

CALENDAR OF EVENTS

4308

TRS-80 USERS GROUP :  
 FRIDAY, AUGUST 5TH, 8:00 PM,  
 SOUTHWESTERN PUBLIC SERVICE CENTER,  
 4201 NORTH FRANKFORD AVENUE

APPLE USERS GROUP :  
 FRIDAY, AUGUST 5TH, 8:00 PM,  
 SOUTHWESTERN PUBLIC SERVICE CENTER,  
 4201 NORTH FRANKFORD AVENUE

GENERAL CLUB MEETING :  
 FRIDAY, AUGUST 19TH, 8:00 PM,  
 SOUTHWESTERN PUBLIC SERVICE CENTER,  
 4201 FRANKFORD AVENUE

TI 99/4 USERS GROUP :  
 SATURDAY, AUGUST 20TH, 2:00 PM,  
 UNISOURCE ELECTRONICS,  
 UNIVERSITY SHOPPING CENTER

WHO TO CONTACT :

DEAN POWELL	CLUB PRESIDENT	794-5260
BRETT PIJAN	VICE PRESIDENT	765-0102
JOHN CRAWFORD	SEC./TREAS.	745-5604
KELLY HAYS	TRS-80 USERS	763-6579
ROBERT DARKO	APPLE USERS	765-6630
SAM WELLER	TI 99/4 USERS	793-8367
MAX MARTIN	COMMODORE USERS	747-6653
JIM SWAN	IBM PC USERS	792-3448

WEST TEXAS TENNIS!!



THE TI 99/4 USERS' GROUP

The TI 99/4 Users' Group held their July meeting on Saturday, 16 July, in a classroom of the Administration Building of the Lubbock Christian College from 2 PM to 4:30 PM on a very warm day. Only a handful of people showed up on this this hot, humid day; many apparently being on vacation. Again Dean Powell and Sam Weller had the session organized. It was announced that the next meeting of the Group would be held on Saturday, 20 Aug 83, at 2 PM at the Unisource Store, located at University and South Loop inside the Loop on the NW corner shopping center next to the Seafood Shop. The new meeting place has many fully-equipped operating positions.

Sam presented a good hour's instruction on "files". More will be presented at future meetings. New books being written on TI equipment and programming were displayed. Titles and authors will be listed in this paper next time, but many are available at Unisource or Agriplex at this time.

Spread the word about the new meeting place and let's all get out for a bigger and better group interaction!

```

1000 REM      HAPPY BIRTHDAY TO JONATHAN
1010 REM      BY BOB McCLUSKEY LUBBOCK, TEXAS
1020 REM      REQUIRES TERMINAL EMULATOR II AND SPEECH SYNTHESIZER
1030 CALL CLEAR
1040 OPEN #1:"SPEECH",OUTPUT
1050 CALL SOUND(-900,147,10,294,15)
1060 TONE$="//47 0"
1070 WORD$="HAP"
1080 GOSUB 2070
1090 CALL SOUND(-900,147,10,294,15)
1100 TONE$="//47 0"
1110 WORD$="P"
1120 GOSUB 2070
1130 CALL SOUND(-900,147,10,330,15)
1140 TONE$="//44 0"
1150 WORD$="BIRTH"
1160 GOSUB 2070
1170 CALL SOUND(-900,147,10,294,15)
1180 TONE$="//47 0"
1190 WORD$="DAY"
1200 GOSUB 2070
1210 CALL SOUND(-900,196,10,392,15)
1220 TONE$="//40 0"
1230 WORD$="2"
1240 GOSUB 2070
1250 CALL SOUND(-900,185,10,370,15)
1260 TONE$="//42 0"
1270 WORD$="U"
1280 GOSUB 2070
1290 CALL SOUND(-900,147,10,294,15)
1300 TONE$="//47 0"
1310 WORD$="HAP"
1320 GOSUB 2070
1330 CALL SOUND(-900,147,10,294,15)
1340 TONE$="//47 0"
1350 WORD$="P"
1360 GOSUB 2070
1370 CALL SOUND(-900,165,10,330,15)
1380 TONE$="//47 0"
1390 WORD$="BIRTH"
1400 GOSUB 2070
1410 CALL SOUND(-900,147,10,294,15)
1420 TONE$="//44 0"
1430 WORD$="DAY"
1440 GOSUB 2070
1450 CALL SOUND(-900,220,10,440,15)
1460 TONE$="//38 0"
1470 WORD$="2"
1480 GOSUB 2070
1490 CALL SOUND(-900,196,10,392,15)
1500 TONE$="//40 0"
1510 WORD$="U"
1520 GOSUB 2070
1530 CALL SOUND(-900,147,10,294,15)
1540 TONE$="//47 0"
1550 WORD$="HAP"
1560 GOSUB 2070
1570 CALL SOUND(-900,147,10,294,15)
1580 TONE$="//47 0"
1590 WORD$="P"
1600 GOSUB 2070
1610 CALL SOUND(-900,294,10,588,15)
1620 TONE$="//30 0"
1630 WORD$="BIRTH"
1640 GOSUB 2070
1650 CALL SOUND(-900,247,10,494,15)
1660 TONE$="//35 0"
1670 WORD$="DAY"
1680 GOSUB 2070
1690 CALL SOUND(-900,196,10,392,15)
1700 TONE$="//40 0"
1710 WORD$="DEER"
1720 GOSUB 2070
1730 CALL SOUND(-900,185,10,370,15)
1740 TONE$="//41 0"
1750 WORD$="JON"
1760 GOSUB 2070
1770 CALL SOUND(-900,165,10,330,15)
1780 TONE$="//44 0"
1790 WORD$="NUTHIN"
1800 GOSUB 2070
1810 CALL SOUND(-900,262,10,524,15)
1820 TONE$="//34 0"
1830 WORD$="HAP"
1840 GOSUB 2070
1850 CALL SOUND(-900,262,10,524,15)
1860 TONE$="//34 0"
1870 WORD$="P"
1880 GOSUB 2070
1890 CALL SOUND(-900,247,10,494,15)
1900 TONE$="//35 0"
1910 WORD$="BIRTH"
1920 GOSUB 2070
1930 CALL SOUND(-900,196,10,392,15)
1940 TONE$="//40 0"
1950 WORD$="DAY"
1960 GOSUB 2070
1970 CALL SOUND(-900,220,10,440,15)
1980 TONE$="//38 0"
1990 WORD$="2"
2000 GOSUB 2070
2010 CALL SOUND(-900,196,10,392,15)
2020 TONE$="//40 0"
2030 WORD$="U"
2040 GOSUB 2070
2050 CLOSE #1
2060 END
2070 PRINT #1:TONE$
2080 PRINT #1:WORD$
2090 RETURN

