

```

CARD # LCC      CCDE      CARD
1      ;MEMORY ADDRESS TEST
2      ;FOR EACH LCC IN TEST RANGE
3      ;CLEAR W-CLE RANGE
4      ; SET LOC TO $FF
5      ; VERIFY W-CLE RANGE $00 EXCEPT (LCC)
6      ; VERIFY (LCC) TO BE $FF
7      ;BREAK TO MONITOR ON ERROR WITH LOC IN (C,1)
8      ;PRINT "*" ON COMPLETION OF PASS & REPEAT
9      ;
10     0000      *=$0000      ;PAGE 0
11     ;
12     WRT      = $72C2
13     0000      LCC      *=$+2      ;TEST CELL ADDR
14     0002      LCH      *=$+2      ;LOWER LIMIT OF TEST
15     0004      HGH      *=$+2      ;UPPER LIMIT OF TEST+1
16     0006      PTR      *=$+2      ;POINTER TO CELL UNDER TEST
17     ;
18     0008      *=$0C10      ;START ADDR
19     ;
20     0010      A9 00      MAC      LDA #100      ;TYPE CR
21     0012      20 C2 72      JSR WRT
22     0015      A9 0A      LDA #10A      ;& LF
23     0017      20 C2 72      JSR WRT
24     ;
25     001A      20 68 00      JSR RSTLCC      ;LCC=LCH
26     001D      20 71 00      JSR RSTPTR      ;PTR=LCH
27     0020      A2 00      LDX #0
28     ;
29     ;CLEAR MEMORY AREA UNDER TEST
30     0022      A9 00      MLI      LDA #0
31     0024      E1 C6      STA (PTR,X)      ;STORE ZERO
32     0026      20 7A 00      JSR INCPTR      ;INCREMENT & TEST
33     0029      00 F7      BNE MLI      ;NEXT LCC
34     ;
35     ;PUT $FF IN SELETED CELL
36     002B      A9 FF      TEST      LDA #1FF
37     002C      81 00      STA (LOC,X)
38     ;VERIFY ALL CELLS ZERO EXCEPT (LCC)
39     002F      20 71 00      JSR RSTPTR      ;PTR=LCH
40     ;
41     0032      A1 06      VLCCP      LDA (PTR,X)      ;GET CELL
42     0034      FC 17      BEQ NEXTC      ;CK IF ZERO
43     0036      A4 06      LDY PTR      ;NOT ZERO--IS THIS (LCC)?
44     0038      C4 00      CFY LCC
45     003A      FC C1      BEQ CK1
46     003C      00      BRK      ;NOT (LCC)
47     ;
48     003D      A4 C7      OX1      LDY PTR+1

```

CARD #	LCC	CODE	CARD	
49	003F	C4 01	CFY LCC+1	
50	CC41	FC C1	PEC CK2	
51	0042	CC	BRK	;NCT (LCC)
52				
53	CC44	C9 FF	CK2 CMP #3FF	;IS (LCC)--IS DATA CK?
54	CC46	FC 01	BEQ CK3	
55	CC48	CC	BRK	;WRONG DATA
56				
57	CC49	A9 00	OK3 LDA #0	;RESET (LOC)
58	CC4B	81 C0	STA (LCC,X)	
59				
60	004D	20 7A 00	NEXTC JSR INCPTR	;NEXT CELL
61	0050	CC EG	BNE VLOOP	;IF NCT AT LIMIT
62				
63	0052	A5 CC	LDA LCC	;PRINT STAR EVERY PAGE ECUNDAI
64	0054	D0 07	BNE NCSTAR	
65	CC5E	A9 2A	LEA #10	
66	0058	20 C2 72	JSR WRT	
67	005B	A2 00	LCX #0	;FIX X AFTER MON CALL
68				
69	005D	20 8B 00	NGSTAR JSR INCLOC	;NEXT LCC
70	CC6C	CC C9	BNE TEST	
71				
72	CC62	20 68 00	JSR RSTLCC	;PASS COMPLETE
73	0065	4C 10 00	JMP MAC	;NEXT PASS
74				
75			;RESET LCC TO LOW	
76	0068	A5 02	RSTLCC LDA LCW	
77	CC6A	E5 CC	STA LCC	
78	006C	A5 C3	LDA LOW+1	
79	CC6E	E5 C1	STA LCC+1	
80	CC7C	6C	RTS	
81				
82			;RESET PTR TO LCW	
83	CC71	A5 C2	RSTPTR LDA LCW	
84	0073	85 C6	STA PTR	
85	CC75	A5 C3	LEA LCW+1	
86	CC77	85 C7	STA PTR+1	
87	CC79	6C	RTS	
88				
89			;INCREMENT PTR & CHECK FOR LIMIT	
90	CU7A	E6 C6	INCPTR INC PTR	;INCREMENT
91	CC7C	DC C2	BNE INC1	
92				
93	CC7E	E6 C7	INC PTR+1	
94				
95	CC80	A5 C4	INC1 LDA HIGH	;CHECK
96	CC82	C5 C6	CMP PTR	
97	CC84	DC C4	BNE IPRET	;NCT AT LIMIT

CARD #	LCC	CCCE	CARD
98			;
99	CC86	A5 C5	LDA HIGH+1
100	0088	C5 07	CMP PTR+1 ;Z=1 IF AT LIMIT
101			;
102	008A	60	IPRET RTS
103			;
104			;INCREMENT LCC & CHECK FOR LIMIT
105	008B	E6 C0	INCLOC INC LCC ;INCR
106	CC8C	DC 02	BNE INC2
107			;
108	008F	E6 01	INC LOC+1
109			;
110	CC91	A5 C4	INC2 LDA HIGH ;CHECK
111	0093	C5 00	CMP LOC
112	CC95	DC C4	BNE ILRET
113	0097	A5 05	LDA HIGH+1
114	0099	C5 01	CMP LCC+1 ;Z=1 IF AT LIMIT
115			
116	009B	60	ILRET RTS

SYMBOL TABLE

SYMBOL	VALUE	LINE	DEFINED	CROSS-REFERENCES
HIGH	CC04	15	95	99 110 113
ILRET	009B	116	112	
INCLOC	008B	105	60	
INC PTR	007A	50	32	60
INC1	008C	55	51	
INC2	CC91	110	106	
IPRET	008A	102	97	
LOC	CCCC	13	37	44 ^a 49 58 63 77 79 105 108 111
LOC			114	
LOC	CC02	14	76	78 83 85
MAC	CC1C	20	72	
ML1	CC22	30	32	
NEXTL	CC4C	60	42	
NCSTAR	CC5C	65	64	
CX1	CC3C	48	45	
CX2	CC44	53	50	
CX3	CC49	57	54	
PTR	CC06	16	31	41 43 48 84 86 90 93 96 100
RSTLOC	CC6E	76	25	72
RSTPTR	CC71	82	26	39
TEST	CC2E	36	70	
VLCPP	CC32	41	61	
WRT	72C2	12	21	22 66

```

TIM VERSION 1.0 - MEM PAGE C
CA#C # LCC CCDE CARD
 2
 3
 4
 5
 6
 7
 8 -----
 9
10 : PROMPTING CHARACTER IS A PERIOD (.)
11 : -----
12 :
13 :
14 : DISPLAY COMMANDS
15 : -----
16 :
17 : .R          DISPLAY REGISTERS (PC,F,A,X,Y,SP)
18 : .M ADDR    DISPLAY MEMORY ( 8 BYTES BEGINNING AT ADDR )
19 :
20 :
21 : ALTER COMMAND (:)
22 : -----
23 : .: DATA    ALTERS PREVIOUSLY DISPLAYED ITEM OR NEXT ITEM
24 :
25 :
26 : PAPER TAPE I/O COMMANDS
27 : -----
28 :
29 : .LH          LOAD HEX TAPE
30 : .WB ADDR1 ADDR2  WRITE BINARY TAPE (FROM LOW ADDR1 TO HIGH ADDR2)
31 : .WH ADDR1 ADDR2  WRITE HEX TAPE (FROM LOW ADDR1 TO HIGH ADDR2)
32 :
33 : CONTROL COMMANDS
34 : -----
35 :
36 : .G          GO, CONTINUE EXECUTION FROM CURRENT PC ADDRESS
37 :
38 : .H          TOGGLE HIGH-SPEED-READER OPTION
39 :             (IF ITS ON, TURNS IT OFF; IF OFF, TURNS ON)
40 :
41 : BRK AND NMI ENTRY POINTS TO TIM
42 : -----
43 :
44 : TIM IS NORMALLY ENTERED WHEN A 'BRK' INSTRUCTION IS
45 : ENCOUNTERED DURING PROGRAM EXECUTION. AT THAT
46 : TIME CPL REGISTERS ARE OUTPUT:  PC F A X Y SP
47 : AND CONTROL IS GIVEN TO THE KEYBOARD.
48 : USER MAY ENTER TIM BY PROGRAMMED BRK OR INDUCED NMI. NMI
49 : ENTRIES CAUSE A '*' TO PRECEDE THE '*' IN THE CPU REGISTER
50 : PRINTOUT FORMAT
51 :
52 : NON-BRK INTRQ (EXTERNAL DEVICE) INTERRUPT HANDLING
53 : -----

