

APPENDIX I

KIM-1 PROGRAM LISTINGS

CARD #	LOC	CODE	CARD
3			666666 555555 333333 000000
4			6 5 3 0 0
5			6 5 3 0 0
6			666666 555555 333333 0 0
7			6 6 5 3 0 0
8			6 6 5 3 0 0
9			666666 555555 333333 000000
10			
11			
12			
13			000000 000000 333333
14			0 0 0 0 3
15			----- 0 0 0 0 3
16			----- 0 0 0 0 333333
17			----- 0 0 0 0 3
18			0 0 0 0 3
19			000000 000000 333333
20			
21			
22			
23			
24			
25			COPYRIGHT
26			MOS TECHNOLOGY, INC
27			DATE OCT 18 1975 REV D
28			
29			
30			
31			6530-003 IS AN AUDIO CASSETT TAPE
32			RECORDER EXTENSION OF THE BASIC
33			KIM MONITOR
34			
35			IT FEATURES TWO BASIC ROUTINES
36			LOADT-LOAD MEM FROM AUDIO TAPE
37			DUMPT-STOR MEM ONTO AUDIO TAPE
38			
39			LOADT
40			ID=00 IGNORE ID
41			ID=FF IGN. ID USE SA FOR START ADDR
42			ID=01-FE IGN.ID USE ADDR ON TAPE
43			
44			DUMPT
45			ID=00 SHOULD NOT BE USED
46			ID=FF SHOULD NOT BE USED
47			ID=01-FE NORMAL ID RANGE
48			SAL LSB STARTING ADDRESS
49			SAH MSB
50			EAL LSB ENDING ADDRESS
51			EAH MSB
52			

CARD #	LOC	CODE	CARD
54			;
55			;
56			;
57			;
58			SAD = \$1740 6530 A DATA
59			PADD = \$1741 6530 A DATA DIRECTION
60			SBD = \$1742 6530 B DATA
61			PBDD = \$1743 6530 B DATA DIRECTION
62			CLK1T = \$1744 DIV BY 1 TIME
63			CLK8T = \$1745 DIV BY 8 TIME
64			CLK64T = \$1746 DIV BY 64 TIME
65			CLKKT = \$1747 DIV BY 1024 TIME
66			CLKRDI = \$1747 READ TIME OUT BIT
67			CLKRDT = \$1746 READ TIME
68			;
69	0000		*=\$00EF
70			;
71			MPU REG. SAVX AREA IN PAGE 0
72	00EF		PCL *=\$+1 PROGRAM CNT LOW
73	00F0		PCH *=\$+1 PROGRAM CNT HI
74	00F1		PREG *=\$+1 CURRENT STATUS REG.
75	00F2		SPUSER *=\$+1 CURRENT STACK POINT
76	00F3		ACC *=\$+1 ACCUMULATOR
77	00F4		YREG *=\$+1 Y INDEX
78	00F5		XREG *=\$+1 X INDEX
79			;
80			;
81			KIM FIXED AREA IN PAGE 0
82	00F6		CHKHI *=\$+1
83	00F7		CHKSUM *=\$+1
84	00F8		INL *=\$+1 INPUT BUFFER
85	00F9		INH *=\$+1 INPUT BUFFER
86	00FA		POINTL *=\$+1 LSB OF OPEN CELL
87	00FB		POINTH *=\$+1 MSB OF OPEN CELL
88	00FC		TEMP *=\$+1
89	00FD		TMPX *=\$+1
90	00FE		CHAR *=\$+1
91	00FF		MODE *=\$+1
92			;
93			;
94			KIM FIXED AREA IN PAGE 23
95	0100		*=\$17E7
96	17E7		CHKL *=\$+1
97	17E8		CHKH *=\$+1 CHKSUM
98	17E9		SAVX *=\$+3
99	17EC		VEB *=\$+6 VOLATILE EXECUTION BLOCK
100	17F2		CNTL30 *=\$+1 TTY DELAY
101	17F3		CNTH30 *=\$+1 TTY DELAY
102	17F4		TIMH *=\$+1
103	17F5		SAL *=\$+1 LOW STARTING ADDRESS
104	17F6		SAH *=\$+1 HI STARTING ADDRESS
105	17F7		EAL *=\$+1 LOW ENDING ADDRESS

CARD #	LOC	CODE	CARD	
106	17F8		EAH	◆=◆+1 HI ENDING ADDRESS
107	17F9		ID	◆=◆+1
108			;	
109			;	INTERRUPT VECTORS
110			;	
111	17FA		NMIV	◆=◆+2 STOP VECTOR (STOP=1C00)
112	17FC		RSTV	◆=◆+2 RST VECTOR
113	17FE		IRQV	◆=◆+2 IRQ VECTOR (BRK= 1C00)
114			;	

CARD #	LOC	CODE	CARD
116	1800		♦=\$1800
117			;
118			INIT VOLATILE EXECUTION BLOCK
119			DUMP MEM TO TAPE
120			;
121	1800	A9 AD	DUMPT LDA #BAD LOAD ABSOLUTE INST
122	1802	8D EC 17	STA VEB
123	1805	20 32 19	JSR INTVEB
124			;
125	1808	A9 27	LDA #B27 TURN OFF DATAIN PB5
126	180A	8D 42 17	STA SB0
127	180D	A9 BF	LDA #BBF CONVERT PB7 TO OUTPUT
128	180F	8D 43 17	STA PBDD
129			;
130	1812	A2 64	LDX #B64 100 CHAR'S
131	1814	A9 16	DUMPT1 LDA #B16 SYN CHAR'S
132	1816	20 7A 19	JSR OUTCHT
133	1819	CA	DEX
134	181A	D0 F8	BNE DUMPT1
135			;
136			;
137	181C	A9 2A	LDA #1♦ START CHAR
138	181E	20 7A 19	JSR OUTCHT
139			;
140	1821	AD F9 17	LDA ID OUTPUT ID
141	1824	20 61 19	JSR OUTBT
142			;
143	1827	AD F5 17	LDA SAL OUTPUT STARTING
144	182A	20 5E 19	JSR OUTBTC ADDRESS
145	182D	AD F6 17	LDA SAH
146	1830	20 5E 19	JSR OUTBTC
147			;
148	1833	AD ED 17	DUMPT2 LDA VEB+1 CHECK FOR LAST
149	1836	CD F7 17	CMP EAL DATA BYTE
150	1839	AD EE 17	LDA VEB+2
151	183C	ED F8 17	SBC EAH
152	183F	90 24	BCC DUMPT4
153			;
154	1841	A9 2F	LDA #1/ OUTPUT END OF DATA CHR
155	1843	20 7A 19	JSR OUTCHT
156	1846	AD E7 17	LDA CHKL LAST BYTE HAS BEEN
157	1849	20 61 19	JSR OUTBT OUT PUT NOW OUTPUT
158	184C	AD E8 17	LDA CHKH CHKSUM
159	184F	20 61 19	JSR OUTBT
160			;
161			;
162	1852	A2 02	LDX #B02 2 CHAR'S
163	1854	A9 04	DUMPT3 LDA #B04 EOT CHAR
164	1856	20 7A 19	JSR OUTCHT
165	1859	CA	DEX
166	185A	D0 F8	BNE DUMPT3
167			;

CARD #	LOC	CODE	CARD		
168	185C	A9 00	LDA	#\$00	DISPLAY 0000
169	185E	85 FA	STA	POINTL	FOR NORMAL EXIT
170	1860	85 FB	STA	POINTH	
171	1862	4C 4F 1C	JMP	START	
172					
173	1865	20 EC 17	DUMPT4 JSR	VEB	DATA BYTE OUTPUT
174	1868	20 5E 19	JSR	OUTBTC	
175					
176	186B	20 EA 19	JSR	INCVB	
177	186E	4C 33 18	JMP	DUMPT2	
178					
179				LOAD MEMORY FROM TAPE	
180					
181					
182	1871	0F 19	TAB	.WORD	LOAD12
183	1873	A9 8D	LOADT LDA	#\$8D	INIT VOLATILE EXECUTION
184	1875	8D EC 17	STA	VEB	BLOCK WITH STA ABS.
185	1878	20 32 19	JSR	INTVEB	
186					
187	187B	A9 4C	LDA	#\$4C	JUMP TYPE RTRN
188	187D	8D EF 17	STA	VEB+3	
189	1880	AD 71 18	LDA	TAB	
190	1883	8D F0 17	STA	VEB+4	
191	1886	AD 72 18	LDA	TAB+1	
192	1889	8D F1 17	STA	VEB+5	
193					
194	188C	A9 07	LDA	#\$07	RESET PBS=0 (DATA IN)
195	188E	8D 42 17	STA	SBD	
196					
197	1891	A9 FF	SYNC LDA	#\$FF	CLEAR SAVX FOR SYNC AREA
198	1893	8D E9 17	STA	SAVX	
199					
200	1896	20 41 1A	SYNC1 JSR	RDBIT	GET A BIT
201	1899	4E E9 17	LSR	SAVX	SHIFT BIT INTO CHAR
202	189C	0D E9 17	ORA	SAVX	
203	189F	8D E9 17	STA	SAVX	
204	18A2	AD E9 17	LDA	SAVX	GET NEW CHAR
205	18A5	C9 16	CMP	#\$16	SYN CHAR
206	18A7	D0 ED	BNE	SYNC1	
207					
208	18A9	A2 0A	LDX	#\$0A	TEST FOR 10 SYN CHARS
209	18AB	20 24 1A	SYNC2 JSR	RDCHT	
210	18AE	C9 16	CMP	#\$16	
211	18B0	D0 DF	BNE	SYNC	IF NOT 10 CHAR RE-SYNC
212	18B2	0A	DEX		
213	18B3	D0 F6	BNE	SYNC2	
214					
215					
216	18B5	20 24 1A	LOADT4 JSR	RDCHT	LOOK FOR START OF
217	18B8	C9 2A	CMP	#\$16	DATA CHAR
218	18BA	F0 06	BEQ	LOAD11	
219	18BC	C9 16	CMP	#\$16	IF NOT 16 SHOULD BE SYN

