ENTRY POINTS TO THE ROUTINES
4015	FE7B	41		BYT 'A'
4016	FE7C	38	PATCH1	SEC ;ADJUST BAUD
4017	FE7D	E9 2C		SBC #44
4018	FE7F	8D 18 A4		STA CNTL30
4019	FEB2	60		RTS
4021	FEB3	8A	CUREAD	TXA ;SAVE X , OUTPUT CUR
4022	FEB4	48		PHA
4023	FEB5	AE 15 A4		LIX CURP02
4024	FEB8	E0 14		CPX #20 ;ONLY IF C 20
4025	FEB8	B0 05		BCS PAT2A
4026	FEB8	A9 DE		LDA #*DE
4027	FEBE	20 7B EF		JSR OUTD1
4028	FE91	68	PAT2A	PLA
4029	FE92	AA		TAX
4030	FE93	4C 3C E9		JMP READ ;CONTINUE
4032	FE96	20 3C E9	RED1	JSR READ ;READ & ECHO WITHOUT CURSOR
4033	FE99	4C 76 E9		JMP RED2
4035	FE9C	AE 15 A4	PATCH4	LIX CURP02 ;DONT DO ANYTHING IF *BD*
4036	FE9F	C9 8D		CMP #*BD ;SO (CR) FOR TV & NOT FOR DISP
4037	FEA1	D0 0B		BNE PAT4A
4038	FEA3	A9 A0		LDA #*A0 ;CLR CURSOR
4039	FEA5	20 7B EF		JSR OUTD1
4040	FEA8	20 44 EB		JSR CLR ;CLR PNTRS
4041	FEAB	4C 76 EF		JMP OUTD7 ;EXIT
4042	FEAE	4C 17 EF	PAT4A	JMP OUTD1A ;CONTINUE
4044	FEB1	8D 11 A4	PATCH5	STA PRIFLG ;TURN PRI OFF
4045	FEB4	4C 73 F0		JMP IPO3
4047	FEB7	A9 1C	PATCH6	LDA #*ADDR ;SIMULATE LDA (ADDR),Y
4048	FEB9	4C 58 EB		JMP LDAY
4050	FEB8	20 3C E9	PATCH8	JSR READ ;READ & ECHO WITH CARROTS
4051	FEBF	48		PHA
4052	FEC0	20 D8 E7		JSR EQUAL
4053	FEC3	A9 3C		LDA #'C
4054	FEC5	20 7A E9		JSR OUTPUT
4055	FEC8	68		PLA
4056	FEC9	48		PHA
4057	FECA	C9 0D		CMP #CR
4058	FECB	F0 03		BEQ PATCBC
4059	FECE	20 7A E9		JSR OUTPUT
4060	FED1	A9 3E	PATCBC	LDA #'C

LINE #	LOC	CODE	LINE	
4061	FED3	20 7A E9		JSR OUTPUT
4062	FED6	68		PLA
4063	FED7	60		RTS
4065	FED8	C9 F7	PATCH9	CMP #*F7 ;CHCK LOWER TRANSITION OF TIMER
4066	FEDA	B0 06		BCS PAT9A
4067	FEDC	CD 08 A4		CMP TSPEED
4068	FEDF	4C 9D EE		JMP CKF3A
4069	FEE2	CD 08 A4	PAT9A	CMP TSPEED
4070	FEE5	68		PLA
4071	FEE6	C9 FF		CMP #*FF
4072	FEE8	60	PAT9B	RTS
4074	FEE9	20 F0 E9	PATC10	JSR CRLF ;CLR DISP (ONLY 1 (CR))
4075	FEEC	4C 85 E1		JMP STA1
4077	FEF7	F0 F7	PATC11	BEQ PAT9B ;GO OUTPUT PROMPT
4078	FEF1	C9 4C		CMP #'L ;NO PROMPT FOR 'T' OR 'L'
4079	FEF3	F0 F3		BEQ PAT9B
4080	FEF5	4C C5 E7		JMP PROMP1
4082	FEF8	48	PATC12	PHA ;CLEAR PRIFLG SO WE CAN OUTPUT
4083	FEF9	AD 11 A4		LDA PRIFLG ;TO PRINTER IF FLG WAS ON (MSB)
4084	FEFC	29 F0		AND #*F0
4085	FEFE	8D 11 A4		STA PRIFLG
4086	FF01	68		PLA
4087	FF02	60		RTS
4089	FF03	AD 12 A4	PATC13	LDA INFLG ;TURN TAPES ON ONLY IF TAPES
4090	FF06	C9 54		CMP #'T
4091	FF08	D0 DE		BNE PAT9B
4092	FF0A	4C 29 E5		JMP DU14 ;TURN ON TAPES & SET DEF DEV
4094	FF0D	AD 13 A4	PATC14	LDA OUTFLG ;TURN ON TAPES ONLY IF TAPES
4095	FF10	C9 54		CMP #'T
4096	FF12	D0 D4		BNE PAT9B
4097	FF14	4C 0A E5		JMP DU11
4099	FF17	20 F0 E9	PATC15	JSR CRLF ;DECODE COMMAND
4100	FF1A	8A		TXA ;SAVE INDEX
4101	FF1B	0A		ASL A
4102	FF1C	AA		TAX
4103	FF1D	BD B8 FA		LDA JTRL,X ;PART OF ENTRY
4104	FF20	8D 1A A4		STA S1
4105	FF23	60		RTS
4107	FF24	20 D8 E5	PATC16	JSR CGPC1 ;ADDR TO PC
4108	FF27	4C AA FB		JMP STARTM ;BACK TO MNEMONIC START
4110	FF2A	F0 0E	PATC17	BEQ PAT17B ;RUB ,SO READ ANOTHER
4111	FF2C	C9 00		CMP #0
4112	FF2E	F0 03		BEQ PAT17A
4113	FF30	4C B5 F7		JMP INO2A
4114	FF33	20 93 E9	PAT17A	JSR INALL ;NEITHER ,CONTINUE
4115	FF36	C9 7F		CMP #*7F ;SKIP ON ZEROS
				;UNTILL RUB