```

```

0000      00100      TITLE      '<Warm Boot Utility>'
00110 ;
00120 ; This program creates a configuration program on any
00130 ; drive.  When the WARMBOOT/CMD program is executed,
00140 ; all current device and driver configuration will be
00150 ; stored in the FILE.  Each time the FILE is executed
00160 ; the configuration will be stored and set.  The following
00170 ; items will be saved:
00180 ;
00190 ;      1.  All items referanced by the SYSTEM command
00200 ;          in LDOS.
00210 ;      2.  Most FILTERing, LINKing, ROUTEing, and
00220 ;          SETting that has been done.
00230 ;      3.  Any active background tasks. (TRACE, SPOOL,
00240 ;          or DEBUG)
00250 ;      4.  All memory from HIGH$ to the physical top
00260 ;          of memory will be written to the FILE
00270 ;          specified by the user.
00280 ;
00290 ; NOTE:  Questions and comments should be sent to:
00300 ;          J. Kelly Hays
00310 ;          Rt. 1 Box 61
00320 ;          Lubbock, TX 79401
00330 ;          (806) 763-6579
00340 ;          Compuserve # [70260,244]
00350 ;
00360 ; Parts of WARMBOOT/CMD are based on Les Mikesell's
00370 ; program SYSGEN/CMD 11/14/81 found on the LDOS board
00380 ; on Compuserve.
00390 ;
00400 ;
00410 ; |-----|
00420 ; |  WARMBOOT (QUERY,DRIVE=d,LABLE="diskname",
00430 ; |          AUTO="command",FILE="filespec",SCREEN)
00440 ; |
00450 ; |  QUERY          Ignore all other parameters except
00460 ; |                  SCREEN and prompt user for information.
00470 ; |
00480 ; |  DRIVE=d       Select drive to be checked for correct
00490 ; |                  lable.  If DRIVE is found alone or d is
00500 ; |                  greater than 7 then the user will be
00510 ; |                  prompted for a valid entry.
00520 ; |
00530 ; |  LABLE="diskname"  When booting check the diskname
00540 ; |                    on DRIVE.  If no match to "diskname"
00550 ; |                    then prompt user for correct disk.
00560 ; |                    If LABLE is found alone on command line
00570 ; |                    the default diskname will be listed and
00580 ; |                    the user will be prompted for input.
00590 ; |                    Note:  "diskname" should 1 to 8 char
00600 ; |                    in length and NO UPPER/lower case
00610 ; |                    conversion will be done.  Also when
00620 ; |                    prompted for input UPPER case will
00630 ; |                    be forced.
00640 ; |
00650 ; |  AUTO="command"  This works very much like the auto

```

```

00660 ;| library command in LDOS. If AUTO is
00670 ;| found alone on the line the user will
00680 ;| be prompted for input. NOTE Use AUTO
00690 ;| alone on command line if your command
00700 ;| contains any of these char. " , )
00710 ;|
00720 ;| FILE="filespec" This allows the user to enter the
00730 ;| the boot routine file name on the
00740 ;| command line. The default extension
00750 ;| is /CMD. If FILE is not found the user
00760 ;| will be prompted for the filespec.
00770 ;|
00780 ;| SCREEN This tells warmboot to save the screen.
00790 ;|
00800 ;| abbr: QUERY=Q, DRIVE=D, LABEL=L, AUTO=A, FILE=F,
00810 ;| and SCREEN=S.
00820 ;|
00830 ;|
00840 ;|

```

```

0000 00850 SUBTTL '<Define Lables and Macros>'