CARD #	LOC	CODE	CARD		PAGE 7
220	18BE	D0 D1	BNE	SYNC	
221	18C0	F0 F3	BEQ	LOADT4	
222					
223	18C2	20 F3 19	LOADT11 JSR	RDBYT	READ ID FROM TAPE
224	18C5	CD F9 17	CMP	ID	COMPARE WITH REQUESTED ID
225	18C8	F0 00	BEQ	LOADT5	
226	18CA	AD F9 17	LDA	ID	DEFAULT 00 READ RECORD
227	18CD	C9 00	CMP	#\$00	ANYWAY
228	18CF	F0 06	BEQ	LOADT5	
229	18D1	C9 FF	CMP	#\$FF	DEFAULT FF IGNORE SA ON
230	18D3	F0 17	BEQ	LOADT6	TAPE
231	18D5	D0 9C	BNE	LOADT	
232					
233	18D7	20 F3 19	LOADT5 JSR	RDBYT	GET SA FROM TAPE
234	18DA	20 4C 19	JSR	CHKT	
235	18DD	8D ED 17	STA	VEB+1	SAVX IN VEB+1,2
236	18E0	20 F3 19	JSR	RDBYT	
237	18E3	20 4C 19	JSR	CHKT	
238	18E6	8D EE 17	STA	VEB+2	
239	18E9	4C F8 18	JMP	LOADT7	
240					
241	18EC	20 F3 19	LOADT6 JSR	RDBYT	GET SA BUT IGNORE
242	18EF	20 4C 19	JSR	CHKT	
243	18F2	20 F3 19	JSR	RDBYT	
244	18F5	20 4C 19	JSR	CHKT	
245					
246					
247	18F8	A2 02	LOADT7 LDX	#\$02	GET 2 CHARS
248	18FA	20 24 1A	LOADT13 JSR	RDCHT	GET CHAR(X)
249	18FD	C9 2F	CMP	##	LOOK FOR LAST CHAR
250	18FF	F0 14	BEQ	LOADT8	
251	1901	20 00 1A	JSR	PACKT	CONVERT TO HEX
252	1904	D0 23	BNE	LOADT9	Y=1 NON-HEX CHAR
253	1906	CA	DEX		
254	1907	D0 F1	BNE	LOADT13	
255					
256	1909	20 4C 19	JSR	CHKT	COMPUTE CHECKSUM
257	190C	4C EC 17	JMP	VEB	SAVX DATA IN MEMORY
258	190F	20 5A 19	LOADT12 JSR	INCVEB	INCREMENT DATA POINTER
259	1912	4C F8 18	JMP	LOADT7	
260					
261	1915	20 F3 19	LOADT8 JSR	RDBYT	END OF DATA COMPARE CHKSUM
262	1918	CD E7 17	CMP	CHKL	
263	191B	D0 0C	BNE	LOADT9	
264	191D	20 F3 19	JSR	RDBYT	
265	1920	CD E8 17	CMP	CHKH	
266	1923	D0 04	BNE	LOADT9	
267	1925	A9 00	LDA	#\$00	NORMAL EXIT
268	1927	F0 02	BEQ	LOADT10	
269					
270	1929	A9 FF	LOADT9 LDA	#\$FF	ERROR EXIT
271	192B	85 FA	LOADT10 STA	POINTL	

CARD #	LOC	CODE	CARD		PAGE 8
272	192D	85 FB	STA	POINTH	
273	192F	4C 4F 1C	JMP	START	
274					

CARD #	LOC	CODE	CARD		PAGE	7
220	18BE	D0 D1	BNE	SYNC		
221	18C0	F0 F3	BEQ	LOADT4		
222						
223	18C2	20 F3 19	LOADT11 JSR	RDBYT	READ ID FROM TAPE	
224	18C5	CD F9 17	CMP	ID	COMPARE WITH REQUESTED ID	
225	18C8	F0 00	BEQ	LOADT5		
226	18CA	AD F9 17	LDA	ID	DEFAULT 00 READ RECORD	
227	18CD	C9 00	CMP	#\$00	ANYWAY	
228	18CF	F0 06	BEQ	LOADT5		
229	18D1	C9 FF	CMP	#\$FF	DEFAULT FF IGNORE SA ON	
230	18D3	F0 17	BEQ	LOADT6	TAPE	
231	18D5	D0 9C	BNE	LOADT		
232						
233	18D7	20 F3 19	LOADT5 JSR	RDBYT	GET SA FROM TAPE	
234	18DA	20 4C 19	JSR	CHKT		
235	18DD	8D ED 17	STA	VEB+1	SAVX IN VEB+1,2	
236	18E0	20 F3 19	JSR	RDBYT		
237	18E3	20 4C 19	JSR	CHKT		
238	18E6	8D EE 17	STA	VEB+2		
239	18E9	4C F8 18	JMP	LOADT7		
240						
241	18EC	20 F3 19	LOADT6 JSR	RDBYT	GET SA BUT IGNORE	
242	18EF	20 4C 19	JSR	CHKT		
243	18F2	20 F3 19	JSR	RDBYT		
244	18F5	20 4C 19	JSR	CHKT		
245						
246						
247	18F8	A2 02	LOADT7 LDX	#\$02	GET 2 CHARS	
248	18FA	20 24 1A	LOADT13 JSR	RDCHT	GET CHAR(X)	
249	18FD	C9 2F	CMP	##	LOOK FOR LAST CHAR	
250	18FF	F0 14	BEQ	LOADT8		
251	1901	20 00 1A	JSR	PACKT	CONVERT TO HEX	
252	1904	D0 23	BNE	LOADT9	Y=1 NON-HEX CHAR	
253	1906	CA	DEX			
254	1907	D0 F1	BNE	LOADT13		
255						
256	1909	20 4C 19	JSR	CHKT	COMPUTE CHECKSUM	
257	190C	4C EC 17	JMP	VEB	SAVX DATA IN MEMORY	
258	190F	20 5A 19	LOADT12 JSR	INCVEB	INCREMENT DATA POINTER	
259	1912	4C F8 18	JMP	LOADT7		
260						
261	1915	20 F3 19	LOADT8 JSR	RDBYT	END OF DATA COMPARE CHKSUM	
262	1918	CD E7 17	CMP	CHKL		
263	191B	D0 0C	BNE	LOADT9		
264	191D	20 F3 19	JSR	RDBYT		
265	1920	CD E8 17	CMP	CHKH		
266	1923	D0 04	BNE	LOADT9		
267	1925	A9 00	LDA	#\$00	NORMAL EXIT	
268	1927	F0 02	BEQ	LOADT10		
269						
270	1929	A9 FF	LOADT9 LDA	#\$FF	ERROR EXIT	
271	192B	85 FA	LOADT10 STA	POINTL		

CARD #	LOC	CODE	CARD		PAGE	8
272	192D	85 FB	STA	POINTH		
273	192F	4C 4F 1C	JMP	START		
274						


```

CARD # LOC      CODE      CARD
276
277              ;
278              ;      SUBROUTINES FOLLOW
279              ;
280              ;      SUB TO MOVE SA TO VEB+1,2
281 1932 AD F5 17  INTVEB LDA   SAL
282 1935 8D ED 17      STA   VEB+1
283 1938 AD F6 17      LDA   SAH
284 193B 8D EE 17      STA   VEB+2
285 193E A9 60          LDA   #$60      RTS INST
286 1940 8D EF 17      STA   VEB+3
287 1943 A9 00          LDA   #$00      CLEAR CHKSUM AREA
288 1945 8D E7 17      STA   CHKL
289 1948 8D E8 17      STA   CHKH
290 194B 60            RTS
291
292              ;
293              ;      COMPUTE CHKSUM FOR TAPE LOAD
294              ;      RTN USES Y TO SAVX A
295 194C A8            CHKT  TAY
296 194D 18            CLC
297 194E 6D E7 17      ADC   CHKL
298 1951 8D E7 17      STA   CHKL
299 1954 AD E8 17      LDA   CHKH
300 1957 69 00          ADC   #$00
301 1959 8D E8 17      STA   CHKH
302 195C 98            TYA
303 195D 60            RTS
304
305              ;
306              ;      OUTPUT ONE BYTE USE Y
307              ;      TO SAVX BYTE
308 195E 20 4C 19  OUTBTC JSR   CHKT      COMP CHKSUM
309 1961 A8            OUTBT TAY      SAVX DATA BYTE
310 1962 4A            LSR   A          SHIFT OFF LSD
311 1963 4A            LSR   A
312 1964 4A            LSR   A
313 1965 4A            LSR   A
314 1966 20 6F 19  JSR   HEXOUT      OUT PUT MSD
315 1969 98            TYA
316 196A 20 6F 19  JSR   HEXOUT      OUT PUT LSD
317 196D 98            TYA
318 196E 60            RTS
319
320              ;
321              ;      CONVERT LSD OF A TO ASCII
322              ;      AND OUTPUT TO TAPE
323 196F 29 0F      HEXOUT AND   #$0F
324 1971 09 0A      CMP   #$0A
325 1973 18            CLC
326 1974 30 03      BMI   HEX1
327 1976 69 07      ADC   #$07

