

00002 ; This program is designed to be relocatable and  
00003 uses memory locations M(0000) thru M(0007) for  
00004 temporary variable storage.  
00005  
00006 ; This program is entered from a CALL "EXEC", A\$  
00007 statement in Basic after printing the memory ad-  
00008 dress labels for one 256 byte page. This program  
00009 scans one page of memory and displays each byte  
00010 as two hex characters and then returns to Basic.  
00011  
00012 ; Defined memory locations and CPU registers:  
00013 M(0000)=low nibble of displayed byte  
00014 M(0001 and 0002)=start of display chr data  
00015 M(0003 and 0004)=address of individual chr  
00016 M(0005)=temporary hi X D/A byte storage  
00017 M(0006 and 0007)=old stack ptr address  
00018 M(8794 and 8796)=X PIA D/A (hi and lo)  
00019 M(878C and 878E)=Y PIA D/A (hi and lo)  
00020 and also X&Y chr D/A  
00021 A and B are working registers  
00022 X=address of individual character column  
00023 SP=address of displayed memory location  
00024  
00025 02B5 > ORG 02B5H ;SET ORIGIN TO 02B5 HEX  
00026 02B5 0F INIT SEI ;DISABLE INTERRUPTS  
00027 02B6 9600 LDA A 00H ;GET M(0000) THRU M(0007)  
00028 02B8 36 PSH A ;SAVE ON BASIC PROGRAM STACK  
00029 02B9 9601 LDA A 01H  
00030 02BB 36 PSH A  
00031 02BC 9602 LDA A 02H  
00032 02BE 36 PSH A  
00033 02BF 9603 LDA A 03H  
00034 02C1 36 PSH A

00035	0202	9604	LDA	A	04H	
00036	0204	36	PSH	A		
00037	0205	9605	LDA	A	05H	
00038	0207	36	PSH	A		
00039	0208	9606	LDA	A	06H	
00040	020A	36	PSH	A		
00041	020B	9607	LDA	A	07H	
00042	020D	36	PSH	A		
00043	02CE	9F06	STS	06H		;SAVE SP AT M(0006 AND 0007)
00044	02D0	B68794	LDA	A	8794H	;LOAD AND SAVE X D/A PIA
00045	02D3	8403	AND	A	#03H	;HIGH BYTE AT M(0005)
00046	02D5	8B70	ADD	A	#70H	;AFTER DISABLING THE
00047	02D7	9705	STA	A	05H	VECTOR BITS
00048	02D9	B78794	STA	A	8794H	;ALSO RESTORE AT X PIA
00049	02DC	8E0002	LDS	#	0002H	;INITIALIZE STACK POINTER
00050	02DF	8639	LDA	A	#39H	;LOAD A WITH RTS INSTRUCTION
00051	02E1	9700	STA	A	00H	;STORE RTS AT M(0000)
00052	02E3	BD0000	JSR	00H		;JUMP TO M(0000) THEN CON-
00053						TINUE BUT LEAVE PC AT
00054						M(0001 AND 0002)

Tektronix      M6800 ASM V3.3    TEK 4051 MEMORY DUMP      Page      2

00055	02E6	86D4	LDA	A	#0D4H	;LOAD A WITH OFFSET TO INDIVIDUAL CHR DATA
00056						
00057	02E8	9B02	ADD	A	02H	;UPDATE M(0001) AND M(0002)
00058	02EA	9702	STA	A	02H	;TO POINT TO THE
00059	02EC	8600	LDA	A	#00H	CHARACTER
00060	02EE	9901	ADC	A	01H	DATA STACK
00061	02F0	9701	STA	A	01H	AND STORE
00062	02F2	8E7FFF	LDS	#	7FFFH	;LOAD SP WITH 1ST BYTE-1 TO BE DISPLAYED: MEMORY > SP
00063						
00064	02F5	33	LOOP	PUL	B	;M(SP+1) TO B, SP+1 TO SP
00065	02F6	17		TBA		;ALSO STORE BYTE IN A
00066	02F7	44		LSR	A	;MASK LOW NIBBLE OF MEMORY
00067	02F8	44		LSR	A	BYTE TO BE DISPLAYED AND