```

## Define Lables and Macros

```

0000 00860 PAGE
001B 00870 @PUT EQU 001BH
0040 00880 @KEYIN EQU 0040H
00E0 00890 INTMASK EQU 00E0H
00FE 00900 PORTFE EQU 00FEH
3C00 00910 VIDMEM EQU 3C00H
4209 00920 @CKDRV EQU 4209H
4488 00930 @CKDRV1 EQU 4488H
402D 00940 @EXIT EQU 402DH
4030 00950 @ABORT EQU 4030H
4049 00960 HIGH1$ EQU 4049H
4213 00970 UNKNWNA EQU 4213H
421D 00980 @ICNFG EQU 421DH
4303 00990 @ICNFG1 EQU 4303H
42A0 01000 UNKNWNB EQU 42A0H
4409 01010 @ERROR EQU 4409H
4411 01020 HIGH$ EQU 4411H
441C 01030 @FSPEC EQU 441CH
4420 01040 @INIT EQU 4420H
4428 01050 @CLOSE EQU 4428H
442B 01060 SFLAG$ EQU 442BH
430F 01070 SFLAG1$ EQU 430FH
429F 01080 KFLAG$ EQU 429FH
4423 01090 KFLAG1$ EQU 4423H
444B 01100 @FEXT EQU 444BH
4454 01110 @PARAM EQU 4454H
4467 01120 @DSPLY EQU 4467H
4473 01130 @FEXT1 EQU 4473H
4476 01140 @PARA1 EQU 4476H
FFFF 01150 TRUE EQU -1
4299 01160 @CMDI EQU 4299H
4419 01170 @DODIR EQU 4419H
0049 01180 @KEY EQU 0049H
4463 01190 @DODIR1 EQU 4463H
01200 ;

```

```

0000      01210 *GET AUTOIND
00010 ;AUTOIND/ASM : Automatic Mod I/III independence
00020 ;
00030 *LIST OFF
00050 *LIST ON
01220 ;
01230 ;Similate the instruction LD (#ADDR),TRUE
0000      01240 HL_TRUE MACRO #ADDR
0000      01250 LD HL,#ADDR
0000      01260 LD (HL),TRUE
0000      01270 INC HL
0000      01280 LD (HL),TRUE
0000      01290 ENDM
0000      01300 SUBTTL '<Boot time Module>'
Boot time Module

0000      01310 PAGE
5200      01320 ORG 5200H
5200 3AE952 01330 START LD A,(DSKLBL) ;Is there a lable to
5203 FE20 01340 CP 20H ;check for.
5205 284C 01350 JR Z,BEGIN ;NO, forget it.
01360 ;Check lable if match go on if no match then prompt user
5207 010000 01370 DRVLBL LD BC,0 ;Load C with drive #
520A CD0942 01380 CALL @CKDRV ;Is drive ready?
520D 01390 MOD_1 @CKDRV1
520D 201A 01400 JR NZ,NOMATCH ;NO, skip @DODIR.
520F 0604 01410 LD B,4 ;Tell @dodir name only
5211 21F252 01420 LD HL,DSKLBL1
5214 CD1944 01430 CALL @DODIR ;Get lable from disk.
5217 01440 MOD_1 @DODIR1
5217 21F252 01450 LD HL,DSKLBL1 ;Compare lables
521A 0608 01460 LD B,8
521C 11E952 01470 LD DE,DSKLBL
521F 1A 01480 LOOP2 LD A,(DE)
5220 BE 01490 CP (HL)
5221 2006 01500 JR NZ,NOMATCH ;Go prompt user for disk.
5223 13 01510 INC DE
5224 23 01520 INC HL
5225 10F8 01530 DJNZ LOOP2
5227 182A 01540 JR BEGIN ;Good they match continue
5229 ED4B0852 01550 NOMATCH LD BC,(DRVLBL+1) ;Prompt user with
522D 79 01560 LD A,C
522E C630 01570 ADD A,30H
5230 323D53 01580 LD (INSERT1+8),A
5233 210653 01590 LD HL,INSERT
5236 CD6744 01600 CALL @DSPLY ;correct Drive #,
5239 21E952 01610 LD HL,DSKLBL
523C CD6744 01620 CALL @DSPLY ;and lable.
523F 213553 01630 LD HL,INSERT1
5242 CD6744 01640 CALL @DSPLY
5245 CD4900 01650 KEY CALL @KEY ;Wait for input.
5248 FE01 01660 CP 01H ;Is it <BREAK> ?
524A CA2D40 01670 JP Z,@EXIT ;YES, then END.
524D FE0D 01680 CP 0DH ;Is it <ENTER> ?
524F 28B6 01690 JR Z,DRVLBL ;YES, then try again.
5251 18F2 01700 JR KEY ;NO, wait some more
5253 F3 01710 BEGIN DI
5254 3E04 01720 LD A,00000100B ;Enable Real time clock
5256 D3E0 01730 OUT (INTMASK),A ;int. No effect on mod I.
01740 ;