```

CARD #	LOC	CODE	CARD			
328	1978	69 30	HEX1	ADC	#B30	
329						
330						
331						
332						
333	197A	8E E9 17	OUTCHT	STX	SAVX	
334	197D	8C EA 17		STY	SAVX+1	
335	1980	A0 08		LDY	#B09	START BIT
336	1982	20 9E 19	CHT1	JSR	ONE	
337	1985	4A		LSR	A	GET DATA BIT
338	1986	B0 06		BCS	CHT2	
339	1988	20 9E 19		JSR	ONE	DATA BIT=1
340	198B	4C 91 19		JMP	CHT3	
341	198E	20 C4 19	CHT2	JSR	ZRD	DATA BIT=0
342	1991	20 C4 19	CHT3	JSR	ZRD	
343	1994	88		DEY		
344	1995	D0 EB		BNE	CHT1	
345	1997	AE E9 17		LDX	SAVX	
346	199A	AC EA 17		LDY	SAVX+1	
347	199D	60		RTS		
348						
349						
350						
351						
352						
353	193E	A2 09	ONE	LDX	#B09	
354	19A0	48		PHA		SAVX A
355	19A1	2C 47 17	ONE1	BIT	CLKRDI	WAIT FOR TIME OUT
356	19A4	10 FB		BPL	ONE1	
357	19A6	A9 7E		LDA	#126	
358	19A8	8D 44 17		STA	CLK1T	
359	19AB	A9 A7		LDA	#BA7	
360	19AD	8D 42 17		STA	SBD	SET PB7=1
361	19B0	2C 47 17	ONE2	BIT	CLKRDI	
362	19B3	10 FB		BPL	ONE2	
363	19B5	A9 7E		LDA	#126	
364	19B7	8D 44 17		STA	CLK1T	
365	19BA	A9 27		LDA	#B27	
366	19BC	8D 42 17		STA	SBD	RESET PB7=0
367	19BF	CA		DEX		
368	19C0	D0 DF		BNE	ONE1	
369	19C2	68		PLA		
370	19C3	60		RTS		
371						
372						
373						
374						
375						
376	19C4	A2 06	ZRD	LDX	#B06	
377	19C6	48		PHA		SAVX A
378	19C7	2C 47 17	ZRD1	BIT	CLKRDI	
379	19CA	10 FB		BPL	ZRD1	

CARD #	LOC	CODE	CARD		
380	190C	A9 C3	LDA	#195	
381	190E	8D 44 17	STA	CLK1T	
382	19D1	A9 A7	LDA	#1A7	
383	19D3	8D 42 17	STA	SBD	SET PB7=1
384	19D6	2C 47 17	ZR02	BIT	CLKRDI
385	19D9	10 FB	BPL	ZR02	
386	19DB	A9 C3	LDA	#195	
387	19DD	8D 44 17	STA	CLK1T	
388	19E0	A9 27	LDA	#127	
389	19E2	8D 42 17	STA	SBD	RESET PB7=0
390	19E5	CA	DEX		
391	19E6	D0 DF	BNE	ZR01	
392	19E8	68	PLA		RESTORE A
393	19E9	60	RTS		
394					
395				SUB TO INC VEB+1,2	
396					
397	19EA	EE ED 17	INCVEB	INC	VEB+1
398	19ED	D0 03	BNE	INCVE1	
399	19EF	EE EE 17	INC	VEB+2	
400	19F2	60	INCVE1	RTS	
401					
402				SUB TO READ BYTE FROM TAPE	
403					
404	19F3	20 24 1A	RDBYT	JSR	RDCHT
405	19F6	20 00 1A		JSR	PACKT
406	19F9	20 24 1A	RDBYT2	JSR	RDCHT
407	19FC	20 00 1A		JSR	PACKT
408	19FF	60		RTS	
409					
410				PACK A=ASCII INTO SAVX	
411				AS HEX DATA	
412					
413	1A00	C9 30	PACKT	CMP	#130
414	1A02	30 1E		BMI	PACKT3
415	1A04	C9 47		CMP	#147
416	1A06	10 1A		BPL	PACKT3
417	1A08	C9 40		CMP	#140
418	1A0A	30 03		BMI	PACKT1
419	1A0C	18		CLC	
420	1A0D	69 09		ADC	#109
421	1A0F	2A	PACKT1	ROL	A
422	1A10	2A		ROL	A
423	1A11	2A		ROL	A
424	1A12	2A		ROL	A
425	1A13	A0 04		LDY	#104
426	1A15	2A	PACKT2	ROL	A
427	1A16	2E E9 17		ROL	SAVX
428	1A19	88		DEY	
429	1A1A	D0 F9		BNE	PACKT2
430	1A1C	A0 E9 17		LDA	SAVX
431	1A1F	A0 00		LDY	#100

Y=0 VALID HEX CHAR

CARD #	LOC	CODE	CARD		
432	1A21	60	RTS		Y=0 VALID HEX
433	1A22	C8	PACKT3 INY		Y=1 NOT HEX
434	1A23	60	RTS		
435			;		
436			;	GET 1 CHAR FROM TAPE AND RETURN	
437			;	WITH CHAR IN A USE SAVX+1 TO ASM CHAR	
438			;		
439	1A24	3E EB 17	RDCHT STX	SAVX+2	
440	1A27	A2 08	LDX	#B08	READ 8 BITS
441	1A29	20 41 1A	RDCHT1 JSR	RDBIT	GET NEXT DATA BIT
442	1A2C	4E EA 17	LSR	SAVX+1	RIGHT SHIFT CHAR
443	1A2F	00 EA 17	ORA	SAVX+1	OR IN SIGN BIT
444	1A32	8D EA 17	STA	SAVX+1	REPLACE CHAR
445	1A35	CA	DEX		
446	1A36	D0 F1	BNE	RDCHT1	
447			;		
448	1A38	AD EA 17	LDA	SAVX+1	MOVE CHAR INTO A
449	1A3B	2A	ROL	A	SHIFT OFF PARITY
450	1A3C	4A	LSR	A	
451	1A3D	AE EB 17	LDX	SAVX+2	
452	1A40	60	RTS		
453			;		
454			;	THIS SUB GETS ONE BIT FROM	
455			;	TAPE AND RETURNS IT IN SIGN OF A	
456			;		
457	1A41	2C 42 17	RDBIT BIT	SBD	WAIT FOR END OF START BIT
458	1A44	10 FB	BPL	RDBIT	
459	1A46	AD 46 17	LDA	CLKRDT	GET START BIT TIME
460	1A49	A0 FF	LDY	#BFF	A=256-T1
461	1A4B	8C 46 17	STY	CLK64T	SET UP TIMER
462			;		
463	1A4E	A0 14	LDY	#B14	
464	1A50	38	RDBIT3 DEY		DELAY 100 MICROSEC
465	1A51	D0 FD	BNE	RDBIT3	
466			;		
467	1A53	2C 42 17	RDBIT2 BIT	SBD	
468	1A56	30 FB	BMI	RDBIT2	WAIT FOR NEXT START BIT
469			;		
470	1A58	38	SEC		
471	1A59	ED 46 17	SBC	CLKRDT	(256-T1)-(256-T2)=T2-T1
472	1A5C	A0 FF	LDY	#BFF	
473	1A5E	8C 46 17	STY	CLK64T	SET UP TIMER FOR NEXT BIT
474			;		
475	1A61	A0 07	LDY	#B07	
476	1A63	88	RDBIT4 DEY		DELAY 50 MICROSEC
477	1A64	D0 FD	BNE	RDBIT4	
478			;		
479	1A66	49 FF	EOR	#BFF	COMPLEMENT SIGN OF A
480	1A68	29 80	AND	#B80	MASK ALL EXCEPT SIGN
481	1A6A	60	RTS		

CARD #	LOC	CODE	CARD
483			;
484			;
485			DIAGNOSTICS
486			MEMORY
487			PLLCAL
488			;
489			;
490			PLLCAL OUTPUT 166 MICROSEC
491			PULSE STRING
492			;
493	1A6B	A9 27	PLLCAL LDA #B27
494	1A6D	8D 42 17	STA SBD
495	1A70	A9 BF	LDA #BBF
496	1A72	8D 43 17	STA PBDD
497			;
498	1A75	2C 47 17	PLL1 BIT CLKRDI
499	1A78	10 FB	BPL PLL1
500	1A7A	A9 9A	LDA #154
501	1A7C	8D 44 17	STA CLK1T
502	1A7F	A9 A7	LDA #BA7
503	1A81	8D 42 17	STA SBD
504			;
505	1A84	2C 47 17	PLL2 BIT CLKRDI
506	1A87	10 FB	BPL PLL2
507	1A89	A9 9A	LDA #154
508	1A8B	8D 44 17	STA CLK1T
509	1A8E	A9 27	LDA #B27
510	1A90	8D 42 17	STA SBD
511	1A93	4C 75 1A	JMP PLL1
512			;
513			;
514			;
515			INTERRUPTS PAGE 27
516	1A96		***\$0164 RESERVED FOR TEST
517	1BFA	6B 1A	NMIP27 .WORD PLLCAL
518	1BFC	6B 1A	RSTP27 .WORD PLLCAL
519	1BFE	6B 1A	IRQP27 .WORD PLLCAL
520			;

