

```

0001      TITLE      SIMPLE DEBUG ROUTINE
0002      PRINTR      D70,W80,P
0003      MLIST
0004      ;
0005      ;   This is a sample SOURCE PROGRAM File.
0006      ;
0007      ;   Use it to practice EDIT functions. You can also use
0008      ;   it as input to the Assembler. Use a right margin of
0009      ;   80 so that print truncation does not occur. Or set
0010      ;   your printer to 'condensed printing' and change the
0011      ;   PRINTR statement above to D70,W132 for full width
0012      ;   screen use.
0013      ;
0014      ;   The DEBUG routine below will display all registers when
0015      ;   called. It then waits for the ESCAPE key to be pressed
0016      ;   before returning to the caller.
0017      ;
0018      ;   Assemble PROGRAM.001 to TEMP.BIN, then test from BASIC.
0019      ;
0020      ;   005 mode 2                      ;*****
0021      ;   010 memory &4FFF                ;# PRESS THE DOWN-ARROW #;
0022      ;   020 LOAD "TEMP.BIN"              ;# KEY UNTIL THE SCREEN #;
0023      ;   030 CALL &5000                  ;# STARTS MOVING ..... #;
0024      ;   040 STOP                        ;*****
0025      ;*****
0026      ;# BASIC EDIT KEYS.                  #;
0027      ;# ~~~~~~                          #;
0028      ;# Up,Down,Left,Right Arrows - Move Cursor, Scroll ... #;
0029      ;# ~~~~~~                          #;
0030      ;# Shift and Control Up,Down - Single, Screen Scroll. #;
0031      ;# ~~~~~~                          #;
0032      ;# Shift Left,Right                - CHARACTER Delete, Insert. #;
0033      ;# ~~~~~~                          #;
0034      ;# Control Left,Right              - LINE Delete, Insert. #;
0035      ;# ~~~~~~                          #;
0036      ;# Tab & Control.Tab                - Tab right or right. #;
0037      ;# ~~~~~~                          #;
0038      ;# CLR and the DEL keys            - Delete character, Reverse #;
0039      ;# ~~~~~~                          #;
0040      ;# ESCAPE key                      - Display [ FUNCTION KEYS] #;
0041      ;# ~~~~~~                          #;
0042      ;# Island ENTER key                - Start, Stop KS Memory. #;
0043      ;# Island 1 - 9 keys              - Replay a KeyStroke Memory #;
0044      ;# Island (. ) key                 - Print Code from cursor #;
0045      ;# ~~~~~~                          #;
0046      ;# SUGGEST YOU PLAY WITH THE TEXT ON THIS SCREEN ..... #;
0047      ;# THEN PRESS (CTRL-A) FOLLOWED BY (Y) TO (ABORT RESET) #;
0048      ;*****
0049      ;
0050      ;- - - - -
0051      ;- - - - -
0052      ;- - - - -
0053      ;- - - - -
0054      ;
0055      ;   ( SET THE PROGRAM ORIGIN POINT )
0056      ;
0057      ORG      5000H
0058      ;
0059      ;   ( NOW, DEFINE A MACRO WHICH WILL BE USED TO OUTPUT )
0060      ;   ( THE VARIOUS REGISTER PAIRS. #PJ WILL BE CHANGED )
0061      ;   ( WITH EACH USE. (NOTE THE RO$YM LABEL CHANGES..) )
0062      ;
0063      ;

```

```