```

```

01750 ;The following routine reconfigures memory and pointers
01760 ;according to the data that is found following the program
01770 ;
5258 215A53 01780 LD HL,ENTRY ;End of boot module+1
525B 4E 01790 LOOP0 LD C,(HL) ;# of bytes to move in BC
525C 23 01800 INC HL
525D 46 01810 LD B,(HL)
525E 23 01820 INC HL
525F 79 01830 LD A,C ;Zero bytes?
5260 B0 01840 OR B
5261 2805 01850 JR Z,HIGHMEM ;YES, Check for high mem
5263 CD8052 01860 CALL MOVIT ;NO, Continue
5266 18F3 01870 JR LOOP0
5268 4E 01880 HIGHMEM LD C,(HL) ;# of bytes in BC
5269 23 01890 INC HL
526A 46 01900 LD B,(HL)
526B 23 01910 INC HL
526C 79 01920 LD A,C ;Zero bytes?
526D B0 01930 OR B
526E C47352 01940 CALL NZ,MOVHIGH ;NO, Move high mem.
5271 1814 01950 JR INTCNFG ;YES, Initialize config.
5273 5E 01960 MOVHIGH LD E,(HL)
5274 23 01970 INC HL
5275 56 01980 LD D,(HL)
5276 23 01990 INC HL
5277 EB 02000 EX DE,HL ;Calculate addresses
5278 09 02010 ADD HL,BC
5279 EB 02020 EX DE,HL
527A 09 02030 ADD HL,BC
527B 2B 02040 DEC HL
527C 1B 02050 DEC DE
527D EDB8 02060 LDDR ;Move it.
527F C9 02070 RET
5280 5E 02080 MOVIT LD E,(HL) ;Destination in DE.
5281 23 02090 INC HL
5282 56 02100 LD D,(HL)
5283 23 02110 INC HL ;Source in HL.
5284 EDB0 02120 LDIR
5286 C9 02130 RET
5287 CD1D42 02140 INTCNFG CALL @ICNFG ;Initalize configuration.
528A 02150 MOD_1 @ICNFG1
528A 3A1342 02160 LD A,(UNKNWNA)
528D D3E0 02170 OUT (INTMASK),A ;No effect in mod I
528F 212B44 02180 LD HL,SFLAG#
5292 02190 MOD_1 SFLAG1#
5292 CBAE 02200 RES 05H,(HL)
5294 CB5E 02210 BIT 03H,(HL)
5296 2807 02220 JR Z,DOAUTO
5298 3AA042 02230 LD A,(UNKNWNB)
529B 02240 MOD_1 ZERO
529B F601 02250 OR 01H
529D D3FE 02260 OUT (PORTFE),A
529F FB 02270 DOAUTO EI
52A0 21AC52 02280 LD HL,AUTOBUF ;Check for auto command
52A3 7E 02290 LD A,(HL)
52A4 FE0D 02300 CP 0DH ;Is there an auto?
52A6 CA2D40 02310 JP Z,@EXIT ;NO, end
52A9 C39942 02320 JP @CMDI ;YES, execute auto
52AC 0D 02330 AUTOBUF DM 0DH
52AD 20 02340 DC 60,20H
52E9 20 02350 DSKLBL DC 8,20H

```

```

52F1 0D      02360      DB      0DH
52F2 20      02370 DSKLBL1 DC      20,20H
5306 1C      02380 INSERT DM      1CH,1FH
5308 50      02390      DM      'Please insert the correct '
5322 64      02400      DM      'disk in the drive',0AH,03H
5335 44      02410 INSERT1 DM      'Drive # ',0AH,0AH
5340 50      02420      DM      'Press <enter> when ready.',0DH
535A        02430      SUBTTL  '<Create Boot Module>'
535A        02440      PAGE
535A E5      02450 ENTRY  PUSH   HL              ;Save command line.
535B        02460      CKMOD          ;Convert for MOD I if needed.
538C E1      02470      POP    HL              ;Get command line.
538D E5      02480      PUSH  HL              ;Save it again.
538E 2B      02490      DEC   HL
538F 23      02500 LOOP4  INC   HL
5390 7E      02510      LD    A,(HL)          ;Get char. from C. L.
5391 FE0D    02520      CP    0DH             ;Is it CR?
5393 280D    02530      JR    Z,HEADING      ;YES, No param. on C. L.
5395 FE2B    02540      CP    '('             ;Is it '(' ?
5397 20F6    02550      JR    NZ,LOOP4       ;NO, Look some more.
5399 11F655  02560      LD    DE,PTABLE      ;YES, Get Parameters.
539C CD5444  02570      CALL @PARAM
539F        02580      MOD_1 @PARA1
539F C27455  02590      JP    NZ,ERROR       ;Parameter Error.
53A2 212457  02600 HEADING LD    HL,MSG1        ;Put heading on screen.
53A5 CD6744  02610      CALL @DSPLY
53A8 210000  02620 QUERY  LD    HL,0            ;Does QUERY+1=TRUE ?
53AB 23      02630      INC  HL
53AC 7C      02640      LD   A,H
53AD B5      02650      OR   L
53AE 2020    02660      JR   NZ,DRIVE        ;NO, Continue.
53B0        02670      HL_TRUE FILE+1       ;YES, Set all parameters
53B8        02680      HL_TRUE AUTO+1       ; Equal to TRUE.
53C0        02690      HL_TRUE LABLE+1
53C8        02700      HL_TRUE DRVLBL+1
53D0 3A0852  02710 DRIVE  LD    A,(DRVLBL+1)
53D3 FE0B    02720      CP    0BH
53D5 D28555  02730      JP    NC,GETDRV
53D8 210000  02740 LABLE  LD    HL,0
53DB 7C      02750      LD   A,H
53DC B5      02760      OR   L
53DD 281D    02770      JR   Z,AUTO
53DF 23      02780      INC  HL
53E0 7C      02790      LD   A,H
53E1 B5      02800      OR   L
53E2 CC9655  02810      CALL Z,GETLBL
53E5 060B    02820      LD   B,B
53E7 11E952  02830      LD   DE,DSKLBL
53EA EDA0    02840 MOVE  LDI
53EC B0      02850      OR   B
53ED 280D    02860      JR   Z,AUTO
53EF 7E      02870      LD   A,(HL)
53F0 FE0D    02880      CP    0DH
53F2 280B    02890      JR   Z,AUTO
53F4 FE22    02900      CP    ""
53F6 2804    02910      JR   Z,AUTO
53F8 FE29    02920      CP    ')'
53FA 20EE    02930      JR   NZ,MOVE
53FC 210000  02940 AUTO  LD    HL,0
53FF 7C      02950      LD   A,H
5400 B5      02960      OR   L