```

TIM VERSION 1.0 - MEM PAGE C

CARD #	LOC	CODE	CARD
54			;
55			;
56			A NCA-BRK INTRQ INTERRUPT CAUSES AN INDIRECT JUMP TO THE ADDRESS
57			LOCATED AT 'UINT' (HEX FFFB). THIS LOCATION CAN BE SET
58			USING THE ALTER CMD, OR LOADED AUTOMATICALLY IN PAPER TAPE
59			FORM WITH THE LH CMD IF THE USER ASSIGNS HIS INTRQ INTERRUPT
60			VECTOR TO \$FFFB IN THE SOURCE ASSEMBLY PROGRAM.
61			IF NOT RESET BY THE USER, UINT IS SET TO CAUSE EXTERNAL
62			DEVICE INTERRUPTS TO ENTER TIM AS NMIS. I.E.,
63			IF A NMI OCCURS WITHOUT AN INDUCED NMI SIGNAL, IT IS
64			AN EXTERNAL DEVICE INTERRUPT.
65			;
66			SETTING AND RESETTING PROGRAM BREAKPOINTS
67			-----
68			;
69			BREAKPOINTS ARE SET AND RESET USING THE MEMORY DISPLAY
70			AND ALTER COMMANDS. BRK HAS A '00' OPERATION CODE.
71			TO SET A BREAKPOINT SIMPLY DISPLAY THE MEMORY LOCATION
72			(FIRST INSTRUCTION BYTE) AT WHICH THE BREAKPOINT IS
73			TO BE PLACED THEN ALTER THE LOCATION TO '00'. THERE IS
74			NO LIMIT TO THE NUMBER OF BREAKPOINTS THAT CAN BE
75			ACTIVE AT ONE TIME.
76			TO RESET A BREAKPOINT, RESTORE THE ALTERED MEMORY LOCATION
77			TO ITS ORIGINAL VALUE.
78			WHEN AND IF A BREAKPOINT IS ENCOUNTERED DURING EXECUTION,
79			THE BREAKPOINT DATA PRECEDED BY AN '*' IS DISPLAYED.
80			THE PROGRAM COUNTER VALUE DISPLAYED IS THE BRK
81			INSTRUCTION LOCATION + 1.
82			;
83			-----
84			MOPK = \$CCCC1011C ; X,X,X,POR,DATA-Avail,GCT-CATA,SERIAL-CUT,IN
85			DAvail = \$C8
86			GCTCAT = \$C4
87			ICBASE = \$6F00
88			MPA = ICBASE+0
89			MPA = ICBASE+1
90			MPB = ICBASE+2
91			MPB = ICBASE+3
92			MCLKIT = ICBASE+4
93			MCLKRD = ICBASE+4
94			MCLKIF = ICBASE+5
95			UINT = \$FFFB
96			NCMDS = 7
97			MPC = \$7C0C
98			MP1 = \$7100
99			MP2 = \$720C
100			MP3 = \$730C
101			;
102			; ZERO PAGE MONITOR RESERVE AREA
103			;
104			CRCLY = 227 ; DELAY FOR CR IN BIT-TIMES
105			WRAP = 228 ; ADDRESS WRAP-AROUND FLAG

TIM VFRSIGN 1.0 - MEM PAGE C

CARD #	LCC	CCDE	CARD
106			DIFF =229
107			FSPTR =231
108			FSPCP =232
109			PREVC =233
110			MAJORT =234
111			MINCRT =235
112			ACMD =236
113			TMP0 =238
114			TMP2 =240
115			TMP4 =242
116			TMP6 =244
117			PCL =246
118			PCF =247
119			FLGS =248
120			ACC =249
121			XR =250
122			YR =251
123			SP =252
124			SAVX =253
125			TMPC =254
126			TMPC2 =255
127			RCNT =TMPC
128			LCNT =TMPC2
129			;
130			; 64 BYTE RAM MONITER RESERVE AREA
131			;
132			RAM64 =1FFC
133	0000		RAM64 =RAM64

MPO TIM PAGE 0

CARD #	LOC	CODE	CARD	
135			:	
136			:	
137			:	TIM PAGE 0 (RELATIVE)
138	FFC0			*=MPO
139			:	
140	7000	85 F9	NMINT	STA ACC ; SAVE A
141	7002	A9 23		LCA #*# ; SET A=# TO INDICATE NMINT ENTRY
142	7004	DC 55		RNF R3 ; JMP R3
143			:	
144	7006	A9 16	RESET	LCA #MDRK ; INIT DIR REG, PCR TC 1 RELCCATES
145			:	
146	7008	8C C3 6F		STA MDR
147			:	
148	7008	A2 C8		LDX #8 ; X=0
149	700C	8C F7 73	R1	LCA INTVEC-1,X ; INITIALIZE INT VECTORS
150	7010	9D F7 FF		STA UINT-1,X
151	7013	CA		DEX
152	7014	DO F7		RNE R1
153			:	
154	7016	86 EA		STX MAJORT ; INIT MAJOR T COUNT TO ZERO
155	7018	86 F7		STX HSPTR ; CLEAR HSPTR FLAGS
156	701A	86 FB		STX HSRCP
157	701C	CA		DEX ; X=FF
158	701D	9A		TXS ; SP=FF
159			:	
160			:	
161			:	COMPUTE BIT-TIME CONSTANT, X=FF
162	701E	AC C1		LDY #1 ; SET TC MEASLRE 2 BITS
163	7020	84 E3		STY CRDLY ;INIT CR DELAY TIME PARAMETER
164	7022	AD C2 6F	RC	LCA MPB ; WAIT FOR START
165	7025	4A		LSR A
166	7026	9C FA		RCC R0
167			:	
168	7028	8F 04 6E	R2	STX MCLKIT ; START CLOCK INITIALLY WITH FF
169	702B	AD 05 6F	R3	LCA MCLKIF
170	702F	1C 04		BPL R4
171	7030	E6 EA		INC MAJCRT ; COUNT MAJOR T
172	7032	DC F4		RNE R2 ; GC RESTART CLOCK WITH X = FF
173			:	
174	7034	98	R4	TYA
175	7035	4C C2 6E		ECR MPB
176	7038	29 01		AND #1
177	703A	FC FF		PEC R3 ; WAIT FOR Y BIT 0 AND SERIAL-IN NOT EQU
178	703C	88		CFY
179	703D	1C EC		RPL R3 ; LOOP UNTIL START OF BIT 2
180			:	
181	703F	AC 04 6E		LCA MCLKPD
182	7042	49 FF		ECR #FFF ; COMPLEMENT RESIDUE
183	7044	4A	R5	LSR A ; HALF IT
184	7045	46 EA		LSR MAJCRT ; HALF MAJOR
185	7047	9C 02		RCC R6
186	7049	C9 8C		ORA #8C ; PROPAGATE HC TO LC