TURN OFF DATIN PBS=1
 CONVERT PB7 TO OUTPUT

WAIT 166 MICRO SEC

OUTPUT PB7=1

PB7=0

CARD # LOC	CODE	CARD				
522	:					
523	:					
524	:					
525	:					
526	:	666666	555555	333333	000000	
527	:	6	5	3	0	0
528	:	6	5	3	0	0
529	:	666666	555555	333333	0	0
530	:	6 6	5	3	0	0
531	:	6 6	5	3	0	0
532	:	666666	555555	333333	000000	
533	:					
534	:					
535	:					
536	:		000000	000000	222222	
537	:		0 0	0 0		2
538	:	-----	0 0	0 0		2
539	:	-----	0 0	0 0	222222	
540	:	-----	0 0	0 0	2	
541	:		0 0	0 0	2	
542	:		000000	000000	222222	
543	:					

CARD #	LOC	CODE	CARD
545			
546			
547			
548			
549			COPYRIGHT
550			MOS TECHNOLOGY INC.
551			DATE OCT 13 1975 REV E
552			
553			KIM :TTY INTERFACE
554			:KEYBOARD INTERFACE
555			:7 SE5 6 DIGIT DISPLAY
556			
557			TTY CMDS:
558			G GOEXEC
559			OR OPEN NEXT CELL
560			LF OPEN PREV. CELL
561			. MODIFY OPEN CELL
562			SP OPEN NEW CELL
563			L LOAD (OBJECT FORMAT)
564			Q DUMP FROM OPEN CELL ADDR TO HI LIMIT
565			RO RUB OUT - RETURN TO START (KIM)
566			((ALL ILLEGAL CHAR ARE IGNORED))
567			
568			KEYBOARD CMDS:
569			ADDR SETS MODE TO MODIFY CELL ADDRESS
570			DATA SETS MODE TO MODIFY DATA IN OPEN CELL
571			STEP INCREMENTS TO NEXT CELL
572			RST SYSTEM RESET
573			RUN GOEXEC
574			STOP \$1000 CAN BE LOADED INTO NMIV TO
575			USE STOP FEATURE
576			PC DISPLAY PC
577			
578			CLOCK IS NOT DISABLED IN SIGMA 1
579			
580			
581			
582			

CARD #	LOC	CODE	CARD		
584	1000			◆=\$1000	
585				;	
586				;	
587	1000	85 F3	SAVE	STA	ACC KIM ENTRY VIA STOP (NMI)
588	1002	68		PLA	OR BRK (IRQ)
589	1003	85 F1		STA	PREG
590	1005	68	SAVE1	PLA	KIM ENTRY VIA JSR (A LOST)
591	1006	85 EF		STA	PCL
592	1008	85 FA		STA	POINTL
593	100A	68		PLA	
594	100B	85 F0		STA	PCH
595	100D	85 FB		STA	POINTH
596	100F	84 F4	SAVE2	STY	YREG
597	1011	86 F5		STX	XREG
598	1013	BA		TSX	
599	1014	86 F2		STX	SPUSER
600	1016	20 88 1E		JSR	INITS
601	1019	40 4F 1C		JMP	START
602				;	
603	1010	60 FA 17	NMIT	JMP	(NMIV) NON-MASKABLE INTERRUPT TRAP
604	101F	60 FE 17	IRQT	JMP	(IRQV) INTERRUPT TRAP
605				;	
606	1022	A2 FF	RST	LDX	#\$FF KIM ENTRY VIA RST
607	1024	9A		TXS	
608	1025	86 F2		STX	SPUSER
609	1027	20 88 1E		JSR	INITS
610				;	
611				;	
612	102A	A9 FF	DETCPS	LDA	#\$FF COUNT START BIT
613	102C	8D F3 17		STA	CNTH30 ZERO CNTH30
614	102F	A9 01		LDA	#\$01 MASK HI ORDER BITS
615	1031	2C 40 17	DET1	BIT	SAD TEST
616	1034	D0 19		BNE	START KEYBD SSM TEST
617	1036	30 F9		BMI	DET1 START BIT TEST
618	1038	A9 FC		LDA	#\$FC
619	103A	18	DET3	CLC	THIS LOOP COUNTS
620	103B	69 01		ADC	#\$01 THE START BIT TIME
621	103D	90 03		BCC	DET2
622	103F	EE F3 17		INC	CNTH30
623	1042	AC 40 17	DET2	LDY	SAD CHECK FOR END OF START BIT
624	1045	10 F3		BPL	DET3
625	1047	8D F2 17		STA	CNTH30
626	104A	A2 09		LDX	#\$08
627	104C	20 6A 1E		JSR	GET5 GET REST OF THE CHAR
628				;	TEST CHAR HERE
629				;	
630				;	
631				;	
632				;	
633				;	
634				;	MAKE TTY/KB SELECTION
635				;	

CARD #	LOC	CODE	CARD		
636	104F	20 8C 1E	START	JSR	INIT1
637	1052	A9 01		LDA	#\$01
638	1054	2C 40 17		BIT	SAD
639	1057	D0 1E		BNE	TTYKB
640	1059	20 2F 1E		JSR	CRLF
641	105C	A2 0A		LDX	#\$0A
642	105E	20 31 1E		JSR	PRIST
643	1061	4C AF 1D		JMP	SHOW1
644					
645	1064	A9 00	CLEAR	LDA	#\$00
646	1066	85 F8		STA	INH
647	1068	85 F9		STA	INH
648	106A	20 5A 1E	READ	JSR	GETCH
649	106D	C9 01		CMR	#\$01
650	106F	F0 06		BEQ	TTYKB
651	1071	20 AC 1F		JSR	PACK
652	1074	4C DB 1D		JMP	SCAN
653					
654					
655					
656					
657	1077	20 19 1F	TTYKB	JSR	SCAND
658	107A	D0 D3		BNE	START
659	107C	A9 01	TTYKB1	LDA	#\$01
660	107E	2C 40 17		BIT	SAD
661	1081	F0 0C		BEQ	START
662	1083	20 19 1F		JSR	SCAND
663	1086	F0 F4		BEQ	TTYKB1
664	1088	20 19 1F		JSR	SCAND
665	108B	F0 EF		BEQ	TTYKB1
666					
667	108D	20 6A 1F	GETK	JSR	GETKEY
668	1090	C9 15		CMR	#\$15
669	1092	10 B8		BPL	START
670	1094	C9 14		CMR	#\$14
671	1096	F0 44		LDQ	PCOMD
672	1098	C9 10		CMR	#\$10
673	109A	F0 2C		BEQ	ADDRM
674	109C	C9 11		CMR	#\$11
675	109E	F0 2C		BEQ	DATAM
676	10A0	C9 12		CMR	#\$12
677	10A2	F0 3F		BEQ	STEP
678	10A4	C9 13		CMR	#\$13
679	10A6	F0 31		BEQ	GOV
680	10A8	0A	DATA	ASL	A
681	10A9	0A		ASL	A
682	10AA	0A		ASL	A
683	10AB	0A		ASL	A
684	10AC	85 FC		STA	TEMP
685	10AE	A2 04		LDX	#\$04
686	10B0	A4 FF	DATA1	LDY	MODE
687	10B2	D0 0A		BNE	ADDR

PRT CR LF
TYPE OUT KIM

CLEAR INPUT BUFFER

GET CHAR

MAIN ROUTINE FOR KEY BOARD
AND DISPLAY

IF A=0 NO KEY

DISPLAY PC
ADDR MODE=1

DATA MODE=1

STEP

RUN

SHIFT CHAR INTO HIGH
ORDER NIBBLE

STORE IN TEMP

TEST MODE 1=ADDR
MODE=0 DATA

CARD #	LOC	CODE	CARD			
688	1CB4	B1 FA	LDA	(POINTL),Y	GET DATA	
689	1CB6	06 FC	ASL	TEMP	SHIFT CHAR	
690	1CB8	2A	ROL	A	SHIFT DATA	
691	1CB9	91 FA	STA	(POINTL),Y	STORE OUT DATA	
692	1CBB	4C C3 1C	JMP	DATA2		
693						
694	1CBE	0A	ADDR	ASL	A	SHIFT CHAR
695	1CBF	26 FA		ROL	POINTL	SHIFT ADDR
696	1CC1	26 FB		ROL	POINTH	SHIFT ADDR HI
697	1CC3	CA	DATA2	DEX		
698	1CC4	D0 EA		BNE	DATA1	DO 4 TIMES
699	1CC6	F0 08		BEQ	DATAM2	EXIT HERE
700						
701	1CC8	A9 01	ADDRM	LDA	#\$01	
702	1CCA	D0 02		BNE	DATAM1	
703						
704	1CCC	A9 00	DATAM	LDA	#\$00	
705	1CCE	85 FF	DATAM1	STA	MODE	
706	1CD0	4C 4F 1C	DATAM2	JMP	START	
707						
708	1CD3	20 63 1F	STEP	JSR	INCPT	
709	1CD6	4C 4F 1C		JMP	START	
710						
711	1CD9	4C C8 1D	GOV	JMP	GOEXEC	
712						
713						
714					DISPLAY PC BY MOVING	
715					PC TO POINT	
716						
717	1CDC	A5 EF	PCCMD	LDA	PCL	
718	1CDE	85 FA		STA	POINTL	
719	1CE0	A5 F0		LDA	PCH	
720	1CE2	85 FB		STA	POINTH	
721	1CE4	4C 4F 1C		JMP	START	
722						
723					LOAD PAPER TAPE FROM TTY	
724						
725	1CE7	20 5A 1E	LOAD	JSR	GETCH	LOOK FOR FIRST CHAR
726	1CEA	C9 3B		CMP	#\$3B	SMICOLON
727	1CEC	D0 F9		BNE	LOAD	
728	1CEE	A9 00	LOADS	LDA	#\$00	
729	1CF0	85 F7		STA	CHKSUM	
730	1CF2	85 F6		STA	CHKHI	
731						
732	1CF4	20 9D 1F		JSR	GETBYT	GET BYTE CNT
733	1CF7	AA		TAX		SAVE IN X INDEX
734	1CF8	20 91 1F		JSR	CHK	COMPUTE CHKSUM
735						
736	1CFB	20 9D 1F		JSR	GETBYT	GET ADDRESS HI
737	1CFE	35 FB		STA	POINTH	
738	1D00	20 91 1F		JSR	CHK	
739	1D03	20 9D 1F		JSR	GETBYT	GET ADDRESS LO