```

5401	2828	02970	JR	Z,FILE		
5403	23	02980	INC	HL		
5404	7C	02990	LD	A,H		
5405	B5	03000	OR	L		
5406	CCD155	03010	CALL	Z,GETAUTO		
5409	063C	03020	LD	B,60		
540B	11AC52	03030	LD	DE,AUTOBUF		
540E	EDA0	03040	MOVEBTE	LDI		
5410	3E00	03050	LD	A,0		
5412	B0	03060	OR	B		
5413	2819	03070	JR	Z,FILE		
5415	7E	03080	LD	A,(HL)		
5416	FE0D	03090	CP	0DH		
5418	2810	03100	JR	Z,ENDAUTO		
541A	3A5B56	03110	LD	A,(AUTOFLAG)		
541D	B7	03120	OR	A		
541E	20EE	03130	JR	NZ,MOVEBTE		
5420	FE22	03140	CP	' ''		
5422	2806	03150	JR	Z,ENDAUTO		
5424	FE29	03160	CP	' )'		
5426	2802	03170	JR	Z,ENDAUTO		
5428	18E4	03180	JR	MOVEBTE		
542A	360D	03190	ENDAUTO	LD	(HL),0DH	
542C	EDA0	03200		LDI		
542E	210000	03210	FILE	LD	HL,0	
5431	7C	03220		LD	A,H	
5432	B5	03230		OR	L	
5433	2803	03240		JR	Z,FILE1	
5435	23	03250		INC	HL	
5436	7C	03260		LD	A,H	
5437	B5	03270		OR	L	
5438	CCDD55	03280	FILE1	CALL	Z,GETFILE	
543B	118457	03290		LD	DE,BUFF1	
543E	CD1C44	03300		CALL	@FSPEC	
5441	C27D55	03310		JP	NZ,DERROR	
5444	212057	03320		LD	HL,DFEXT	
5447	CD4B44	03330		CALL	@FEXT	
544A		03340		MOD_1	@FEXT1	
544A	0600	03350		LD	B,00H	
544C	21A457	03360		LD	HL,BUFF2	
544F	CD2044	03370		CALL	@INIT	
5452	C27D55	03380		JP	NZ,DERROR	
5455	3E05	03390		LD	A,05H	
5457	CD6B55	03400		CALL	PUTTOD	
545A	3E06	03410		LD	A,06H	
545C	CD6B55	03420		CALL	PUTTOD	
545F	212457	03430		LD	HL,MSG1	;Put header to file.
5462	0606	03440		LD	B,06H	;put 6 bytes.
5464	7E	03450	LOOP1	LD	A,(HL)	
5465	23	03460		INC	HL	
5466	CD6B55	03470		CALL	PUTTOD	
5469	10F9	03480		DJNZ	LOOP1	
546B	3AF455	03490		LD	A,(SCRFLG)	;Is screen flag set?
546E	B7	03500		OR	A	
546F	CA8854	03510		JP	Z,PUTBTM	;NO, Continue.
5472	21003C	03520		LD	HL,VIDMEM	;YES, Write video mem.
5475	22EF55	03530		LD	(WRTADR),HL	
5478	010004	03540		LD	BC,400H	;# of bytes in vid. mem.
547B	AF	03550		XOR	A	
547C	32F355	03560		LD	(FLAG),A	;Flag=0
547F	CD1755	03570		CALL	PUTBLK	
5482	210052	03580		LD	HL,START	



```

5485 22EF55 03590 LD (WRTADR),HL
5488 210052 03600 PUTBTM LD HL,START ;PUT Boot Time Module.
548B 015A01 03610 LD BC,ENTRY-START ;Length of BTM.
548E AF 03620 XOR A
548F 32F355 03630 LD (FLAG),A ;Flag=0
5492 CD1755 03640 CALL PUTBLK ;Do it.
5495 DD218856 03650 LD IX,MOD351
5499 03660 MOD_1 MOD151
5499 DD4E00 03670 LOOP3 LD C,(IX+00H)
549C DD23 03680 INC IX
549E DD4600 03690 LD B,(IX+00H)
54A1 DD23 03700 INC IX
54A3 79 03710 LD A,C
54A4 B0 03720 OR B
54A5 280F 03730 JR Z,FINDTOP
54A7 DD6E00 03740 LD L,(IX+00H)
54AA DD23 03750 INC IX
54AC DD6600 03760 LD H,(IX+00H)
54AF DD23 03770 INC IX
54B1 CD0655 03780 CALL PUTDATA
54B4 18E3 03790 JR LOOP3
54B6 CDF854 03800 FINDTOP CALL PUTZERO ;Find top of physical Mem
54B9 21FFFF 03810 LD HL,TRUE
54BC 7E 03820 LOOPS LD A,(HL)
54BD 47 03830 LD B,A
54BE 2F 03840 CPL
54BF 77 03850 LD (HL),A
54C0 BE 03860 CP (HL)
54C1 70 03870 LD (HL),B
54C2 2806 03880 JR Z,PUTHIGH
54C4 7C 03890 LD A,H
54C5 D640 03900 SUB 0040H
54C7 67 03910 LD H,A
54C8 18F2 03920 JR LOOPS
54CA ED5B1144 03930 PUTHIGH LD DE,(HIGH*) ;Put High Memory to File
54CE 03940 MOD_1 HIGH1*
54CE B7 03950 OR A
54CF ED52 03960 SBC HL,DE
54D1 DAF354 03970 JP C,NOHIGH
54D4 E5 03980 PUSH HL
54D5 C1 03990 POP BC
54D6 EB 04000 EX DE,HL
54D7 281A 04010 JR Z,NOHIGH
54D9 23 04020 INC HL
54DA CD0655 04030 CALL PUTDATA
54DD 3E02 04040 PUTTRAN LD A,02H ;Put Transfer address,
54DF CD6B55 04050 CALL PUTTOD ; Close file, Exit.
54E2 3E02 04060 LD A,02H
54E4 CD6B55 04070 CALL PUTTOD
54E7 210052 04080 LD HL,START ;START= entry point.
54EA CD6255 04090 CALL PUTADD
54ED CD2844 04100 CALL @CLOSE
54F0 C32D40 04110 JP @EXIT
54F3 CDF854 04120 NOHIGH CALL PUTZERO
54F6 18E3 04130 JR PUTTRAN
54F8 010200 04140 PUTZERO LD BC,02H
54FB 215756 04150 LD HL,ZERO
54FE AF 04160 XOR A
54FF 32F355 04170 LD (FLAG),A ;Flag = 0
5502 CD1755 04180 CALL PUTBLK
5505 C9 04190 RET
5506 E5 04200 PUTDATA PUSH HL ;Save address