LINE #	LOC	CODE	LINE	
4116	FF38	D0 F9	RNE PAT17A	
4117	FF3A	4C 7A F7	PAT17B JMP IN02	;GO BACK
4119	FF3D	20 F8 FE	PATC18 JSR PATC12	;RESET PRIFLG
4120	FF40	48	PHA	
4121	FF41	20 42 EB	JSR TTYTST	;IF TTY JUST RTN
4122	FF44	D0 02	RNE PAT18A	
4123	FF46	68	PLA	
4124	FF47	60	RTS	
4125	FF48	20 FE EB	PAT18A JSR LL	;SET TO LOW SPEED
4126	FF4B	20 45 F0	JSR IPST	;PRINT WHAT IS IN BUFFER
4127	FF4E	20 44 EB	JSR CLR	;CLR PRINTER BUFFER BY OUTPUTTING
4128	FF51	20 3E EB	JSR BLANK	;AN SPACE
4129	FF54	20 44 EB	JSR CLR	
4130	FF57	68	PLA	;RTN ACC
4131	FF58	60	RTS	
4133	FF59	D8	PAT19 CLD	
4134	FF5A	20 24 EA	JSR CRCK	
4135	FF5D	4C 85 E1	JMP STA1	
4137	FF60	F0 0D	PAT20 BEQ VECK4	;END (DATA BYTES=0)
4138	FF62	18	CLC	
4139	FF63	69 04	ADC #4	
4140	FF65	AA	TAX	
4141	FF66	20 93 E9	VECK5 JSR INALL	;SKIP OVER DATA
4142	FF69	CA	DEX	
4143	FF6A	D0 FA	RNE VECK5	
4144	FF6C	4C 9E E6	JMP VECK1	;PROCESS NEXT RCD
4145	FF6F	4C 20 E5	VECK4 JMP DU13	
4147	FF72	A0 00	PAT21 LDY #0	
4148	FF74	B9 88 FF	PAT21A LDA POMSG,Y	;RESET MSG
4149	FF77	F0 06	BEQ PAT21B	
4150	FF79	20 7A E9	JSR OUTPUT	
4151	FF7C	C8	INY	
4152	FF7D	D0 F5	RNE PAT21A	
4153	FF7F	20 F0 E9	PAT21B JSR CRLF	
4154	FF82	20 F0 E9	JSR CRLF	
4155	FF85	4C 82 E1	JMP START	
4157	FF88	20 20	POMSG .BYT ' ROCKWELL AIM 65'	
4158	FF99	00	.BYT 0	
4160	FF9A	EE 68 01	PAT22 INC BLKO	
4161	FF9D	4C BD ED	JMP ADDBK1	
4162	FFA0			
4163	FFA0	A9 FF	PAT23 LDA #FF	;START TIMER
4164	FFA2	BD 97 A4	STA DI1024	
4165	FFA5	AD 85 A4	PAT23A LDA RINT	;TIME OUT?
4166	FFAB	30 08	BMI PAT23B	;YES
4167	FFAA	AD 0D AB	LDA IFR	;START SIGNAL?
4168	FFAD	29 10	AND #MPRST	
4169	FFAF	F0 F4	BEQ PAT23A	;NO
4170	FFB1	60	RTS	;YES

LINE #	LOC	CODE	LINE	
4171	FFB2	A9 00	PAT23B LDA #0	;TIME OUT RETURN
4172	FFB4	60	RTS	
4174	FFB5	20 75 EE	PATC24 JSR CKFREQ	;READ BIT FROM FOURTH HALF PULSE
4175	FFB8	6A	ROR A	
4176	FFB9	29 80	AND #80	
4177	FFBB	60	RTS	
4179	FFBC	2C 0D AB	PATC25 BIT IFR	;WAIT TILL TIMES OUT
4180	FFBF	50 FB	BVC PATC25	
4181	FFC1	AD 04 AB	LDA T1L	;CLR INTERRUPT FLG
4182	FFC4	60	RTS	
4184	FFC5		*=\$FFF9	
4185	FFF9		;INTERRUPT VECTORS	
4186	FFF9	FA	.BYT \$FA	
4187	FFFA	75 E0	.WORD NMIV1,RSET,IRQV1	;SET UP VECTORS
4187	FFFC	BF E0		
4187	FFFE	7B E0		
4188	0000		.END A0/1	

ERRORS = 0000 (0000)