CARD #	LDC	CODE	CARD			
740	1D06	95 FA		STA	POINTL	
741	1D08	20 91 1F		JSR	CHK	
742						
743	1D0B	9A		TXA		IF CNT=0 DONT
744	1D0C	F0 0F		BEQ	LOAD3	GET ANY DATA
745						
746	1D0E	20 9D 1F	LOAD2	JSR	GETBYT	GET DATA
747	1D11	91 FA		STA	(POINTL),Y	STORE DATA
748	1D13	20 91 1F		JSR	CHK	
749	1D16	20 63 1F		JSR	INCPY	NEXT ADDRESS
750	1D19	CA		DEX		
751	1D1A	D0 F2		BNE	LOAD2	
752	1D1C	68		INX		X=1 DATA RECORD
753						X=0 LAST RECORD
754	1D1D	20 9D 1F	LOAD3	JSR	GETBYT	COMPARE CHKSUM
755	1D26	C5 F6		CMP	CHKHI	
756	1D22	D0 17		BNE	LOADE1	
757	1D24	20 9D 1F		JSR	GETBYT	
758	1D27	C5 F7		CMP	CHKSUM	
759	1D29	D0 13		BNE	LOADER	
760						
761	1D2B	9A		TXA		X=0 LAST RECORD
762	1D2C	D0 B9		BNE	LOAD	
763						
764	1D2E	A2 0C	LOAD7	LDX	#\$0C	X-OFF KIM
765	1D30	A9 27	LOAD8	LDA	#\$27	
766	1D32	8D 42 17		STA	SBD	DISABLE DATA IN
767	1D35	20 31 1E		JSR	PRST	
768	1D38	4C 4F 1C		JMP	START	
769						
770	1D3B	20 9D 1F	LOADE1	JSR	GETBYT	DUMMY
771	1D3E	A2 11	LOADER	LDX	#\$11	X-OFF ERR KIM
772	1D40	D0 EE		BNE	LOAD8	
773						
774						
775						
776						
777						
778	1D42	A9 00	DUMP	LDA	#\$00	
779	1D44	85 F8		STA	INL	
780	1D46	85 F9		STA	INH	CLEAR RECORD COUNT
781	1D48	A9 00	DUMP0	LDA	#\$00	
782	1D4A	85 F6		STA	CHKHI	CLEAR CHKSUM
783	1D4C	85 F7		STA	CHKSUM	
784						
785	1D4E	20 2F 1E	DUMP1	JSR	CRLF	PRINT CR LF
786	1D51	A9 3B		LDA	#\$3B	PRINT SMICOLON
787	1D53	20 A0 1E		JSR	OUTCH	
788	1D56	A5 FA		LDA	POINTL	TEST POINT GT OR ET
789	1D59	CD F7 17		CMP	EAL	HI LIMIT GO TO EXIT
790	1D5B	A5 FB		LDA	POINTH	
791	1D5D	ED F8 17		SBC	EAH	

CARD #	LOC	CODE	CARD			
792	1D60	90 18	BCC	DUMP4		
793						
794	1D62	A9 00	LDA	#000	PRINT LAST RECORD	
795	1D64	20 3B 1E	JSR	PRTBYT	0 BYTES	
796	1D67	20 0C 1F	JSR	OPEN		
797	1D6A	20 1E 1E	JSR	PRTPNT		
798						
799	1D6D	A5 F6	LDA	CHKHI	PRINT CHKSUM	
800	1D6F	20 3B 1E	JSR	PRTBYT	FOR LAST RECORD	
801	1D72	A5 F7	LDA	CHKSUM		
802	1D74	20 3B 1E	JSR	PRTBYT		
803	1D77	4C 64 1C	JMP	CLEAR		
804						
805	1D7A	A9 18	DUMP4	LDA	#018	PRINT 24 BYTE CNT
806	1D7C	AA	TAX		SAVE AS INDEX	
807	1D7D	20 3B 1E	JSR	PRTBYT		
808	1D80	20 91 1F	JSR	CHK		
809	1D83	20 1E 1E	JSR	PRTPNT		
810						
811	1D86	A0 00	DUMP2	LDY	#000	PRINT 24 BYTES
812	1D88	B1 FA	LDA	(POINTL),Y	GET DATA	
813	1D8A	20 3B 1E	JSR	PRTBYT	PRINT DATA	
814	1D8D	20 91 1F	JSR	CHK	COMP CHKSUM	
815	1D90	20 63 1F	JSR	INCPT	INCREMENT POINT	
816	1D93	CA	DEX			
817	1D94	D0 F0	BNE	DUMP2		
818						
819	1D96	A5 F6	LDA	CHKHI	PRINT CHKSUM	
820	1D98	20 3B 1E	JSR	PRTBYT		
821	1D9B	A5 F7	LDA	CHKSUM		
822	1D9D	20 3B 1E	JSR	PRTBYT		
823	1DA0	E6 F8	INC	INL	INCREMENT RECORD CNT	
824	1DA2	D0 02	BNE	DUMP3		
825	1DA4	E6 F9	INC	INH		
826	1DA6	4C 48 1D	DUMP3	JMP	DUMP0	
827						
828	1DA9	20 0C 1F	SPACE	JSR	OPEN	OPEN NEW CELL
829	1DAC	20 2F 1E	SHOW	JSR	CRLF	PRINT CR LF
830	1DAF	20 1E 1E	SHOW1	JSR	PRTPNT	
831	1DB2	20 9E 1E	JSR	OUTSP	PRT SPACE	
832	1DB5	A0 00	LDY	#000	PRINT DATA SPECIFIED	
833	1DB7	B1 FA	LDA	(POINTL),Y	BY POINT AD = LDA EXT	
834	1DB9	20 3B 1E	JSR	PRTBYT		
835	1DBC	20 9E 1E	JSR	OUTSP	PRT SPACE	
836	1DBF	4C 64 1C	JMP	CLEAR		
837						
838	1DC2	20 63 1F	RTRN	JSR	INCPT	OPEN NEXT CELL
839	1DC5	4C AC 1D		JMP	SHOW	
840						
841	1DC8	A6 F2	GOEXEC	LDX	SPUSER	
842	1DCA	9A	TXS			
843	1DCB	A5 FB	LDA	POINTH	PROGRAM RUNS FROM	

CARD #	LOC	CODE	CARD		
844	1DCD	48	PHA		OPEN CELL ADDRESS
845	1DCE	A5 FA	LDA	POINTL	
846	1DD0	48	PHA		
847	1DD1	A5 F1	LDA	PREG	
848	1DD3	48	PHA		
849	1DD4	A6 F5	LDX	XREG	RESTORE REGS
850	1DD6	A4 F4	LDY	YREG	
851	1DD8	A5 F3	LDA	ACC	
852	1DDA	40	RTI		
853			;		
854	1DD8	C9 20	SCAN	CMP	#\$20 OPEN CELL
855	1DD0	F0 0A		BEQ	SPACE
856	1DDF	C9 7F		CMP	#\$7F RUB OUT (KIM)
857	1DE1	F0 18		BEQ	STV
858	1DE3	C9 00		CMP	#\$00 NEXT CELL
859	1DE5	F0 D8		BEQ	RTRN
860	1DE7	C9 0A		CMP	#\$0A PREV CELL
861	1DE9	F0 1C		BEQ	FEED
862	1DEB	C9 2E		CMP	#\$1. MODIFY CELL
863	1DED	F0 26		BEQ	MODIFY
864	1DEF	C9 47		CMP	#\$16 GO EXEC
865	1DF1	F0 D5		BEQ	GOEXEC
866	1DF3	C9 51		CMP	#\$10 DUMP FROM OPEN CELL TO HI LIMIT
867	1DF5	F0 0A		BEQ	DUMPV
868	1DF7	C9 4C		CMP	#\$1L LOAD TAPE
869	1DF9	F0 03		BEQ	LOADV
870	1DFB	4C 6A 1C		JMP	READ IGNORE ILLEGAL CHAR
871			;		
872	1DFE	4C 4F 1C	STV	JMP	START
873	1E01	4C 42 1D	DUMPV	JMP	DUMP
874	1E04	4C E7 1C	LOADV	JMP	LOAD
875			;		
876	1E07	38	FEED	SEC	
877	1E08	A5 FA		LDA	POINTL DEC DOUBLE BYTE
878	1E0A	E9 01		SBC	#\$01 AT POINTL AND POINTH
879	1E0C	85 FA		STA	POINTL
880	1E0E	B0 02		BCS	FEED1
881	1E10	C6 FB		DEC	POINTH
882	1E12	4C AC 1D	FEED1	JMP	SHOW
883			;		
884	1E15	A0 00	MODIFY	LDY	#\$00 GET CONTENTS OF INPUT BUFF
885	1E17	A5 F8		LDA	INL INL AND STOR IN LOC
886	1E19	91 FA		STA	(POINTL),Y SPECIFIED BY POINT
887	1E1B	4C C2 1D		JMP	RTRN
888			;		
889			;		END OF MAIN LINE

```

CARD # LOC      CODE      CARD
891          ;          SUBROUTINES FOLLOW
892          ;
893          ;
894          ;
895          ;          SUB TO PRINT POINTL,POINTH
896          ;
897 1E1E A5 FB      PRTPNT LDA    POINTH
898 1E20 20 3B 1E      JSR    PRTTYT
899 1E23 20 91 1F      JSR    CHK
900 1E26 A5 FA      LDA    POINTL
901 1E28 20 3B 1E      JSR    PRTTYT
902 1E2B 20 91 1F      JSR    CHK
903 1E2E 60          RTS
904          ;
905          ;          PRINT STRING OF ASCII CHAR FROM
906          ;          TOP+X TO TOP
907          ;
908 1E2F A2 07      ORLF   LDX    #$07
909 1E31 BD D5 1F      PRST   LDA    TOP,X
910 1E34 20 A0 1E      JSR    OUTCH
911 1E37 0A          DEX
912 1E38 10 F7      BPL    PRST      STOP ON INDEX ZERO
913 1E3A 60          PR1    RTS
914          ;
915          ;          PRINT 1 HEX BYTE AS TWO ASCII CHAR'S
916          ;
917 1E3B 85 FC      PRTTYT STA    TEMP
918 1E3D 4A          LSR     A          SHIFT CHAR RIGHT 4 BITS
919 1E3E 4A          LSR     A
920 1E3F 4A          LSR     A
921 1E40 4A          LSR     A
922 1E41 20 4C 1E      JSR    HEXTA      CONVERT TO HEX AND PRINT
923 1E44 A5 FC      LDA    TEMP      GET OTHER HALF
924 1E46 20 4C 1E      JSR    HEXTA      CONVERT TO HEX AND PRINT
925 1E49 A5 FC      LDA    TEMP      RESTORE BYTE IN A AND RETURN
926 1E4B 60          RTS
927          ;
928 1E4C 29 0F      HEXTA  AND    #$0F      MASK HI 4 BITS
929 1E4E 09 0A      CMP    #$0A
930 1E50 18          CLO
931 1E51 30 02      BMI    HEXTA1
932 1E53 69 07      ADC    #$07      ALPHA HEX
933 1E55 69 30      HEXTA1 ADC    #$30      DEC HEX
934 1E57 4C A0 1E      JMP    OUTCH      PRINT CHAR
935          ;
936          ;          SET 1 CHAR FROM TTY
937          ;          RETURN FROM SUB WITH CHAR IN A
938          ;          X IS PRESERVED AND Y RETURNED = FF
939          ;
940 1E5A 86 FD      GETCH  STX    TMPX      SAVE X REG
941 1E5C A2 08      LDX    #$08      SET UP 8 BIT CNT
942 1E5E A9 01      LDA    #$01