```

5507	ED43F155	04210		LD	(LENSTOR),BC	;Save length
5508	210400	04220		LD	HL,04H	
550E	09	04230		ADD	HL,BC	;Length=length+4
550F	44	04240		LD	B,H	;New length in BC
5510	4D	04250		LD	C,L	
5511	E1	04260		POP	HL	;Restore address
5512	3EFF	04270		LD	A,TRUE	
5514	32F355	04280		LD	(FLAG),A	;flag=-1
5517	78	04290	PUTBLK	LD	A,B	;Put block to disk
5518	B7	04300		OR	A	; BC = Length of block
5519	280E	04310		JR	Z,SMLBLK	; HL = Load address
551B	C5	04320		PUSH	BC	
551C	010001	04330		LD	BC,100H	
551F	CD2955	04340		CALL	SMLBLK	
5522	C1	04350		POP	BC	
5523	05	04360		DEC	B	
5524	79	04370		LD	A,C	
5525	B0	04380		OR	B	
5526	C8	04390		RET	Z	
5527	18EE	04400		JR	PUTBLK	
5529	3E01	04410	SMLBLK	LD	A,01H	;Put small block to file.
552B	CD6B55	04420		CALL	PUTTOD	;255 or less bytes
552E	79	04430		LD	A,C	
552F	3C	04440		INC	A	
5530	3C	04450		INC	A	
5531	CD6B55	04460		CALL	PUTTOD	
5534	E5	04470		PUSH	HL	
5535	2AEF55	04480		LD	HL,(WRTADR)	
5538	CD6255	04490		CALL	PUTADD	
5538	09	04500		ADD	HL,BC	
553C	22EF55	04510		LD	(WRTADR),HL	
553F	E1	04520		POP	HL	
5540	3AF355	04530		LD	A,(FLAG)	
5543	B7	04540		OR	A	
5544	2813	04550		JR	Z,PTCBYT	
5546	E5	04560		PUSH	HL	
5547	2AF155	04570		LD	HL,(LENSTOR)	
554A	CD6255	04580		CALL	PUTADD	
554D	0D	04590		DEC	C	
554E	0D	04600		DEC	C	
554F	E1	04610		POP	HL	
5550	CD6255	04620		CALL	PUTADD	
5553	0D	04630		DEC	C	
5554	0D	04640		DEC	C	
5555	AF	04650		XOR	A	
5556	32F355	04660		LD	(FLAG),A	
5559	7E	04670	PTCBYT	LD	A,(HL)	;Put Bytes to disk
555A	23	04680		INC	HL	; number of bytes is in C
555B	CD6B55	04690		CALL	PUTTOD	
555E	0D	04700		DEC	C	
555F	20F8	04710		JR	NZ,PTCBYT	
5561	C9	04720		RET		
5562	7D	04730	PUTADD	LD	A,L	;Put address in HL to disk
5563	CD6B55	04740		CALL	PUTTOD	
5566	7C	04750		LD	A,H	
5567	CD6B55	04760		CALL	PUTTOD	
556A	C9	04770		RET		
556B	118457	04780	PUTTOD	LD	DE,BUFF1	
556E	CD1B00	04790		CALL	@PUT	
5571	200A	04800		JR	NZ,DERROR	
5573	C9	04810		RET		
5574	21D757	04820	ERROR	LD	HL,EMSG	

```

5577 CD6744 04830 CALL @DSPLY
557A C33040 04840 JP @ABORT
557D F6C0 04850 DERROR OR 0C0H ;Disk error routine.
557F CD0944 04860 CALL @ERROR
5582 C33040 04870 JP @ABORT
5585 21CD57 04880 GETDRV LD HL, INDRIVE
5588 0601 04890 LD B, 1
558A CDE255 04900 CALL INPUT
558D 7E 04910 LD A, (HL)
558E D630 04920 SUB 30H
5590 320852 04930 LD (DRVLBL+1), A
5593 C3D053 04940 JP DRIVE
5596 3A0852 04950 GETLBL LD A, (DRVLBL+1)
5599 4F 04960 LD C, A
559A CD0942 04970 CALL @CKDRV ;Is drive ready?
559D 04980 MOD_1 @CKDRV1
559D 2008 04990 JR NZ, NODFT ;NO, skip @DODIR.
559F 0604 05000 LD B, 4
55A1 21FA57 05010 LD HL, DFTLBL
55A4 CD1944 05020 CALL @DODIR
55A7 05030 MOD_1 @DODIR1
55A7 3E0D 05040 NODFT LD A, 0DH
55A9 320258 05050 LD (DFTLBL+8), A
55AC 21E957 05060 LD HL, DFTMSG
55AF CD6744 05070 CALL @DSPLY
55B2 219F42 05080 LD HL, KFLAG$ ;Get Keyboard flag.
55B5 05090 MOD_1 KFLAG1$
55B5 7E 05100 LD A, (HL)
55B6 325956 05110 LD (OLDKFLG), A ;Save it.
55B9 CBEE 05120 SET S, (HL) ;Force CAPS lock.
55BB 21C057 05130 LD HL, INLBL ;INLBL= Msg to display.
55BE 0608 05140 LD B, 8 ;8 = max length for lable
55C0 CDE255 05150 CALL INPUT
55C3 3A5956 05160 LD A, (OLDKFLG) ;Return Kflag$ to
55C6 329F42 05170 LD (KFLAG$), A ;previous state.
55C9 05180 MOD_1 KFLAG1$
55C9 3E00 05190 LD A, 0
55CB B8 05200 CP B
55CC C0 05210 RET NZ
55CD 21FA57 05220 LD HL, DFTLBL
55D0 C9 05230 RET
55D1 21AC57 05240 GETAUTO LD HL, INAUTO
55D4 3EFF 05250 LD A, TRUE
55D6 325B56 05260 LD (AUTOFLAG), A
55D9 061F 05270 LD B, 1FH
55DB 1805 05280 JR INPUT
55DD 219657 05290 GETFILE LD HL, INFILE
55E0 061F 05300 LD B, 1FH
55E2 CD6744 05310 INPUT CALL @DSPLY
55E5 212457 05320 LD HL, MSG1
55E8 CD4000 05330 CALL @KEYIN
55EB D0 05340 RET NC
55EC C32D40 05350 JP @EXIT
55EF 05360 SUBTTL '<Boot Data>'
55EF 05370 PAGE
55EF 0052 05380 WRTADR DW 5200H
55F1 0000 05390 LENSTOR DW 0000H
55F3 00 05400 FLAG DB 00H
55F4 0000 05410 SCRFLG DW 0000H
55F6 53 05420 PTABLE DM 'S' ;Parameter table
55FC F455 05430 DW SCRFLG
55FE 53 05440 DM 'SCREEN'