MPO TIM PAGE C

CARD #	LCC	CCDE	CARD	
187	7C4B	CB	R6	INX
188	7C4C	FC F6		REC R5
189	7C4E	85 EB		STA MINGPT
190				;
191	7C50	58		CLI ; FNABLE INTS
192	7C51	CC		BRK ; ENTER TIM BY BRK
193				;
194	7052	85 F9	INTRQ	STA ACC ; SAVE ACC
195	7C54	68		PLA ; FLAGS TO A
196	7055	4F		PHA ; RESTORE STACK STATUS
197	7C56	29 10		AND #10 ; TEST BRK FLAG
198	7C58	FC 27		REC BX ; USER INTERRUPT
199				;
200	7C5A	CA		ASL A ; SET A=SPACE (10 X 2 = 20)
201	7C5B	85 FE	B3	STA TMPC ; SAVE INT TYPE FLAG
202	705C	CB		CLC ; CLEAR DECIMAL MODE
203	7C5E	4A		LSP A ; # IS ODD, SPACE IS EVEN
204				;
205				;
206	7C5F	86 FA		STX XR ; SAVE X
207	7C61	84 FB		STY YR ; Y
208	7C63	68		PLA
209	7C64	85 FB		STA FLGS ; FLAGS
210	7066	68		PLA
211	7C67	69 FF		ACC #FF ; CY SET TO PC-1 FOR BRK
212	7C69	85 FE		STA PCL
213	7068	68		PLA
214	7C6C	69 FF		ACC #FF
215	7C6E	85 F7		STA PCH
216	7C7C	EA		TSX
217	7C71	86 FC		STX SP ; SAVE ORIG SP
218				;
219	7C73	2C BA 72	B5	JSR CRLF
220	7C76	A6 FE		LDX TMPC
221				;
222	7C78	A9 2A		LEA #*
223	7C7A	2C CC 72		JSR WRTWD
224	7C7D	A9 52		LEA #*R ; SET FOR R DISPLAY TO PERMIT
225	7C7F	DC 16		BNE S0 ; IMMEDIATE ALTER FOLLOWING BREAKPOINT.
226				;
227	7C81	A5 F9	RX	LEA ACC
228	7C83	6C FB FF		JMP (LINT) ; CONTROL TO USER INTRQ SERVICE ROUTINE
229				;
230	7C86	A9 00	START	LEA #0 ;NEXT COMMAND FROM USER
231	7088	85 F7		STA HSPTR ;CLEAR P. S. PAPER TAPE FLAG
232	708A	85 F4		STA WRAP ;CLEAR ADDRESS WRAP-AROUND FLAG
233	7C8C	2C BA 72		JSR CRLF
234	708F	A9 2E		LEA #*. ; TYPE PROMPTING *.*
235	7C91	2C C6 72		JSR WRCC
236	7C94	20 E9 72		JSR RDCC ; READ CMD, CHAR RETURNED IN A
237				;
238	7C97	A2 C6	S0	LDX #ACMDS-1 ; LOCK-UP CMD

CARD #	LCC	CODE	CARD		
239	7099	CC 06 71	S1	CMP CMDS,X	
240	709C	DC 19		PNF S2	
241				:	
242	709E	A5 FC		LCA SAVX	: SAVE PREVIOUS CMD
243	70AC	B5 FS		STA PREVC	
244	7CA2	P6 FC		STX SAVX	: SAVE CURRENT CMD INDEX
245	7CA4	A5 71		LCA #MPI/256	: JMP INDIRECT TO CMD CODE
246	7CA6	B5 FD		STA ACMD+1	: ALL CMD CODE BEGINS CN MPI
247	7CAB	BC 0C 71		LCA ACRS,X	
248	7CAH	B5 EC		STA ACMD	
249	7DAD	EO 03		CFX #3	: IF :, R CR # (0, 1, CR 2) SPACE 2
250	7CAF	PC 03		PCS IJMP	
251	7CB1	2C 74 73		JSR SPAC2	
252				:	
253	7CB4	6C EC 00	IJMP	JMF (ACMD)	
254				:	
255	7CB7	CA	S2	DEX	
256	7CB8	1C DF		PFL S1	: LOOP FOR ALL CMDS
257				:	
258	7DBA	A9 3F	EPRDPP	LCA #? :	: OPERATOR ERR, TYPE '?', RESTART
259	7DBC	2C C6 72		JSP WRCC	
260	7DBF	9C C5		RCC START	: JMP START (WRCC RETURNS CY=0)
261				:	
262	7CC1	3E	DCMP	SEC	: TMP2-TMPO DOUBLE SUBTRACT
263	7CC2	A5 FO		LCA TMP2	
264	7CC4	F5 FE		SPC TMPO	
265	7CC6	B5 F5		STA DIFF	
266	7CCR	A5 F1		LCA TMP2+1	
267	7CCA	E5 EF		SEC TMPO+1	
268	7CCC	AB		TAY	:RETURN HIGH ORDER PART IN Y
269	7CCD	C5 F5		CRA DIFF	: CR LC FOR ECU TEST
270	7CCF	6C		RTS	
271				:	
272	7CDC	A5 EE	PUTP	LCA TMPO	: MOVE TMPO TO PCH,PCL
273	7CD2	B5 F6		STA PCL	
274	7CD4	A5 FF		LCA TMPO+1	
275	7CD6	B5 F7		STA PCH	
276	7CD8	6C		RTS	
277				:	
278	7CDS	A9 CC	ZTMP	LCA #C	: CLEAR REGS
279	7DCB	95 EE		STA TMPC,X	
280	7CDD	95 FF		STA TMPC+1,X	
281	7CDF	6C		RTS	
282				:	
283				: READ AND STORE BYTE. NO STORE IF SPACE OR RCNT=0.	
284				:	
285	7CEC	2C B3 73	PYTE	JSR RCOB	: CHAR IN A, CY=0 IF SP
286	7CE3	9C 1C		RCC BY3	: SPACE
287				:	
288	7CE5	A2 00		LTX #C	: STORE BYTE
289	7CE7	B1 EE		STA (TMPO,X)	
290				:	

MPO TIM PAGE C

CARD #	LCC	CCDF	CARD	
291	7CE9	C1 FE	CMP (TMPO,X)	; TEST FOR VALID WRITE (RAM)
292	7CEB	F0 05	REQ BY2	
293	7CED	68	PLA	; ERR, CLEAR JSR ADR IN STACK
294	7CFF	68	PLA	
295	7CFF	4C BA 7C	JMP ERROPR	
296				
297	70F2	2C 7C 72	BY2 JSR DADD	; INCR CKSUM
298	70F5	2C 97 73	BY3 JSR INCTMP	; GC INCR TMPC ADR
299	7CF8	C6 FE	DEC RCNT	
300	7CFA	6C	RTS	
301				
302	7CFB	A9 FB	SETR LCA #FLGS	; SET TC ACCESS REGS
303	70FD	85 FE	STA TMPC	
304	7CFF	A9 00	LEA #0	
305	71C1	85 FF	STA TMPC+1	
306	7103	A9 05	LEA #5	
307	71C5	6C	RTS	
308				
309	71C6	3A	CMDS .BYTE **:*	
310	71C7	52	.BYTE *R*	
311	7108	4C	.BYTE *M*	
312	71C9	47	.BYTE *G*	
313	710A	46	.BYTE *H*	
314	710B	4C	.BYTE *L*	
315	710C	57	.BYTE *W*	; W MUST BE LAST CMD IN CHAIN
316	710D	3A	ADRS .BYTE ALTER-MP1	
317	71CE	14	.BYTE DSPLYR-MP1	
318	71CF	1C	.BYTE DSPLYM-MP1	
319	711C	5C	.BYTE GC-MP1	
320	7111	6F	.BYTE HSP-MP1	
321	7112	74	.BYTE LH-MP1	
322	7113	C2	.BYTE WO-MP1	