```

CARD #	LOC	CODE	CARD			
943	1E60	20 40 17	GET1	BIT	SAD	
944	1E63	D0 22		BNE	GET6	
945	1E65	30 F9		BMI	GET1	WAIT FOR START BIT
946	1E67	20 D4 1E		JSR	DELAY	DELAY 1 BIT
947	1E6A	20 EB 1E	GET5	JSR	DEHALF	DELAY 1/2 BIT TIME
948	1E6D	AD 40 17	GET2	LDA	SAD	GET 8 BITS
949	1E70	29 80		AND	#\$80	MASK OFF LOW ORDER BITS
950	1E72	46 FE		LSR	CHAR	SHIFT RIGHT CHARACTER
951	1E74	05 FE		DRA	CHAR	
952	1E76	85 FE		STA	CHAR	
953	1E78	20 D4 1E		JSR	DELAY	DELAY 1 BIT TIME
954	1E7B	0A		DEX		
955	1E7C	D0 EF		BNE	GET2	GET NEXT CHAR
956	1E7E	20 EB 1E		JSR	DEHALF	EXIT THIS RTN
957						
958	1E81	A6 FD		LDX	TMPX	
959	1E83	A5 FE		LDA	CHAR	
960	1E85	2A		ROL	A	SHIFT OFF PARITY
961	1E86	4A		LSR	A	
962	1E87	60	GET6	RTS		
963						
964						
965						
966	1E88	A2 01	INIT5	LDX	#\$01	SET KB MODE TO ADDR
967	1E8A	86 FF		STX	MODE	
968						
969	1E8C	A2 00	INIT1	LDX	#\$00	
970	1E8E	8E 41 17		STX	PADD	FOR SIGMA USE SADD
971	1E91	A2 3F		LDX	#\$3F	
972	1E93	8E 43 17		STX	PBDD	FOR SIGMA USE SBDD
973	1E96	A2 07		LDX	#\$07	ENABLE DATA IN
974	1E98	8E 42 17		STX	SBD	OUTPUT
975	1E9B	D8		CLD		
976	1E9C	78		SEI		
977	1E9D	60		RTS		
978						
979						
980						
981						
982						
983	1E9E	A9 20	OUTSP	LDA	#\$20	
984	1EA0	85 FE	OUTCH	STA	CHAR	
985	1EA2	86 FD		STX	TMPX	
986	1EA4	20 D4 1E		JSR	DELAY	10/11 BIT CODE SYNC
987	1EA7	AD 42 17		LDA	SBD	START BIT
988	1EAA	29 FE		AND	#\$FE	
989	1EAC	9D 42 17		STA	SBD	
990	1EAF	20 D4 1E		JSR	DELAY	
991	1EB2	A2 08		LDX	#\$08	
992	1EB4	AD 42 17	OUT1	LDA	SBD	DATA BIT
993	1EB7	29 FE		AND	#\$FE	
994	1EB9	46 FE		LSR	CHAR	

CARD #	LOC	CODE	CARD		
995	1EBB	69 00	ADC	#B00	
996	1EBD	8D 42 17	STA	SBD	
997	1EC0	20 D4 1E	JSR	DELAY	
998	1EC3	CA	DEX		
999	1EC4	D0 EE	BNE	OUT1	
1000	1EC6	AD 42 17	LDA	SBD	STOP BIT
1001	1EC9	09 01	ORA	#B01	
1002	1ECB	8D 42 17	STA	SBD	
1003	1ECE	20 D4 1E	JSR	DELAY	STOP BIT
1004	1ED1	A6 FD	LDX	TMPX	RESTORE INDEX
1005	1ED3	60	RTS		
1006			:		
1007			:	DELAY 1 BIT TIME	
1008			:	AS DETERMEND BY DETCPS	
1009			:		
1010	1ED4	AD F3 17	DELAY LDA	CNTH30	THIS LOOP SIMULATES THE
1011	1ED7	8D F4 17	STA	TIMH	DETCPS SECTION AND WILL DELAY
1012	1EDA	AD F2 17	LDA	CNTL30	1 BIT TIME
1013	1EDD	38	DE2 SEC		
1014	1EDE	E9 01	DE4 SBC	#B01	
1015	1EE0	B0 03	BCS	DE3	
1016	1EE2	CE F4 17	DEC	TIMH	
1017	1EE5	AC F4 17	DE3 LDY	TIMH	
1018	1EE8	10 F3	BPL	DE2	
1019	1EEA	60	RTS		
1020			:		
1021			:	DELAY HALF BIT TIME	
1022	1EEB	AD F3 17	DEHALF LDA	CNTH30	DOUBLE RIGHT SHIFT OF DELAY
1023	1EEE	8D F4 17	STA	TIMH	CONSTANT FOR A DIV BY 2
1024	1EF1	AD F2 17	LDA	CNTL30	
1025	1EF4	4A	LSR	A	
1026	1EF5	4E F4 17	LSR	TIMH	
1027	1EF8	90 E3	BCC	DE2	
1028	1EFA	09 90	ORA	#B80	
1029	1EFC	B0 E0	BCS	DE4	
1030			:		
1031			:	SUB TO DETERMINE IF KEY IS	
1032			:	DEPRESSED OR COMDITON OF SSW	
1033			:	KEY NOT DEP OR TTY MODE A = 0	
1034			:	KEY DEP OR KB MODE A NOT ZERO	
1035			:		
1036			:		
1037	1EFE	A0 03	AK LDY	#B03	3 ROWS
1038	1F00	A2 01	LDX	#B01	DIGIT 0
1039			:		
1040	1F02	A9 FF	ONEKEY LDA	#BFF	
1041	1F04	8E 42 17	AK1 STX	SBD	OUTPUT DIGIT
1042	1F07	E8	INX		GET NXT DIGIT
1043	1F08	E8	INX		
1044	1F09	2D 40 17	AND	SAD	INPUT SEGMENTS
1045	1F0C	88	DEY		
1046	1F0D	D0 F5	BNE	AK1	