0065      REGOUT:      MACRO      #P1      ; DEFINE MACRO
0066      ROWSYM:      CALL      STRING      ; CALL WRITER
0067      LD      A,(DB#P1+1)      ; LD A,(DB??+1)
0068      CALL      HEXOUT      ; CONVERT IT.
0069      LD      A,(DB#P1)      ; LD A,(DB??)
0070      CALL      HEXOUT      ; CONVERT IT.
0071      ENDM
0072      ;
0073      ;
0074      ;      ( CALL TO DEBUG COMES HERE )
0075      ;      ( SAVE ALL REGISTERS LOCALLY )
0076      ;
0077      DEBUG:      EQU      #
0078      5000      323B51      LD      (DBA),A
0079      5003      ED433C51      LD      (DBBC),BC
0080      5007      ED533E51      LD      (DBDE),DE
0081      500B      224051      LD      (DBHL),HL
0082      500E      DD224251      LD      (DBIX),IX
0083      5012      FD224451      LD      (DBIY),IY
0084      5016      ED734051      LD      (DBSP),SP
0085      ;
0086      ;      ( POP THE STACK TO OBTAIN THE )
0087      ;      ( CALLERS ADDRESS THEN RESTORE. )
0088      ;
0089      ;
0090      501A      E1      POP      HL
0091      501B      224651      LD      (DBPC),HL
0092      501E      E5      PUSH     HL
0093      ;
0094      ;      ( OBTAIN THE CURRENT CURSOR POSITION )
0095      ;      ( AND SAVE IT (ON STACK) FOR LATER. )
0096      ;
0097      501F      CD78BB      CALL    0BB78H
0098      5022      E5      PUSH     HL
0099      ;
0100      ;      ( SET THE DISPLAY CURSOR POSITION )
0101      ;      ( H = COLUMN AND L = ROW VALUES. )
0102      ;
0103      5023      2601      LD      H,1
0104      5025      2E18      LD      L,24
0105      5027      CD75BB      CALL    0BB75H
0106      ;
0107      ;      ( REVERSE THE PEN AND PAPER INKS )
0108      ;
0109      502A      CD9CBB      CALL    0BB9CH
0110      ;
0111      ;      ( INITIALISE THE STRING ROUTINE POINTER )
0112      ;
0113      502D      21F050      LD      HL,CSTRG
0114      5030      224B51      LD      (STRGC),HL
0115      ;
0116      ;      ( USING THE SUBROUTINES STRING AND )
0117      ;      ( HEXOUT, EACH PAIR OF REGISTERS ARE )
0118      ;      ( SENT TO THE SCREEN, PRECEDED BY THE )
0119      ;      ( RELEVANT TEXT CHARACTERS, )
0120      ;
0121      5033      CD0950      CALL    STRING
0122      5036      9A3B51      LD      A,(DBA)
0123      5039      CD5DBB      CALL    0BB5DH
0124      503C      9A3B51      LD      A,(DBA)
0125      503F      CD1151      CALL    HEXOUT
0126      ;

```

```

0127 5042                                REGOUT  BC                ; MACRO CALL ;
      5042 CDD850 +R00001:             CALL    STRING            ; CALL WRITER
      5045 3A3D51 +                     LD      A,(DBBC+1)         ; LD A,(DB??+1)
      5048 CD1151 +                     CALL    HEXOUT            ; CONVERT IT.
      504B 3A3C51 +                     LD      A,(DBBG)           ; LD A,(DB??)
      504E CD1151 +                     CALL    HEXOUT            ; CONVERT IT.
0128                                ;
0129 5051                                REGOUT  DE                ; MACRO CALL ;
      5051 CDD950 +R00002:             CALL    STRING            ; CALL WRITER
      5054 3A3F51 +                     LD      A,(DBDE+1)         ; LD A,(DB??+1)
      5057 CD1151 +                     CALL    HEXOUT            ; CONVERT IT.
      505A 3A3E51 +                     LD      A,(DBDE)           ; LD A,(DB??)
      505D CD1151 +                     CALL    HEXOUT            ; CONVERT IT.
0130                                ;
0131 5060                                REGOUT  HL                ; MACRO CALL ;
      5060 CDD950 +R00003:             CALL    STRING            ; CALL WRITER
      5063 3A4151 +                     LD      A,(DBHL+1)         ; LD A,(DB??+1)
      5066 CD1151 +                     CALL    HEXOUT            ; CONVERT IT.
      5069 3A4051 +                     LD      A,(DBHL)           ; LD A,(DB??)
      506C CD1151 +                     CALL    HEXOUT            ; CONVERT IT.
0132                                ;
0133 506F                                REGOUT  IX                ; MACRO CALL ;
      506F CDD950 +R00004:             CALL    STRING            ; CALL WRITER
      5072 3A4351 +                     LD      A,(DBIX+1)         ; LD A,(DB??+1)
      5075 CD1151 +                     CALL    HEXOUT            ; CONVERT IT.
      5078 3A4251 +                     LD      A,(DBIX)           ; LD A,(DB??)
      507B CD1151 +                     CALL    HEXOUT            ; CONVERT IT.