MPI TIM PACF 1

CARD #	LOC	CODE	CARD	
324				;
325				;
326				;
327				;
328				;
329				;
330	7114	20 A6 72	DSPLYR JSR WRPC	; WRITE PC
331	7117	20 FB 7C	JSR SETR	
332	711A	DC 07	BNF #C	; USE DSPLYM
333				;
334	711C	20 A4 73	DSPLYM JSR R0CA	; READ MEM ACR INTO TMP0
335	711F	9C 16	BCC ERRS1	; ERR IF NO ACR
336	7121	A9 08	LCA #8	
337	7123	85 FE	MC STA TMP0	
338	7125	AC 0C	LDY #C	
339	7127	20 77 73	M1 JSR SPACE	; TYPE 8 BYTES CF MEM
340	712A	B1 FE	LCA (TMP0),Y	; (TMP0) PRESERVED FOR POSS ALTER
341	712C	20 B1 72	JSP WROB	
342	712F	C8	INX	; INCR INDEX
343	7130	C6 FE	DEC TMP0	
344	7132	D0 F3	BNE M1	
345	7134	4C E6 7C	BEGS1 JMP START	
346				;
347	7137	4C BA 7C	ERRS1 JMP ERRCPR	
348				;
349				; ALTER LAST DISPLAYED ITEM (ACR IN TMP0)
350				;
351	713A	C6 E9	ALTER DEC PREVC	; R INDEX = 1
352	713C	D0 0C	BNE A3	
353				;
354	713E	20 A4 73	JSR R0CA	; CY=C IF SP
355	7141	9C 03	BCC A2	; SPACE
356	7143	20 DC 7C	JSR PLTP	; ALTER PC
357	7146	20 FB 7C	A2 JSR SETR	; ALTER R'S
358	7149	DC C5	BNE A4	; JMP A4 (SETR RETURNS ACC = 5)
359	714B	20 9A 72	A3 JSR WROA	; ALTER M, TYPE ACR
360	714E	A9 C8	LCA #8	; SET CNT=8
361				;
362	7150	85 FE	A4 STA RCNT	
363	7152	20 77 73	A5 JSR SPACE	; PRESRVES Y
364	7155	20 EC 7C	JSR BYTE	
365	7158	DC F8	BNE A5	
366	715A	FC D8	A9 PEC BECS1	
367				;
368	715C	A6 FC	GC LCX SP	
369	715E	9A	TXS	; ORIG CR NEW SP VALUE TO SP
370	715F	A5 F7	LCA PCH	
371	7161	48	PHA	
372	7162	A5 F6	LCA PCL	
373	7164	48	PHA	
374	7165	A5 F8	LCA FLGS	
375	7167	48	PHA	

MPI TIM PAGE 1

CARD #	LCC	CODE	CARD		
376	7168	A5 F9		LCA	ACC
377	716A	A6 FA		LDX	XR
378	716C	A4 FR		LCY	YR
379	716F	4C		RTI	
380					
381	716F	E6 E8	HSP	INC	HSPROP ; TOGGLE BIT C
382	7171	4C 86 7C		JMP	START
383					
384	7174	20 E9 72	LH	JSR	RCCC ; REAC SECCNC CMD CHAR
385	7177	2C 8A 72		JSR	CRLF
386	717A	A6 E8		LDX	HSPROP ; ENABLE PTR OPTICK IF SET
387	717C	86 E7		STX	HSPTR
388	717E	2C F9 72	LH1	JSR	RCCC
389	7181	C9 3E		CMP	#* ; FIND NEXT RCD MARK (;)
390	7183	DC F5		BNE	LH1
391					
392	7185	A2 04		LDX	#4
393	7187	2C 09 7C		JSR	ZTMP ; CLEAR CKSUM REGS TMP4
394	718A	2C B3 73		JSR	RDOB
395	718C	DC 06		BNE	LH2
396					
397	718F	A2 0C		LDX	#C ; CLEAR HS RCR FLAG
398	7191	86 E7		STX	HSPTR
399	7193	FC 9F		BEC	BECS1 ; FINISHED
400					
401	7195	85 FE	LH2	STA	RCNT ; RCNT
402	7197	2C 7C 72		JSR	DADD ; RCD LGTH TC CKSUM
403	719A	2C B3 73		JSR	RDOB ; SA HC TC TMP4+1
404	719D	85 EF		STA	TMP4+1
405	719F	2C 7C 72		JSR	DADD ; ADD TC CKSUM
406	71A2	2C B3 73		JSR	RCCB ; SA LD TC TMP4
407	71A5	85 EE		STA	TMP4
408	71A7	2C 7C 72		JSR	DADD ; ADD TC CKSUM
409					
410	71AA	2C EC 7C	LH3	JSR	BYTE ; BYTE SUB/R DECRS RCNT ON EXIT
411	71AC	DC FB		BNE	LH3
412	71AF	2C A4 73		JSR	RCCA ; CKSUM FROM HEX RCD TO TMP4
413	71B2	A5 F2		LCA	TMP4 ; TMP4 TC TMP2 FOR DCMP
414	71B4	85 FC		STA	TMP2
415	71B6	A5 F3		LCA	TMP4+1
416	71B8	85 F1		STA	TMP2+1
417	71BA	2C C1 7C		JSR	DCMP
418	71BD	FC BF		BEC	LH1
419	71BF	4C BA 70	ERRP1	JMP	ERRPR
420					
421	71C2	2C E9 72	WC	JSR	RDOC ; RC 2ND CMD CHAR
422	71C5	85 FE		STA	TMP4
423	71C7	2C 77 73		JSR	SPACE
424	71CA	2C A4 73		JSR	RDOA
425	71CC	2C 87 73		JSR	T2T2 ; SA TC TMP2
426	71DC	2C 77 73		JSR	SPACE ; SPACE BEFCRE NEXT ADDRESS
427	71E3	2C A4 73		JSR	PCOA

MPI TIM PAGE 1

CARD #	LCC	CCDE	CARD	
428	71D6	2C 87 73	JSR T2T2	; SA TC TMPD, EA TC TMP2
429	71C9	2C F9 72	JSR RDOC	; DELAY FOR FINAL CR
430	71DC	A5 FE	LCA TMPC	
431				
432	71CE	C9 48	CMP #*H	
433	71EC	DC 59	PNE WB	
434				
435	71E2	A6 E4	WHO LCX WRAP	; IF ADDR HAS WRAPPED AROUND
436	71E4	CC 52	PNE BCCST	; THEN TERMINATE WRITE OPERATION
437				
438	71E6	2C 8A 72	JSR CRLF	
439	71E9	A2 18	LDX #24	
440	71EB	86 FE	STX RCNT	; RCNT=24
441	71ED	A2 04	LCX #4	; CLEAR CKSUM
442	71EF	2C 09 7C	JSR ZTMP	
443				
444	71F2	A9 38	LCA #*;	
445	71F4	2C C6 72	JSR WROC	; WR RCD MARK
446				
447	71F7	2C C1 7C	JSR DCMP	; EA-SA (TMPD+2-TMPO) DIFF IN LOC DIFF,+1
448	71FA	98	TYA	; MS BYTE CF DIFF
449	71FB	DC CA	PNE WF1	
450	71FC	A5 F5	LCA DIFF	
451	71FF	C9 17	CMP #23	
452	72C1	BC C4	BCC WH1	; DIFF GT 24
453	72C3	85 FE	STA RCNT	; INCR LAST RCNT
454	72C5	E6 FE	INC RCNT	
455	72C7	A5 FE	WH1 LCA RCNT	
456	7209	2C 7C 72	JSR DADC	; ADD TO CKSUM
457	720C	2C 81 72	JSR WRCE	; RCC CNT IN A
458	720F	A5 EF	LCA TMPC+1	; SA HC
459	7211	2C 7C 72	JSR CADC	
460	7214	2C 81 72	JSR WRCE	
461	7217	A5 EE	LCA TMPC	; SA LC
462	7219	2C 7C 72	JSR CADC	
463	721C	2C 81 72	JSR WRCE	
464				
465	721F	AC CC	WH2 LCY #0	
466	7221	B1 EE	LCA (TMPO),Y	
467				
468	7223	2C 7C 72	JSR CADC	; INC CKSUM, PRESERVES A
469	7226	2C F1 72	JSR WPOB	
470	7229	2C 57 73	JSR INCTMP	; INC SA
471	722C	C6 FE	DEC RCNT	
472	722E	CC EF	PNE WF2	; LOCP FOR LP TC 24 BYTES
473				
474	7230	2C 5E 72	JSR WRCA4	; WRITE CKSUM
475				
476	7233	2C C1 7C	JSR DCMP	
477	7236	80 AA	BCC WHC	; LOCP WHILE EA GT CR = SA
478	7238	4C 86 7C	BCCST JMP START	
479				