CARD #	LOC	CODE	CARD		
1047					
1048	1F0F	A0 07		LDY	#\$07
1049	1F11	8C 42 17		STY	SBD
1050					
1051	1F14	09 80		DRA	#\$80
1052	1F16	49 FF		EDR	#\$FF
1053	1F18	60		RTS	
1054					
1055				SUB	OUTPUT TO 7 SEGMENT DISPLAY
1056					
1057	1F19	A0 00	SCAND	LDY	#\$00 GET DATA SPECIFIED
1058	1F1B	B1 FA		LDA	(POINTL),Y BY POINT
1059	1F1D	85 F9		STA	INH SET UP DISPLAY BUFFER
1060	1F1F	A9 7F	SCANDS	LDA	#\$7F CHANGE SEG
1061	1F21	8D 41 17		STA	PADD TO OUTPUT
1062					
1063	1F24	A2 09		LDX	#\$09 INIT DIGIT NUMBER
1064	1F26	A0 03		LDY	#\$03 OUTPUT 3 BYTES
1065					
1066	1F28	B9 F8 00	SCAND1	LDA	INL,Y GET BYTE
1067	1F2B	4A		LSR	A GET MSD
1068	1F2C	4A		LSR	A
1069	1F2D	4A		LSR	A
1070	1F2E	4A		LSR	A
1071	1F2F	20 48 1F		JSR	CONVD OUTPUT CHAR
1072	1F32	B9 F8 00		LDA	INL,Y GET BYTE AGAIN
1073	1F35	29 0F		AND	#\$0F GET LSD
1074	1F37	20 48 1F		JSR	CONVD OUTPUT CHAR
1075	1F3A	38		DEY	SET UP FOR NXT BYTE
1076	1F3B	D0 EB		BNE	SCAND1
1077	1F3D	8E 42 17		STX	SBD ALL DIGITS OFF
1078	1F40	A9 00		LDA	#\$00 CHANGE SEG
1079	1F42	8D 41 17		STA	PADD TO INPUTS
1080	1F45	4C FE 1E		JMP	AK GET ANY KEY
1081					
1082					
1083					CONVERT AND DISPLAY HEX
1084					USED BY SCAND ONLY
1085	1F48	84 FC	CONVD	STY	TEMP SAVE Y
1086	1F4A	A8		TAY	USE CHAR AS INDEX
1087	1F4B	B9 E7 1F		LDA	TABLE,Y LOOK UP CONVERSION
1088	1F4E	A0 00		LDY	#\$00 TURN OFF SEGMENTS
1089	1F50	3C 40 17		STY	SAD
1090	1F53	8E 42 17		STX	SBD OUTPUT DIGIT ENABLE
1091	1F56	8D 40 17		STA	SAD OUT PUT SEGMENTS
1092					
1093	1F59	A0 7F		LDY	#\$7F DELAY 500 CYCLES APPROX.
1094	1F5B	88	CONVD1	DEY	
1095	1F5C	D0 FD		BNE	CONVD1
1096					
1097	1F5E	E8		INX	GET NEXT DIGIT NUM
1098	1F5F	E8		INX	ADD 2

CARD #	LOC	CODE	CARD			
1099	1F60	A4 FC		LDY	TEMP	RESTORE Y
1100	1F62	60		RTS		
1101						
1102						
1103						SUB TO INCREMENT POINT
1104	1F63	E6 FA	INCPT	INC	POINTL	
1105	1F65	D0 02		BNE	INCPT2	
1106	1F67	E6 FB		INC	POINTH	
1107	1F69	60	INCPT2	RTS		
1108						
1109						GET KEY FROM KEY BOARD
1110						RETURN WITH A=KEY VALUE
1111						A GT. 15 THEN ILLEGAL OR NO KEY
1112						
1113						
1114	1F6A	A2 21	GETKEY	LDX	#\$21	START AT DIGIT 0
1115	1F6C	A0 01	GETKEY5	LDY	#\$01	GET 1 ROW
1116	1F6E	20 02 1F		JSR	ONEKEY	
1117	1F71	D0 07		BNE	KEYIN	A=0 NO KEY
1118	1F73	E0 27		CPX	#\$27	TEST FOR DIGT 2
1119	1F75	D0 F5		BNE	GETKEY5	
1120	1F77	A9 15		LDA	#\$15	15=NO KEY
1121	1F79	60		RTS		
1122	1F7A	A0 FF	KEYIN	LDY	#\$FF	
1123	1F7C	0A	KEYIN1	ASL	A	SHIFT LEFT
1124	1F7D	B0 03		BCS	KEYIN2	UNTIL Y=KEY NUM
1125	1F7F	C8		INY		
1126	1F80	10 FA		BPL	KEYIN1	
1127	1F82	8A	KEYIN2	TXA		
1128	1F83	29 0F		AND	#\$0F	MASK MSD
1129	1F85	4A		LSR	A	DIV BY 2
1130	1F86	AA		TAX		
1131	1F87	98		TYA		
1132	1F88	10 03		BPL	KEYIN4	
1133	1F8A	18	KEYIN3	CLC		
1134	1F8B	69 07		ADC	#\$07	MULT (X-1) TIMES A
1135	1F8D	CA	KEYIN4	DEX		
1136	1F8E	D0 FA		BNE	KEYIN3	
1137	1F90	60		RTS		
1138						
1139						SUB TO COMPUTE CHECK SUM
1140						
1141	1F91	18	CHK	CLC		
1142	1F92	65 F7		ADC	CHKSUM	
1143	1F94	85 F7		STA	CHKSUM	
1144	1F96	A5 F6		LDA	CHKHI	
1145	1F98	69 00		ADC	#\$00	
1146	1F9A	85 F6		STA	CHKHI	
1147	1F9C	60		RTS		
1148						
1149						GET 2 HEX CHAR'S AND PACK
1150						INTO INL AND INH

CARD #	LOC	CODE	CARD
1151			; X PRESERVED Y RETURNED = 0
1152			; NON HEX CHAR WILL BE LOADED AS NEAREST HEX EQU
1153			;
1154	1F9D	20 5A 1E	GETBYT JSR GETCH
1155	1FA0	20 AC 1F	JSR PACK
1156	1FA3	20 5A 1E	JSR GETCH
1157	1FA6	20 AC 1F	JSR PACK
1158	1FA9	A5 F8	LDA INL
1159	1FAB	60	RTS
1160			;
1161			; SHIFT CHAR IN A INTO
1162			; INL AND INH
1163			;
1164	1FAC	C9 30	PACK CMP #\$30 CHECK FOR HEX
1165	1FAE	30 1B	BMI UPDAT2
1166	1FB0	C9 47	CMPI #\$47 NOT HEX EXIT
1167	1FB2	10 17	BPL UPDAT2
1168	1FB4	C9 40	HEXNUM CMP #\$40 CONVERT TO HEX
1169	1FB6	30 03	BMI UPDATE
1170	1FB8	18	HEXALP CLC
1171	1FB9	69 09	ADC #\$09
1172	1FBB	2A	UPDATE ROL A
1173	1FBC	2A	ROL A
1174	1FBD	2A	ROL A
1175	1FBE	2A	ROL A
1176	1FBF	A0 04	LDY #\$04 SHIFT INTO I/O BUFFER
1177	1FC1	2A	UPDAT1 ROL A
1178	1FC2	26 F8	ROL INL
1179	1FC4	26 F9	ROL INH
1180	1FC6	88	DEY
1181	1FC7	D0 F8	BNE UPDAT1
1182	1FC9	A9 00	LDA #\$00 A=0 IF HEX NUM
1183	1FCB	60	UPDAT2 RTS
1184			;
1185	1FCC	A5 F8	OPEN LDA INL MOVE I/O BUFFER TO POINT
1186	1FCE	85 FA	STA POINTL
1187	1FD0	A5 F9	LDA INH TRANSFER INH- POINTH
1188	1FD2	85 FB	STA POINTH
1189	1FD4	60	RTS
1190			;
1191			;
1192			; END OF SUBROUTINES

```

CARD # LOC      CODE      CARD
1194      ;
1195      ;      TABLES
1196      ;
1197 1FD5 00      TOP      .BYTE $00,$00,$00,$00,$00,$00,$0A,$0D,'MIK'
1197 1FD6 00
1197 1FD7 00
1197 1FD8 00
1197 1FD9 00
1197 1FDA 00
1197 1FDB 0A
1197 1FDC 0D
1197 1FDD 4D 49 4B
1198 1FE0 20      .BYTE /',,$13,'RRE',/ ',,$13
1198 1FE1 13
1198 1FE2 52 52 45
1198 1FE5 20
1198 1FE6 13
1199      ;
1200      ;      TABLE HEX TO 7 SEGMENT
1201      ;      0 1 2 3 4 5 6 7
1202 1FE7 BF      TABLE .BYTE $BF,$86,$DB,$CF,$E6,$ED,$FD,$87
1202 1FE8 86
1202 1FE9 DB
1202 1FEA CF
1202 1FEB E6
1202 1FEC ED
1202 1FED FD
1202 1FEE 87
1203      ;
1204 1FEF FF      .BYTE $FF,$EF,$F7,$FC,$B9,$DE,$F9,$F1
1204 1FF0 EF
1204 1FF1 F7
1204 1FF2 FC
1204 1FF3 B9
1204 1FF4 DE
1204 1FF5 F9
1204 1FF6 F1

```

```

CARD # LOC      CODE      CARD
1206      ;
1207      ;
1208      ;
1209      ;
1210      ;      INTERRUPT VECTORS
1211      ;
1212 1FF7      *=$1FFA
1213 1FFA 1C 1C      NMIENT .WORD NMIT
1214 1FFC 22 1C      RSTENT .WORD RST
1215 1FFE 1F 1C      IRQENT .WORD IRQT
1216      .END