00068	02F9	44		LSR	A	LEAVE HIGH NIBBLE IN THE
00069	02FA	44		LSR	A	4 LSBITS OF A
00070	02FB	C40F		AND	B #0FH	MASK OUT HIGH NIBBLE
00071	02FD	CB80		ADD	B #80H	PUT NIBBLE COUNT IN BYTE
00072	02FF	D700		STA	B 00H	STORE LOW NIBBLE AT M(0000)
00073	0301	16	CALC	TAB		MOVE NIBBLE TO B
00074	0302	48		ASL	A	MULTIPLY NIBBLE BY 4
00075	0303	48		ASL	A	WITH 2 LEFT SHIFTS
00076	0304	1B		ABA		A+B>A=5 TIMES NIBBLE
00077	0305	9802		ADD	A 02H	ADD OFFSET TO CHR
00078						DATA LOW BYTE
00079	0307	9704		STA	A 04H	STORE LOW CHR DATA
00080						POINTER
00081	0309	8600		LDA	A #00H	ADD CARRY TO HIGH
00082	030B	9901		ADC	A 01H	CHR DATA POINTER
00083	030D	9703		STA	A 03H	STORE HIGH CHR PTR
00084	030F	DE03		LDX	03H	LOAD X WITH CHR PTR
00085	0311	8603		LDA	A #03H	RESET A FOR MASKING
00086	0313	B48780		AND	A 8780H	MASK ALL BUT 2 LSB OF
00087						Y D/A PIA A SIDE
00088	0316	B78780		STA	A 8780H	STORE THE RESET CHR
00089						D/A VALUES
00090	0319	A600	DISPLA	LDA	A X	LOAD CHARACTER COLUMN
00091	031B	08		INX		POINT TO NEXT COLUMN
00092	031C	46		ROR	A	ROLL CHR COLUMN RIGHT 1 BIT
00093	031D	16		TAB		STORE SHIFTED CHR COLUMN
00094	031E	2403		BCC	ADOT+3	IF CARRY CLEAR SKIP 1 STEP
00095	0320	B68794	ADOT	LDA	A 8794H	PRINT A DOT, READ X D/A
00096	0323	8604		LDA	A #04H	LOAD A WITH Y D/A INCREMENT
00097	0325	BB8780		ADD	A 8780H	ADD A TO Y D/A
00098	0328	B78780		STA	A 8780H	AND STORE IN Y D/A PIA
00099	032B	841C		AND	A #1CH	MASK X CHR & Y VECTOR BITS
00100	032D	8100		CMP	A #00H	TEST IF Y CHR BITS=0
00101	032F	2703		BEQ	COLUMN	IF EQUAL SET UP NEXT COLUMN
00102	0331	17		TBA		OTHERWISE GET NEXT BIT