MPL TIM PAGE 1

CARD #	LCC	CODE	CARD	
480			;	
481	723B	E6 FD	WB	INC SAVX ; SAVX TO = NCMDS FOR ASCII SUB/R
482	723D	A5 E4	WB1	LCA WRAP ; IF ADCR HAS WRAPPED ARUNC
483	723F	CC F7		BNE BCCST ; THEN TERMINATE WRITE OPERATION
484			;	
485	7241	A9 04		LCA #4
486	7243	85 EC		STA ACMD
487	7245	2C 8A 72		JSR CRLF
488	7248	2C 9A 72		JSR WROA ; OUTPLT HEX ADR
489			;	
490	724B	2C 77 73	WRNPF	JSR SPACE
491	724E	A2 09		LDX #9
492	7250	86 FE		STX TMPC ; LCCP CNT =9
493	7252	A1 E5		LCA (TMPC-9,X)
494	7254	85 FF		STA TMPC2 ; BYTE TO TMPC2
495	7256	A9 42		LCA #'B
496	7258	00 08		BNE WBF2 ; WRITE B
497			;	
498	725A	A9 5C	WBF1	LCA #'P
499	725C	C6 FF		ASL TMPC2
500	725E	8C C2		BCC WBF2
501	7260	A9 4E		LCA #'N
502			;	
503	7262	2C C6 72	WBF2	JSR WRCC ; WRITE N CR P
504	7265	C6 FE		DEC TMPC
505	7267	CC F1		BNE WBF1 ; LCCP
506	7269	A9 46		LDA #'F
507	726B	2C C6 72		JSR WPOC ; WRITE F
508			;	
509	726E	2C 97 73		JSR INCTMF
510			;	
511	7271	C6 EC		DEC ACMD ; TEST FOR MULTIPLE OF FOUR
512	7273	CC D6		BNE WRNPF
513			;	
514	7275	2C C1 7C		JSR CCMP
515	7278	8C C3		BCC WB1 ; LCCP WHILE EA GT CR = SA
516	727A	9C 8C		BCC BCCST
517			;	
518	727C	4F	WACC	PHA ; SAVE A
519	727D	18		CLC
520	727E	65 F2		ACC TMP4
521	7280	85 F2		STA TMP4
522	7282	A5 F3		LCA TMP4+1
523	7284	69 CC		ADC #C
524	7286	85 F3		STA TMP4+1
525	7288	6E		PLA ; RESTORE A
526	7289	6C		RIS
527			;	
528	728A	A2 0D	CRLF	LDX #10C
529	728C	A9 CA		LCA #10A
530	728E	2C C0 72		JSR WRTWC
531	7291	A6 F3		LDX CRDLY ; BIT-TIME CCUNT FOR DELAY

MPL TIM PAGE 1

CARD #	LCC	CODE	CARD		
532	7293	2C 1C 73	CR1	JSR DLY2	;DELAY CF ONE BIT-TIME
533	7296	CA		DEX	
534	7297	CG FA		BNE CR1	
535	7295	6C		RTS	
536				;	
537				;	WRITE ADR FROM TMPC STORES
538				;	
539	729A	A2 C1	WROA	LDX #1	
540	729C	CO OA		BNE WROA1	
541	729E	A2 C5	WRCA4	LDX #5	
542	72A0	CC C6		BNE WROA1	
543	72A2	A2 07	WRCA6	LDX #7	
544	72A4	DC C2		BNE WRCA1	
545	72A6	A2 09	WRPC	LDX #9	
546	72A8	B5 ED	WRCA1	LCA TMPO-1,X	
547	72AA	4E		PHA	
548	72AB	B5 EE		LCA TMPC,X	
549	72AD	2C B1 72		JSR WRQB	
550	72B0	6E		PLA	
551				;	
552				;	WRITE BYTE - A = BYTE
553				;	UNPACK BYTE DATA INTO TWO ASCII CHARS. A=BYTE; X,A=CHARS
554				;	
555	72B1	4E	WRQB	PHA	
556	72B2	4A		LSR A	
557	72B3	4A		LSR A	
558	72B4	4A		LSR A	
559	72B5	4A		LSR A	
560	72B6	2C 58 73		JSR ASCII	; CONVERT TO ASCII
561	72B9	AA		TAX	
562	72BA	6E		PLA	
563	72BB	25 CF		AND #10F	
564	72BC	2C 58 73		JSR ASCII	
565				;	
566				;	WRITE 2 CHARS - X,A = CHARS
567				;	
568	72CC	4E	WRTW0	PHA	
569	72C1	8A		TXA	
570	72C2	2C C6 72		JSR WRT	
571	72C5	6E		PLA	
572				;	
573				;	WRITE SERIAL OUTPUT
574				;	A = CHAR TO BE OUTPUT
575				;	
576	72C6	2C 1C 73	WRT	JSR DLY2	
577	72C9	A2 C9		LDX #9	
578			WROC	=WRT	
579	72CB	45 FF		FCR #1FF	; COMPLEMENT A
580	72CC	3E		SEC	
581				;	
582	72CE	2C CA 72	WRT1	JSR CLT	
583	72D1	2C 1C 73		JSR DLY2	

MPL TIM PAGE 1

CARD #	LCC	CCDE	CARD		
584	72D4	4A		LSR A	
585	72D5	CA		DEX	
586	72D6	CC F6		BNE WRT1	
587	72D8	FC 3F		BEC RDT5	
588					
589					; *LSE BNE?
590	72DA	4E	OUT	PFA	; SAVE A
591	72DB	AC 02 6E		LCA MPB	; OUTPUT BIT FROM CY
592	72DE	29 FD		AND #111111101	
593	72EC	90 02		RCC OLT1	
594	72E2	C9 02		ORA #100000010	
595	72E4	8C C2 6E	OUT1	STA MPB	
596	72E7	6E		PLA	; RESTORE A
597	72E8	6C		RTS	
598					
599					; OUTPUT RETURNS CHAR IN A
600					
601	72E9	A5 E7	RDT	LCA HSPTR	; TEST HS PTR OPTICA
602	72FB	4A		LSR A	
603	72EC	8C 4F		BCC RCHSR	
604			RDOC	=RCT	
605	72EE	A2 08		LCA #8	
606					
607	72FC	AC C2 6E	RCT1	LCA MPB	
608	72F3	4A		LSR A	; WAIT FOR START BIT
609	72F4	9C FA		BCC RDT1	
610					
611	72F6	2C 2C 73		JSR DLY1	
612	72F9	2C DA 72		JSR CLT	; ECHC START BIT
613					
614	72FC	2C 1D 73	RCT2	JSR DLY2	
615	72FF	AC C2 6E		LCA MPB	; CY = NEXT BIT
616	7302	4A		LSR A	
617	7303	2C DA 72		JSR CLT	; ECHC
618					
619	7306	CB		PFP	; SAVE BIT
620	7307	5E		TYA	; Y CONTAINS CHAR BEING FORMED
621	7308	4A		LSR A	
622	7309	2E		PLF	; RECALL BIT
623	730A	9C C2		BCC RDT4	
624	730C	C9 8C		CFA #180	; ADD IN NEXT BIT
625	730E	AE	RDT4	TAY	
626	730F	CA		DEX	
627	731C	DC EA		BNE RCT2	; LOOP FOR 8 BITS
628	7312	45 FF		ECR #1FF	; COMPLEMENT DATA
629	7314	29 7F		AND #17F	; CLEAR PARITY
630					
631	7316	2C 1D 73		JSR DLY2	
632	7319	18	RDT5	CLC	
633	731A	2C DA 72		JSR CLT	; AND DELAY 2 HALF-BIT-TIMES
634					
635	731D	2C 2C 73	DLY2	JSR DLY1	