```

END OF MOS/TECHNOLOGY 650X ASSEMBLY VERSION 4
NUMBER OF ERRORS = 0, NUMBER OF WARNINGS = 0

SYMBOL TABLE

SYMBOL	VALUE	LINE DEFINED		CROSS-REFERENCES						
ACC	00F3	76	587	851						
ADDR	10BE	694	687							
ADDRM	10C8	701	673							
AK	1EFE	1037	1080							
AK1	1F04	1041	1046							
CHAR	00FE	90	950	951	952	959	984	994		
CHK	1F91	1141	734	738	741	748	808	814	899	902
CHKH	17E8	97	158	265	289	299	301			
CHKHI	00F6	82	730	755	782	799	819	1144	1146	
CHKL	17E7	96	156	262	288	297	298			
CHKSUM	00F7	83	729	758	783	801	821	1142	1143	
CHKT	194C	295	234	237	242	244	256	308		
CHT1	1982	336	344							
CHT2	198E	341	338							
CHT3	1991	342	340							
CLEAR	1064	645	803	836						
CLKKT	1747	65	****							
CLKRDI	1747	66	355	361	378	384	498	505		
CLKRDT	1746	67	459	471						
CLK1T	1744	62	358	364	381	387	501	508		
CLK64T	1746	64	461	473						
CLK8T	1745	63	****							
CNTH30	17F3	101	613	622	1010	1022				
CNTL30	17F2	100	625	1012	1024					
CONVD	1F48	1085	1071	1074						
CONVD1	1F5B	1094	1095							
CRLF	1E2F	908	640	785	829					
DATA	10A8	680	****							
DATAM	10CC	704	675							
DATAM1	10CE	705	702							
DATAM2	10D0	706	699							
DATA1	10B0	686	698							
DATA2	10C3	697	692							
DEHALF	1EEB	1022	947	956						
DELAY	1ED4	1010	946	953	986	990	997	1003		
DETCPS	102A	612	****							
DET1	1031	615	617							
DET2	1042	623	621							
DET3	103A	619	624							
DE2	1EDD	1013	1018	1027						
DE3	1EE5	1017	1015							
DE4	1EDE	1014	1029							
DUMP	1D42	778	873							
DUMPT	1800	121	****							
DUMPT1	1814	131	134							
DUMPT2	1833	148	177							
DUMPT3	1854	163	166							
DUMPT4	1865	173	152							
DUMPV	1E01	873	867							
DUMPO	1D48	781	826							
DUMP1	1D4E	785	****							

SYMBOL	VALUE	LINE	DEFINED	CROSS-REFERENCES
DUMP2	1D86	811	817	
DUMP3	1D96	826	824	
DUMP4	1D7A	805	792	
EAH	17F8	106	151	791
EAL	17F7	105	149	789
FEED	1E07	876	861	
FEED1	1E12	882	880	
GETBYT	1F9D	1154	732	736 739 746 754 757 770
GETCH	1E5A	940	648	725 1154 1156
GETK	1C8D	667	****	
GETKEY	1F6A	1114	667	
GETKES	1F6C	1115	1119	
GET1	1E60	943	945	
GET2	1E6D	948	955	
GET5	1E6A	947	627	
GET6	1E87	962	944	
GOEXEC	1DC8	841	711	865
GOV	1CD9	711	679	
HEXALP	1FB8	1170	****	
HEXNUM	1FB4	1168	****	
HEXOUT	196F	323	314	316
HEXTR	1E4C	938	922	924
HEXTR1	1E55	933	931	
HEX1	1978	328	326	
ID	17F9	107	140	224 226
INCPT	1F63	1104	708	749 815 938
INCPT2	1F69	1107	1105	
INCVEB	19EA	397	176	258
INCVE1	19F2	400	398	
INH	00F9	85	647	780 825 1059 1179 1187
INITS	1E88	966	600	609
INIT1	1E8C	969	636	
INL	00F8	84	646	779 823 885 1066 1072 1158 1178 1185
INTVEB	1932	281	123	185
IRQENT	1FFE	1215	****	
IRQP27	1BFE	519	****	
IRQT	1C1F	604	1215	
IRQV	17FE	113	604	
KEYIN	1F7A	1122	1117	
KEYIN1	1F7C	1123	1126	
KEYIN2	1F82	1127	1124	
KEYIN3	1F8A	1133	1136	
KEYIN4	1F8D	1135	1132	
LOAD	1CE7	725	727	762 874
LOADER	1D3E	771	759	
LOADE1	1D3B	770	756	
LOADS	1CEE	728	****	
LOADT	1873	183	231	
LOADT4	18B5	216	221	
LOADT5	18D7	233	225	228
LOADT6	18EC	241	230	
LOADT7	18F8	247	239	259
LOADT8	1915	261	250	
LOADT9	1929	270	252	263 266

SYMBOL	VALUE	LINE	DEFINED	CROSS-REFERENCES															
LOADV	1E04	874	869																
LOAD10	192B	271	268																
LOAD11	19C2	223	218																
LOAD12	190F	258	182																
LOAD13	18FA	248	254																
LOAD2	1D0E	746	751																
LOAD3	1D1D	754	744																
LOAD7	1D2E	764	♦♦♦♦																
LOAD8	1D30	765	772																
MODE	00FF	91	686	705	967														
MODIFY	1E15	884	863																
NMIENT	1FFA	1213	♦♦♦♦																
NMIP27	1BFA	517	♦♦♦♦																
NMIT	1C1C	603	1213																
NMIV	17FA	111	603																
ONE	199E	353	336	339															
ONEKEY	1F02	1040	1116																
ONE1	19A1	355	356	368															
ONE2	19B0	361	362																
OPEN	1FCC	1185	796	828															
OUTBT	1961	309	141	157	159														
OUTBTC	195E	308	144	146	174														
OUTCH	1EA0	984	787	910	934														
OUTCHT	197A	333	132	138	155	164													
OUTSP	1E9E	983	831	835															
OUT1	1EB4	992	999																
PACK	1FAC	1164	651	1155	1157														
PACKT	1A00	413	251	405	407														
PACKT1	1A0F	421	418																
PACKT2	1A15	426	429																
PACKT3	1A22	433	414	416															
PADD	1741	59	970	1061	1079														
PBDD	1743	61	128	496	972														
PCCMD	1C0C	717	671																
PCH	00F0	73	594	719															
PCL	00EF	72	591	717															
PLLCAL	1A6B	493	517	518	519														
PLL1	1A75	498	499	511															
PLL2	1A84	505	506																
POINTH	00FB	87	170	272	595	696	720	737	790	843	881	897							
			1106	1188															
POINTL	00FA	86	169	271	592	688	691	695	719	740	747	788							
			812	833	845	877	879	886	900	1058	1104	1186							
PREG	00F1	74	589	847															
PRTBYT	1E3B	917	795	800	802	807	813	820	822	834	898	901							
PRTPNT	1E1E	897	797	809	830														
PRTST	1E31	909	642	767	912														
PRT1	1E3A	913	♦♦♦♦																
RDBIT	1A41	457	200	441	458														
RDBIT2	1A53	467	468																
RDBIT3	1A50	464	465																
RDBIT4	1A63	476	477																
RDBYT	19F3	404	223	233	236	241	243	261	264										
RDBYT2	19F9	406	♦♦♦♦																
RDCHT	1A24	439	209	216	248	404	406												
RDCHT1	1A29	441	446																

SYMBOL	VALUE	LINE	DEFINED	CROSS-REFERENCES
READ	106A	648	870	
RST	1022	606	1214	
RSTENT	1FFC	1214	****	
RSTP27	1BFC	518	****	
RSTV	17FC	112	****	
RTRN	1DC2	838	859 887	
SAD	1740	58	615 623 638 660 943 948 1044 1089 1091	
SAH	17F6	104	145 283	
SAL	17F5	103	143 281	
SAVE	1C00	587	****	
SAVE1	1C05	590	****	
SAVE2	1C0F	596	****	
SAVX	17E9	98	198 201 202 203 204 333 334 345 346 427	
			430 439 442 443 444 448 451	
SBD	1742	60	126 195 360 366 383 389 457 467 494 503	
			510 766 974 987 989 992 996 1000 1002 1041	
			1049 1077 1090	
SCAN	1DD8	854	652	
SCAND	1F19	1057	657 662 664	
SCANDS	1F1F	1060	****	
SCAND1	1F28	1066	1076	
SHOW	1DAC	829	839 882	
SHOW1	1DAF	830	643	
SPACE	1DA9	828	855	
SPUSER	00F2	75	599 608 841	
START	1C4F	636	171 273 601 616 658 661 669 706 709 721	
			768 872	
STEP	1CD3	708	677	
STV	1DFE	872	857	
SYNC	1891	197	211 220	
SYNC1	1896	200	206	
SYNC2	18AB	209	213	
TAB	1871	182	189 191	
TABLE	1FE7	1202	1087	
TEMP	00FC	88	684 689 917 923 925 1085 1099	
TIMH	17F4	102	1011 1016 1017 1023 1026	
TPX	00FD	89	940 958 985 1004	
TQP	1FD5	1197	909	
TTYKB	1C77	657	639 650	
TTYKB1	1C7C	659	663 665	
UPDATE	1FBB	1172	1169	
UPDAT1	1FC1	1177	1181	
UPDAT2	1FCB	1183	1165 1167	
VEB	17EC	99	122 148 150 173 184 188 190 192 235 238	
			257 282 284 286 397 399	
XREG	00F4	77	597 849	
YREG	00F5	78	596 850	
ZRD	19C4	376	341 342	
ZRD1	19C7	378	379 391	
ZRD2	19D6	384	385	

INSTRUCTION COUNT

ADC	13
AND	9
ASL	7
BCC	4
BCS	5
BEQ	26
BIT	12
BMI	9
BNE	44
BPL	15
BRK	0
BVC	0
BVS	0
CLC	8
CLD	1
CLI	0
CLV	0
CMP	38
CPX	1
CPY	0
DEC	2
DEX	14
DEY	8
EOR	2
INC	7
INX	5
INY	2
JMP	31
JSR	115
LDA	108
LDX	29
LDY	25
LSR	22
NOP	0
ORA	6
PHA	5
PHP	0
PLA	5
PLP	0
ROL	18
RTI	1
RTS	28
SBC	5
SEC	3
SED	0
SEI	1
STA	81
STX	14
STY	7
TAX	3
TAY	3
TSX	1
TXA	3
TXS	2
TYR	4

SYMBOLS = 204 (LIMIT = 400)

LINES = 1242 (LIMIT = 1500)

CTOP

BYTES = 1690 (LIMIT = 4096)

XREFS = 646 (LIMIT = 900)