00103	0332	20E8		BRA	DISPLA+3	
00104	0334	86FC	COLUMN	LDA	A #0FCH	;LOAD A WITH Y VECTOR MASK
00105	0336	B4878C		AND	A 878CH	;MASK Y VECTOR D/A BITS
00106	0339	81A0		CMP	A #0A0H	;TEST IF CHARACTER COMPLETE
00107	033B	26DC		BNE	DISPLA	;IF NOT DONE CONTINUE CHR
Tektronix	M6800	ASM V3.3	TEK	4051	MEMORY DUMP	Page 3
00108	033D	9600	NIBBLE	LDA	A 00H	;CHECK WHICH NIBBLE IS DONE
00109	033F	8100		CMP	A #00H	;BYTE COMPLETE IF A=0
00110	0341	2706		BEQ	SPACES	;IF BYTE COMPLETE SKIP 2 SPA
00111	0343	860E		LDA	A #14	;PRESET CHR SPACE AS 14 BITS
00112	0345	2004		BRA	SPACES+2	;OTHERWISE SKIP ONE SPACE
00113	0347	20AC	DUMMY	BRA	LOOP	;ALLOWS LATER BRA TO LOOP
00114	0349	862A	SPACES	LDA	A #42	;PRESET CHR SPACE AS 42 BITS
00115	034B	BB8796		ADD	A 8796H	;ADD TO X D/A LOW BYTE
00116	034E	B78796		STA	A 8796H	;AND RESTORE
00117	0351	8600		LDA	A #00H	;AND CARRY RESULT
00118	0353	9905		ADC	A 05H	;TO HIGH BYTE
00119	0355	9705		STA	A 05H	;STORE AT M(0005)
00120	0357	B78794		STA	A 8794H	;AND RESTORE AT D/A
00121	035A	B68796		LDA	A 8796H	;LOAD NEW X AND Y
00122	035D	B78796		STA	A 8796H	;VECTOR
00123	0360	9600		LDA	A 00H	;CHECK WHICH NIBBLE IS DONE
00124	0362	8100		CMP	A #00H	;BYTE COMPLETE IF A=0
00125	0364	2709		BEQ	LINE	;CHECK LINE IF A=0
00126	0366	9600		LDA	A 00H	;LOAD LOW NIBBLE INTO A
00127	0368	840F		AND	A #0FH	;CLEAR NIBBLE COUNTER
00128	036A	7F0000		CLR	00H	;CLEAR M(0000)
00129	036D	2092		BRA	CALC	;CALCULATE CHR OFFSET
00130	036F	9F03	LINE	STS	03H	;STORE STACK PTR FOR TEST
00131	0371	860F		LDA	A #0FH	;LOAD A WITH LINE COMPARISON
00132	0373	9404		AND	A 04H	;COMPARE A AND LSBYTE OF STK
00133	0375	810F		CMP	A #0FH	;WITH 0, IF A=0, NEWLINE
00134	0377	26CE		BNE	DUMMY	;IF NOT, DISPLAY NEXT BYTE
00135	0379	86FF	PAGE1	LDA	A #0FFFH	;LOAD A WITH PAGE COMPARISON

00136	037B	9404		AND	A 04H	;COMPARE A AND LSBYTE OF STK
00137	037D	81FF		CMP	A #0FFH	;WITH END OF PAGE
00138	037F	271E		BEQ	RETURN	;IF NOT, SET NEW LINE
00139	0381	B6878E	NEWLIN	LDA	A 878EH	;LOAD A WITH Y D/A LOW BYTE
00140	0384	8016		SUB	A #22	;SUBTRACT 22 BITS (ONE LINE)
00141	0386	B7878E		STA	A 878EH	;RESTORE Y D/A LOW BYTE
00142	0389	B6878C		LDA	A 878CH	;LOAD A WITH Y D/A HIGH BYTE
00143	038C	8200		SBC	A #00H	;SUBTRACT CARRY
00144	038E	B7878C		STA	A 878CH	;RESTORE Y D/A HIGH BYTE
00145	0391	8670		LDA	A #70H	;LOAD A WITH NEW HIGH X BYTE
00146	0393	9705		STA	A 05H	;STORE AT TEMPORARY BYTE
00147	0395	B78794		STA	A 8794H	;STORE NEW HI BYTE IN D/A
00148	0398	8662		LDA	A #98	;LOAD A WITH NEW LOW X BYTE
00149	039A	B78796		STA	A 8796H	;STORE AND UPDATE X&Y VECTOR
00150	039D	20A8		BRA	DUMMY	;GET NEXT BYTE AND DISPLAY
00151	039F	9E06	RETURN	LDS	06H	;LOAD OLD STACK POINTER
00152	03A1	32		PUL	A	;RESTORE PREVIOUS CONTENTS
00153	03A2	9707		STA	A 07H	;OF M(0000) TO M(0007)
00154	03A4	32		PUL	A	
00155	03A5	9706		STA	A 06H	
00156	03A7	32		PUL	A	
00157	03A8	9705		STA	A 05H	
00158	03AA	32		PUL	A	
00159	03AB	9704		STA	A 04H	
00160	03AD	32		PUL	A	