MPI TIM PAGE 1

CARD #	LCC	CODE	CARD		
636	7320	48	DLY1	PHA	; SAVE FLAGS AND A
637	7321	C8		P+P	
638	7322	8A		TXA	; SAVE X
639	7323	48		P+A	
640	7324	A6 EA		LCX MAJCRT	
641	7326	A5 EB		LCA MINORT	
642					
643	7328	8C C4 6E	DL2	STA MCLK1T	
644					
645	7328	AC C5 6E	DL3	LCA MCLKIF	
646	732E	1C FB		BPL DL3	
647	7230	CA		CEX	
648	7331	C8		P+P	
649	7332	AC C4 6E		LCA MCLKRD	; RESET TIMER INT FLAG
650	7335	28		PLP	
651	7336	1C F3		BPL DL3	
652					
653	7338	68		PLA	; RESTCRE REGS
654	7339	AA		TAX	
655	733A	28		PLP	
656	733B	6E		PLA	
657	733C	6C	DLX	RTS	
658					
659	733D	AC C2 6E	RDHSR	LCA MPB	; LCCF CN DATA AVAIL
660	734C	29 C8		AND #DAVA1L	
661	7342	FC F9		BEC RCHSR	
662					
663	7344	AE 00 6E		LCX MPA	; READ DATA
664	7347	AC C2 6E		LCA MPB	; SEND CCT-DATA PULSE
665	734A	C5 C4		ORA #GOTDAT	
666	734C	8C C2 6E		STA MPB	
667	734F	25 FB		AND #11111011	
668	7351	8C 02 6E		STA MPB	
669	7354	8A		TXA	
670	7355	25 7F		AND #7F	
671	7357	6C		RTS	
672					
673	7358	18	; ASCII	CLC	
674	7359	69 C6		ACC #6	
675	735B	69 FC		ACC #7F0	
676	735C	9C C2		BCC ASC1	
677	735F	69 C6		ACC #106	
678					
679	7361	69 3A	; ASC1	ACC #13A	
680	7363	4E		P+A	; TEST FOR LETTER B IN ADR DURING WBNPF
681	7364	C9 42		CMP #*B	
682	7366	CC CA		BNE ASCX	
683	7368	A5 FD		LCA SAVX	
684	736A	C9 C7		CMP #ACMDS	
685	736C	CC C4		BNE ASCX	; NOT WB CMD
686	736E	68		PLA	
687	736F	A9 2C		LCA #*	; FOR WB, BLANK B'S IN ADR

MPI TIM PAGE 1

CARD #	LCC	CCCC	CARD			
688	7371	48			PFA	
689	7372	68	ASCX		PLA	
690	7373	6C			RTS	
691						
692	7374	2C 77 73	SPAC2	JSR	SPACE	
693	7377	48	SPACE		PFA	; SAVE A,X,Y
694	7378	EA			T>A	
695	7379	48			PFA	
696	737A	98			TYA	
697	737B	48			PFA	
698	737C	A9 2C			LCA #*	
699	737E	2C C6 72		JSR	WRT	; TYPE SP
700	7381	68			PLA	; RESTORE A,X,Y
701	7382	A8			TAY	
702	7383	6F			PLA	
703	7384	AA			TAX	
704	7385	68			PLA	
705	7386	6C			RTS	
706						
707	7387	A2 C2	T2T2		LCX #2	
708	7389	B5 ED	T2T21		LCA TMP0-1,X	
709	738E	48			PFA	
710	738C	B5 FF			LCA TMP2-1,X	
711	738E	95 ED			STA TMP0-1,X	
712	7390	68			PLA	
713	7391	95 FF			STA TMP2-1,X	
714	7393	CA			DEX	
715	7394	DC F3			PNE T2T21	
716	7396	6C			RTS	
717						
718						; INCREMENT (TMP0,TMP0+1) BY 1
719	7397	E6 FE	INCTMP	INC	TMP0	;LCW BYTE
720	7399	FC C1		REQ	INCT1	
721	739E	6C			RTS	
722						
723	739C	E6 EF	INCT1	INC	TMP0+1	;HIGH BYTE
724	739E	FC C1		BEQ	SETWRP	
725	73AC	6C			RTS	
726						
727	73A1	E6 E4	SETWRP	INC	WRAP	;PCINTER HAS WRAPPED AROUND - SET FLAG
728	73A3	6C			RTS	
729						
730						; READ HEX ADR, RETURN HD IN TMP0, LD IN TMP0+1 AND CY=1
731						; IF SP CY=0
732						
733	73A4	2C B3 73	R00A	JSR	R00B	; READ 2 CHAR BYTE
734	73A7	9C C2		BCC	R00A2	; SPACE
735						
736	73A9	E5 EF			STA TMP0+1	
737	73AB	2C B3 73	R00A2	JSR	R00B	
738	73AE	9C C2		PCC	R0EXIT	; SP
739	73BC	P5 EE			STA TMP0	

MPI TIM PAGE 1

CARD #	LCC	CODE	CARD	
740	73B2	6C	RDEXIT	RTS
741			:	
742			:	READ HEX BYTE AND RETURN IN A, AND CY=1
743			:	IF SF CY=0
744			:	Y REG IS PRESERVED
745			:	
746	73B3	58	RCCB	TYA ; SAVE Y
747	73B4	48		PFA
748	73B5	A9 00		LEA #C ; SET DATA = C
749	73B7	85 EC		STA ACMD
750	73B9	20 E9 72		JSR RDOC
751	73BC	C9 CD		CMP #10C ; CR?
752	73BE	DC C6		BNE RDOB1
753	73C0	68		PLA ; YES - GO TO START
754	73C1	68		PLA ; CLEANING STACK UP FIRST
755	73C2	68		PLA
756	73C3	4C 86 7C		JMP START
757			:	
758	73C6	C9 20	RCCB1	CMP #* ; SPACE
759	73C8	DC CA		BNE RCCB2
760	73CA	2C E9 72		JSR RDOC ; READ NEXT CHAR
761	73CC	C9 20		CMP #*
762	73CF	DC CF		BNE RCCB3
763	73D1	1E		CLC ; CY=C
764	73D2	9C 12		BCC RDOB4
765			:	
766	73D4	2C E9 73	RDOB2	JSR HEXIT ; TC HEX
767	73D7	CA		ASL A
768	73D8	CA		ASL A
769	73D9	CA		ASL A
770	73DA	CA		ASL A
771	73DB	85 EC		STA ACMD
772	73DC	2C F9 72		JSR RDOC ; 2ND CHAR ASSUMED HEX
773	73EC	2C E9 73	RCCB3	JSR HEXIT
774	73E3	C5 FC		ORA ACMD
775	73E5	3E		SEC ; CY=1
776	73E6	AA	RDOB4	TAX
777	73E7	68		PLA ; RESTORE Y
778	73E8	A8		TAY
779	73E9	8A		TXA ; SET Z & N FLAGS FOR RETURN
780	73EA	6C		RTS
781			:	
782	73EB	C5 3A	HEXIT	CMP #3A
783	73EC	CB		PFP ; SAVE FLAGS
784	73EE	25 CF		AND #10F
785	73FC	28		PLP
786	73F1	9C C2		BCC HEX09 ; 0-9
787	73F3	65 C8		ACC #E ; ALPHA ADD 8+CY=9
788	73F5	6C	HEXC9	RTS
789			:	
790	73FE			*=MP3+\$FE
791			:	

MPI TIM PAGE 1

CARD #	LCC	CCDE	CARD	
792	73F8	CC 7C	INTVEC	.WCRD NMINT ; DEFAULT USER INTRQ TC NMINT
793	73FA	CO 70		.WCRD NMINT
794	73FC	CE 7C		.WCRD RESET
795	73FE	52 7C		.WCRD INTRQ
756				;