```

5604	F455	05450	DW	SCRFLG	
5606	51	05460	DM	'QUERY'	
560C	A953	05470	DW	QUERY+1	
560E	51	05480	DM	'Q'	
5614	A953	05490	DW	QUERY+1	
5616	46	05500	DM	'F'	
561C	2F54	05510	DW	FILE+1	
561E	46	05520	DM	'FILE'	
5624	2F54	05530	DW	FILE+1	
5626	41	05540	DM	'AUTO'	
562C	FD53	05550	DW	AUTO+1	
562E	41	05560	DM	'A'	
5634	FD53	05570	DW	AUTO+1	
5636	4C	05580	DM	'LABEL'	
563C	D953	05590	DW	LABEL+1	
563E	4C	05600	DM	'L'	
5644	D953	05610	DW	LABEL+1	
5646	44	05620	DM	'DRIVE'	
564C	0852	05630	DW	DRVLBL+1	
564E	44	05640	DM	'D'	
5654	0852	05650	DW	DRVLBL+1	
5656	00	05660	DB	00H	;End of param table.
5657	0000	05670	ZERO DW	0000H	;Always zero
0002		05680	OLDKFLG DS	2	
565B	00	05690	AUTOFLAG DB	00H	
565C	1800	05700	MOD151 DW	18H,4015H	;KIDCB\$,DODCB\$,PRDCB\$
5660	0600	05710	DW	06H,402DH	;@EXIT,@ABORT
5664	0300	05720	DW	03H,4039H	
5668	1C00	05730	DW	1CH,4049H	;HIGH\$,INTIM\$,INTVC\$,DBGSV\$
566C	5000	05740	DW	50H,4152H	
5670	0600	05750	DW	06H,4300H	;@KITSK,@ICNFG
5674	0300	05760	DW	03H,430FH	;SFLAG\$,EXDBG\$
5678	0200	05770	DW	02H,4310H	;EXDBG\$
567C	0300	05780	DW	03H,4315H	;used with DEBUG.
5680	0A00	05790	DW	0AH,43AEH	
5684	4600	05800	DW	46H,4388H	;Device control blocks.
5688	0100	05810	DW	01H,441FH	;DFLAG\$
568C	0100	05820	DW	01H,4423H	;KFLAG\$
5690	1800	05830	DW	18H,4500H	;TCB\$
5694	0300	05840	DW	03H,45DDH	
5698	5000	05850	DW	50H,4700H	;DCT\$
569C	0200	05860	DW	02H,48BEH	
56A0	0200	05870	DW	02H,48CEH	
56A4	0300	05880	DW	03H,48DEH	
56A8	1000	05890	DW	10H,4DF0H	;4DFE-FF = USTOR\$
56AC	00	05900	DC	12,0	
56B8	1A00	05910	MOD351 DW	1AH,4013H	;KI,DO,PRDCB\$ 4015-402C
56BC	0600	05920	DW	06H,402DH	;@EXIT,@ABORT
56C0	5000	05930	DW	50H,4152H	
56C4	0A00	05940	DW	0AH,41F3H	
56C8	0200	05950	DW	02H,4213H	;4214=Video scroll prot.
56CC	0300	05960	DW	03H,421DH	;@icnfg
56D0	0300	05970	DW	03H,4285H	;@KITSK
56D4	0100	05980	DW	01H,4289H	;DFLAG\$
56D8	0100	05990	DW	01H,429FH	;KFLAG\$
56DC	4800	06000	DW	48H,42B8H	;Device control Blocks.
56E0	0200	06010	DW	02H,4400H	;EXTDBG\$
56E4	0300	06020	DW	03H,4405H	;@DBGHK
56E8	0200	06030	DW	02H,4411H	;HIGH\$
56EC	0300	06040	DW	03H,442BH	;SFLAG\$
56F0	0300	06050	DW	03H,4439H	;@WRITE
56F4	1800	06060	DW	18H,4475H	;INTVC\$

```

56F8 1800      06070      DW      18H,4500H      ;TCB$
56FC 5000      06080      DW      50H,4700H      ;DCT$
5700 0200      06090      DW      02H,48B0H
5704 0200      06100      DW      02H,48CDH
5708 0300      06110      DW      03H,48DDH
570C 0F00      06120      DW      0FH,4D53H
5710 1000      06130      DW      10H,4DF0H      ;4DFE-FF = USTOR$
5714 00        06140      DC      12,0
5720 43        06150 DFEXT  DM      'CMD',0DH
5724 57        06160 MSG1   DM      'Warm Boot Utility Ver. 1.4 -- J. Kelly '
5748 48        06170      DM      'Hays 07/25/83 Based on:',0AH
5764 4C        06180      DM      'Ldos Configuration saver -- Les '
5784 40        06190 BUFF1  DM      'Mikesell 11/14/81',0DH
5796 46        06200 INFILE  DM      'Filename for output: ',03H
57A4          06210 BUFF2  EQU     INFILE+14
57AC 41        06220 INAUTO  DM      'Auto command line: ',03H
57C0 44        06230 INLBL  DM      'Disk Label: ',03H
57CD 44        06240 INDRIVE DM      'Drive #: ',03H
57D7 50        06250 EMSG   DM      'Parameter Error!',0DH,06H
57E9 44        06260 DFTMSG  DM      'Default label is '
57FA 20        06270 DFTLBL  DC      20,20H
580E          06280 $PRGEND EQU    $
535A          06290      END      ENTRY
    
```