Tektronix M6800 ASM V3.3 TEK 4051 MEMORY DUMP

Page 4

00161	03AE	9703		STA	A 03H	
00162	03B0	32		PUL	A	
00163	03B1	9702		STA	A 02H	
00164	03B3	32		PUL	A	
00165	03B4	9701		STA	A 01H	
00166	03B6	32		PUL	A	
00167	03B7	9700		STA	A 00H	
00168	03B9	39		RTS		;THEN RETURN TO BASIC PGM

00169	03BA	7C8A	DATA	BYTE	7CH,8AH ;BEGIN CHR DATA, 0 DATA
00170	03BC	92A2		BYTE	92H,0A2H
00171	03BE	7C		BYTE	7CH
00172	03BF	0042		BYTE	00H,42H ;CHR 1 DATA
00173	03C1	FE02		BYTE	0FEH,02H
00174	03C3	00		BYTE	00H
00175	03C4	468A		BYTE	46H,8AH ;CHR 2 DATA
00176	03C6	9292		BYTE	92H,92H
00177	03C8	62		BYTE	62H
00178	03C9	8482		BYTE	84H,82H ;CHR 3 DATA
00179	03CB	92B2		BYTE	92H,0B2H
00180	03CD	DC		BYTE	0DCH
00181	03CE	1828		BYTE	18H,28H ;CHR 4 DATA
00182	03D0	48FE		BYTE	48H,0FEH
00183	03D2	08		BYTE	08H
00184	03D3	E4		BYTE	0E4H ;CHR 5 DATA
00185	03D4	A2A2		BYTE	0A2H,0A2H
00186	03D6	A290		BYTE	0A2H,9CH
00187	03D8	3C52		BYTE	3CH,52H ;CHR 6 DATA
00188	03DA	9292		BYTE	92H,92H
00189	03DC	8C		BYTE	8CH
00190	03DD	808E		BYTE	80H,8EH ;CHR 7 DATA
00191	03DF	90A0		BYTE	90H,0A0H
00192	03E1	C0		BYTE	0C0H
00193	03E2	6C92		BYTE	6CH,92H ;CHR 8 DATA
00194	03E4	9292		BYTE	92H,92H
00195	03E6	6C		BYTE	6CH
00196	03E7	6292		BYTE	62H,92H ;CHR 9 DATA
00197	03E9	9294		BYTE	92H,94H
00198	03EB	78		BYTE	78H
00199	03EC	3E48		BYTE	3EH,48H ;CHR A DATA
00200	03EE	8848		BYTE	88H,48H
00201	03F0	3E		BYTE	3EH
00202	03F1	FE92		BYTE	0FEH,92H;CHR B DATA
00203	03F3	9292		BYTE	92H,92H

00204	03F5	6C	BYTE	6CH
00205	03F6	7C82	BYTE	7CH,82H ;CHR C DATA
00206	03F8	8282	BYTE	82H,82H
00207	03FA	44	BYTE	44H
00208	03FB	FE82	BYTE	0FEH,82H;CHR D DATA
00209	03FD	8244	BYTE	82H,44H
00210	03FF	38	BYTE	38H
00211	0400	FE92	BYTE	0FEH,92H;CHR E DATA
00212	0402	9292	BYTE	92H,92H
00213	0404	82	BYTE	82H

Tektronix M6800 ASM V3.3 TEK 4051 MEMORY DUMP Page 5

00214	0405	FE90	BYTE	0FEH,90H;CHR F DATA
00215	0407	9090	BYTE	90H,90H
00216	0409	80	BYTE	80H

00217  
Tektronix M6800 ASM V3.3 Symbol Table Page 6

#### %DISOBJ (default) Section (040A)

ADOT --- 0320	CALC --- 0301	COLUMN - 0334
DATA --- 03BA	DISPLA - 0319	
DUMMY -- 0347	INIT --- 02B5	LINE --- 036F
LOOP --- 02F5	NEWLIN - 0381	
NIBBLE - 033D	PAGE1 -- 0379	RETURN - 039F
SPACES - 0349		

217 Source Lines 217 Assembled Lines 797 Bytes available

>>> No assembly errors detected <<<

\*COPY\* EOJ