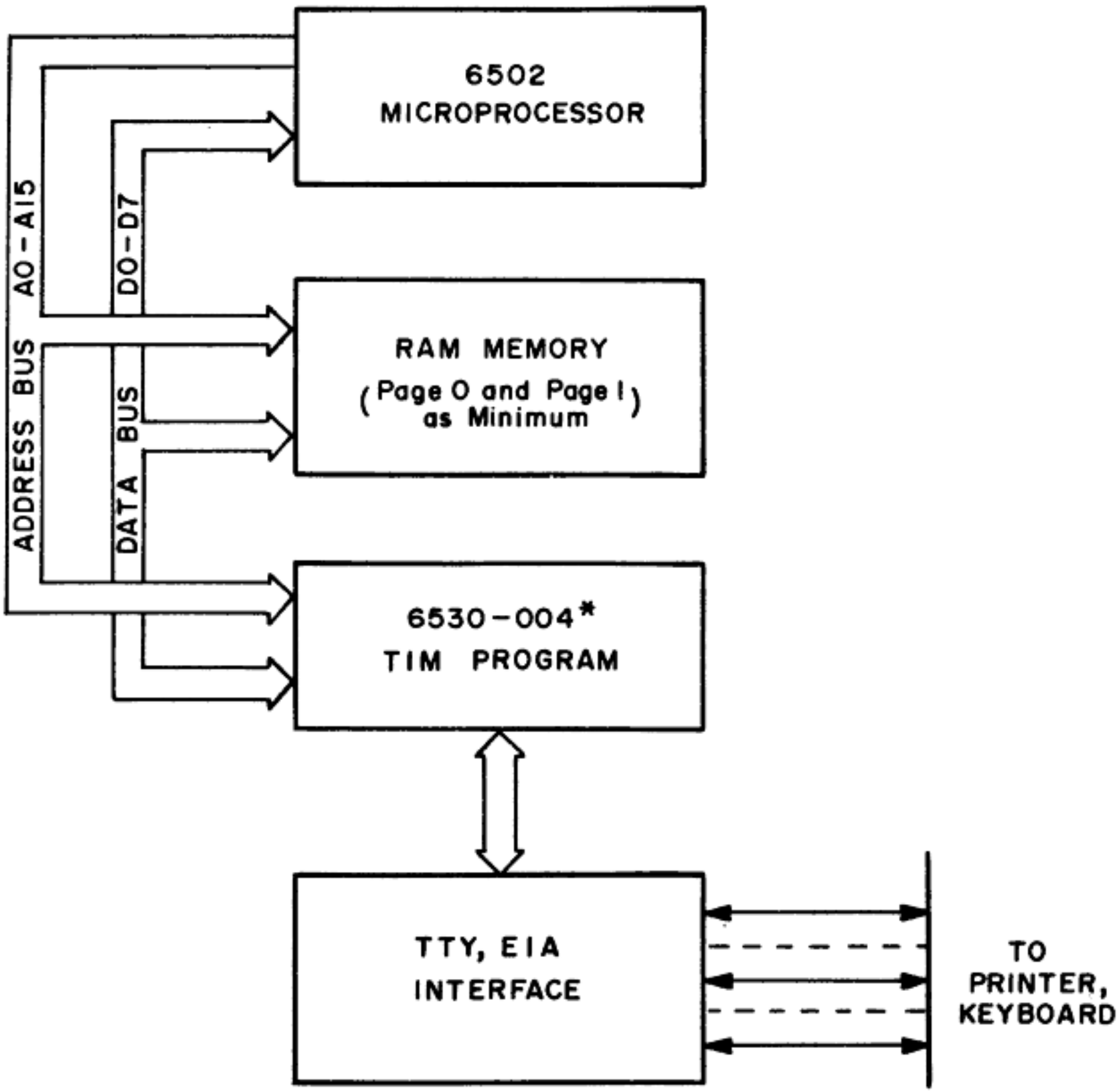
SYMBOL TABLE

SYMBOL	VALUE	LINE	DEFINED	CROSS-REFERENCES								
ACC	00F9	120	140	154	227	376						
ACMD	00EC	112	246	248	253	486	511	749	774	774		
ACRS	7100	316	247									
ALTER	713A	351	316									
ASCII	735E	673	560	564								
ASCX	7372	689	682	685								
ASCI	7361	679	676									
A2	7146	357	355									
A3	714B	359	352									
A4	7150	362	358									
A5	7152	363	365									
A9	715A	366										
BCCST	723B	478	436	483	516							
BEQSI	7134	345	366	359								
BX	7081	227	198									
BYTE	70EC	285	364	410								
BY2	70F2	297	292									
BY3	70F5	298	286									
B3	705B	201	142									
B5	7073	219										
CMDS	7106	309	235									
CRDLY	00E3	104	163	531								
CRLF	72BA	528	215	233	385	438	487					
CR1	7293	532	534									
CADD	727C	518	297	402	405	408	456	459	462	468		
CAVAIL	000B	85	660									
DCMP	70C1	262	417	447	476	514						
DIFF	00E5	106	265	269	450							
CLX	7330	657										
CLY1	7320	636	611	635								
CLY2	7310	635	532	576	583	614	631					
CL2	7328	643										
CL3	732B	645	646	651								
CSFLYM	7110	324	318									
CSPLYR	7114	330	317									
ERROPR	70BA	258	295	347	419							
ERRPI	71PF	419										
ERRSI	7137	347	335									
FLGS	00F8	119	209	302	374							
GC	7150	368	319									
GOTDAT	CCC4	86	665									
HEXIT	73EB	782	766	773								
HFXCS	73F5	788	786									
HSP	716F	381	320									
HSPTR	00E7	107	155	231	387	398	601					
HSROP	CCF8	108	156	381	386							
IJMP	70B4	253	250									
INCTMF	7397	719	298	470	509							
INCT1	739C	723	720									
INTRQ	7052	154	795									
INTVEC	73FB	792	149									

SYMBOL	VALUE	LINE DEFINED		CRCS-REFERENCES											
				87	88	89	90	91	92	93	94				
ICBASE	6ECC		87	88	89	90	91	92	93	94					
LCNT	COFF		128												
LH	7174		384	321											
LH1	717E		388	390	418										
LH2	7195		401	395											
LH3	71AA		410	411											
MAJORT	COEA		110	154	171	184	640								
MCLKIF	6E05		94	165	645										
MCLKRD	6FC4		93	181	649										
MCLKIT	6E04		92	168	643										
MDA	6E01		89												
MCR	6EC3		91	146											
MCRK	CO16		84	144											
MINORT	COFP		111	189	641										
MPA	6ECC		88	663											
MPB	6FC2		90	164	175	591	595	607	615	655	664	666	668		
MPC	70CC		97	138											
MP1	71CC		98	245	316	317	318	319	320	321	322				
MP2	72CC		99												
MP3	73CC		100	790											
MO	7123		337	332											
M1	7127		339	344											
NCMDS	COO7		96	238	664										
NMINT	70CC		140	792	793										
CLT	72CA		590	582	612	617	633								
CUT1	72E4		595	593											
FCH	COF7		118	215	275	370									
PCL	COF6		117	212	273	372									
FFEVC	COE9		109	243	351										
PUTP	70CC		272	356											
RAM64	FFCC		132	133											
FCNT	COFE		127	299	362	401	440	453	454	455	471				
RDEXIT	73B2		740	738											
FCFSR	7330		655	603	661										
RCCA	73A4		733	334	354	412	424	427							
RCCA2	73AB		737	734											
RCCB	73B3		746	285	394	403	406	733	737						
RCCB1	73C6		758	752											
RCCB2	73C4		766	755											
RCCB3	73EC		773	762											
RCCB4	73E6		776	764											
RCCC	72E9		604	236	384	388	421	425	750	760	772				
RCT	72E9		601	604											
RCT1	72FC		607	609											
RCT2	72FC		614	627											
RCT4	73CF		625	623											
RCT5	7319		632	587											
RESET	70C6		144	794											
RC	7022		164	166											
R1	70CC		149	152											
R2	7028		168	172											
R3	702P		165	177	179										
R4	7034		174	170											
R5	7044		183	188											