00001 ;AUTOIND/ASM : Automatic Mod I/III independence

00002 ;

00003 \*LIST OFF

00004 ;\$PRGEND EQU \$ ;Place just before end.

00005 ;A call to the macro CKMOD should be made before any

00006 ; reference to model dependent addresses.

00007 ;After each reference to a model dependent address enter

00008 ; a call to the macro MOD\_1 #ADR\_1 where #ADR\_1 is the

00009 ; address of the model I reference.

```

00010 $PASS DEFL $PASS+1      00031 $CKMOD1 LD      A,B
00011 $MODTBL DEFL $PRGEND    00032      OR      C
00012 $MODPTR DEFL $MODTBL    00033      JR      Z,$CKMOD2
00013      IFEQ $PASS,2      00034      LD      L,(IX+0)
00014 $MODNUM DEFL $MODCTR    00035      LD      H,(IX+1)
00015      ENDIF      00036      PUSH   HL
00016 MOD_1 MACRO #ADR_1      00037      POP    IY
00017 $PCTEMP DEFL $          00038      LD      L,(IX+2)
00018      ORG $MODPTR      00039      LD      H,(IX+3)
00019      DW $PCTEMP-2      00040      LD      (IY+0),L
00020      DW #ADR_1      00041      LD      (IY+1),H
00021 $MODCTR DEFL $MODCTR+1  00042      INC    IX
00022 $MODPTR DEFL $          00043      INC    IX
00023      ORG $PCTEMP      00044      INC    IX
00024      ENDM      00045      INC    IX
00025 CKMOD MACRO      00046      DEC    BC
00026      LD A,(54H)      00047      JR      $CKMOD1
00027      DEC A      00048 $CKMOD2
00028      JR NZ,$CKMOD2    00049      ENDM
00029      LD BC,$MODNUM    00050 *LIST ON
00030      LD IX,$MODTBL
    
```

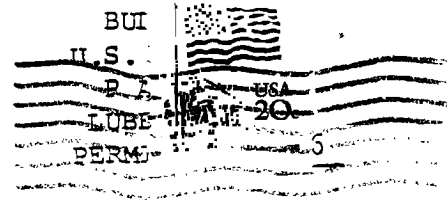
\*\*\*\*\* LOW ADVERTISING RATES \*\*\*\*\*

\* You can get a quarter page ( 4 inch wide by 5 inch tall ) advertisement in \*  
 \* COMPUTERBASE for only \$25.00. Ad copy should be photo ready and delivered \*  
 \* to one of the officers of the club no later than the 20th of the month \*  
 \* before you want your advertisement to run. Call Brett Pijan at 725-0102 \*  
 \* for more information about advertising in COMPUTERBASE. .

\*\*\*\*\*

**COMPUTERBASE**

2006-43rd Street  
Lubbock, Texas  
79412



Edmonton Users Group  
Box 11983  
Edmonton, Alberta, Canada  
T5G 3L1