SYMBOL	VALUE	LINE	DEFINED	CROSS-REFERENCES																	
RE	704R	187	185																		
SAVX	00FC	124	242	244	481	683															
SETR	7CF8	302	331	357																	
SETWRP	73A1	727	724																		
SP	COFC	123	217	368																	
SPACE	7377	653	335	363	423	426	450	652													
SPAC2	7374	692	251																		
START	7086	230	260	345	382	478	756														
SG	7097	238	225																		
S1	7099	239	256																		
S2	70B7	255	240																		
TMPC	COFE	125	127	201	220	337	343	422	430	492	504										
TMPC2	COFF	126	128	494	499																
TMPC	COFE	113	264	267	272	274	279	280	285	291	303	305									
			340	404	407	458	461	466	453	546	548	708									
			711	719	723	736	739														
TMP2	COFC	114	263	266	414	416	710	713													
TMP4	COF2	115	413	415	520	521	522	524													
TMP6	COF4	116																			
T2T2	7387	707	425	428																	
T2T21	7389	708	715																		
LINT	FFFF	95	150	228																	
WB	7238	481	433																		
WBF1	725A	498	505																		
WBF2	7262	503	496	500																	
WBAPF	724E	450	512																		
WB1	723C	482	515																		
WFC	71E2	435	477																		
WF1	72C7	455	449	452																	
WH2	721F	465	472																		
WC	71C2	421	322																		
WRAP	00E4	105	232	435	482	727															
WROA	729A	535	355	488																	
WRCA1	72AE	546	540	542	544																
WRCA4	729E	541	474																		
WROA6	72A2	543																			
WRCE	72B1	555	341	457	460	463	469	545													
WRCC	72C6	578	235	259	445	503	507														
WRPC	72AE	545	330																		
WRT	72C6	576	570	578	699																
WRTWO	72CC	568	223	530																	
WRT1	72CE	582	586																		
XR	COFA	121	206	377																	
YR	CCFB	122	207	378																	
ZTMP	70C9	278	393	442																	

INSTRUCTION	COUNT
ADC	9
AND	9
ASL	6
HCC	15
RCS	6
REC	11
RIT	0
RMI	0
RNE	33
RPL	5
RRK	1
BVC	0
RVS	0
CLC	4
CLD	1
CLI	1
CLV	0
CMP	11
CPX	1
CPY	0
DEC	6
DEX	8
DEY	1
ECR	4
INC	7
INX	0
INY	2
JMP	9
JSR	89
LCA	65
LDX	24
LDY	4
LSR	13
NCP	0
CRA	6
PFA	18
PHP	4
PLA	23
PLP	4
RCL	0
RTI	1
RTS	19
SPC	2
SFC	3
SEC	0
SEI	0
STA	45
STX	11
STY	2
TAX	4
TAY	4
TSX	1
TXA	5
TXS	2
TYA	5



Microelectronic Device Division

REGIONAL SALES OFFICES

WESTERN REGION, U.S.A.

3310 Miraloma Avenue
P.O. Box 3669
Anaheim, Ca. 92803
Phone: (714) 632-3698

EASTERN REGION, U.S.A.

Carolier Office Building
850-870 U.S. Route 1
North Brunswick, New Jersey 08902
Phone: (201) 246-3630

MIDWEST REGION, U.S.A.

Contact:
John G. Twist Company
1301 Higgins Road
Elk Grove Village, Illinois 60007
Phone: (312) 593-0200

CENTRAL REGION, U.S.A.

2855 Coolidge Road, Suite 101
Troy, Michigan 48064
Phone: (313) 435-1638

FAR EAST

Rockwell International Overseas Corp.
Ichiban-cho Central Building
22-1 Ichiban-cho, Chiyoda-ku
Tokyo 102, Japan
Phone: 265-8808

EUROPE

Rockwell International GmbH
Microelectronic Device Division
Fraunhoferstrasse 11
D-8033 Munchen-Martinsried
Germany
Phone: (089) 859-9575



Rockwell International

CFECKOUT PROGRAM -- PRINT BINARY OF TYPED CHARACTER

```
CCCC          * = C          ;VARIABLE STGRAGE IN PAGE ZERO
0000          BINARY * = 3+1 ;STGRAGE FOR CHAR DURING DISSECTION
0001          CCUNT  * = 3+1 ;COUNT OF BITS REMAINING TO PRINT
;
0002          * = $0100      ;PROGRAM BEGINS ON PAGE ONE
;
CRLF          = $728A        ;TIM  CRLF ROUTINE
WRT           = $72C6        ;TIM  WRITE ROUTINE
RDT           = $72E9        ;TIM  READ ROUTINE
SPACE        = $7377        ;TIM  SPACE ROUTINE
;
C100  20 8A 72  PBIN  JSR CRLF      ;PRINT CARRIAGE RETURN & LINE FEED
      103 20 E9 72 JSR RDT         ;GET A CHARACTER
      1C6 85 CC          STA BINARY  ;SAVE FOR DISSECTION
      1C8 20 77 73 JSR SPACE      ;PRINT A SPACE
;
01CB  A9 C8          LDA #B        ;INITIALIZE BIT COUNT
01CC  85 01          STA COUNT
;
01CF  A9 3C          PBLCCP LDA #'0  ;ASSUME ZERO: LOAD ASCII "0"
0111  C6 C0          ASL BINARY    ;C=NEXT BIT
0113  EC 02          BCS PRINT     ;PRINT ZERO
;
0115  A9 31          LDA #'1      ;LOAD ASCII "1"
;
C117  2C C6 72      PRINT JSR WRT   ;PRINT BINARY DIGIT
C11A  C6 01          DEC CCUNT     ;COUNT BIT PRINTED
C11C  1C F1          BPL PBLCCP    ;GO NEXT BIT
;
011E  4C C0 01      JMP PBIN      ;DO IT ALL AGAIN
```

```

;CHECKOUT PROGRAM -- PRINT BINARY CF TYPED CHARACTER
;
;
CCCC      * = 0      ;VARIABLE STORAGE IN PAGE ZERO
CCGG      BINARY * = * + 1 ;STORAGE FOR CHAR DURING DISSECTION
CC01      CCUNT * = * + 1 ;COUNT OF BITS REMAINING TO PRINT
;
0002      * = $0100 ;PROGRAM BEGINS ON PAGE ONE
;
CRLF      = $728A    ;TIM CRLF ROUTINE
WRT       = $72C6    ;TIM WRITE ROUTINE
RDT       = $72E9    ;TIM READ ROUTINE
SPACE     = $7277    ;TIM SPACE ROUTINE
;
CC 2C 8A 72 PBIN JSR CRLF ;PRINT CARRIAGE RETURN & LINE FEED
C103 20 E9 72 JSR RDT ;GET A CHARACTER
0106 85 00 STA BINARY ;SAVE FOR DISSECTION
C108 2C 77 72 JSR SPACE ;PRINT A SPACE
;
C108 A9 C8 LCA #8 ;INITIALIZE BIT COUNT
C10C 85 01 STA CCUNT
;
010F A9 30 PBLCCP LCA #0 ;ASSUME ZERO: LOAD ASCII "0"
0111 C6 C0 ASL BINARY ;C=NEXT BIT
0113 90 C2 BCC PRINT ;PRINT ZERO
;
0115 A9 31 LCA #1 ;LOAD ASCII "1"
;
C117 2C C6 72 PRINT JSR WRT ;PRINT BINARY DIGIT
C11A C6 01 DEC CCUNT ;COUNT BIT PRINTED
C11C 0C F1 BNE PBLCCP ;GO NEXT BIT
;
011E 4C C0 01 JMP PBIN ;DO IT ALL AGAIN

```

CORRECTED PBIN PROGRAM