0134                                ;
0135 507E                                REGOUT  IY                ; MACRO CALL ;
      507E CDD950 +R00005:             CALL    STRING            ; CALL WRITER
      5081 3A4551 +                     LD      A,(DBIY+1)         ; LD A,(DB??+1)
      5084 CD1151 +                     CALL    HEXOUT            ; CONVERT IT.
      5087 3A4451 +                     LD      A,(DBIY)           ; LD A,(DB??)
      508A CD1151 +                     CALL    HEXOUT            ; CONVERT IT.
0136                                ;
0137 508D                                REGOUT  PC                ; MACRO CALL ;
      508D CDD950 +R00006:             CALL    STRING            ; CALL WRITER
      5090 3A4751 +                     LD      A,(DBPC+1)         ; LD A,(DB??+1)
      5093 CD1151 +                     CALL    HEXOUT            ; CONVERT IT.
      5096 3A4651 +                     LD      A,(DBPC)           ; LD A,(DB??)
      5099 CD1151 +                     CALL    HEXOUT            ; CONVERT IT.
0138                                ;
0139 509C                                REGOUT  SP                ; MACRO CALL ;
      509C CDD850 +R00007:             CALL    STRING            ; CALL WRITER
      509F 3A4951 +                     LD      A,(DBSP+1)         ; LD A,(DB??+1)
      50A2 CD1151 +                     CALL    HEXOUT            ; CONVERT IT.
      50A5 3A4851 +                     LD      A,(DBSP)           ; LD A,(DB??)
      50A8 CD1151 +                     CALL    HEXOUT            ; CONVERT IT.
0140                                ;
0141                                ;
0142                                ;      ( WRITE FINAL CLOSE BRACKET TO SCREEN ).
0143                                ;
      0144 50AB 3E29                     LD      A,41
      0145 50AD CD5DBB                   CALL    0BB5DH
0146                                ;
0147                                ;      ( NOW WAIT FOR ESC KEY TO BE PRESSED ).
0148                                ;
      0149 50BB CD18BB                   WAIT_LOOP: CALL    0BB18H
0150                                ;
      0151                                ;      ( KEY WAS PRESSED. IF IT IS NOT THE )
      0152                                ;      ( ESCAPE KEY, IGNORE IT, ELSE EXIT.. )
      0153                                ;
0154 50B3 FEFC                           CP      252                ; esc key?

```

```

0155 50B5 20F9          JR      NZ, WAIT_LOOP      ; no
0156 50B7 E1            POP      HL                ; yes
0157 50B8 CD75BB        CALL     0BB75H            ; Reset cursor
0158                   ;
0159                   ;      ( REVERSE PEN AND PAPER INKS TO NORMAL )
0160                   ;
0161 50BB CD9CBB        CALL     0BB9CH
0162                   ;
0163                   ;      ( RESTORE ALL REGISTERS )
0164                   ;
0165 50BE 3A3B51          LD        A, (DBA)
0166 50C1 E04B3C51       LD        BC, (DBBC)
0167 50C5 E05B3E51       LD        DE, (DBDE)
0168 50C9 2A4051        LD        HL, (DBHL)
0169 50CC DD2A4251       LD        IX, (DBIX)
0170 50D0 FD2A4451       LD        IY, (DBIY)
0171 50D4 ED7B4851       LD        SP, (DBSP)
0172                   ;
0173                   ;      ( RETURN TO CALLING ROUTINE / USER ).
0174                   ;
0175 50D8 C9              RET                ; ***** >>> EXIT >>>
0176                   ;
0177                   ; -----
0178                   ;
0179                   ;      ( SUBROUTINE 'STRING' WRITES NEXT 4      )
0180                   ;      ( CHARACTERS FROM 'CSTRING' EACH TIME )
0181                   ;      ( IT IS CALLED.                          )
0182                   ;
0183 50D9                STRING: EQU      $
0184 50DB LD              LD        IY, (STRGC) ; Get Pointer
0185 50DD 0604            LD        B, 4      ; Set counter
0186 50DF                STRING2: EQU     $
0187 50E1 LD              LD        A, (IY+0) ; Get Char.
0188 50E2 FD23            INC        IY      ; Advance Ptr
0189 50E4 C5             PUSH      BC      ; Save counter
0190 50E5 GD50BB         CALL     0BB5DH    ; Display Char
0191 50E8 C1             POP        BC      ; Get counter
0192 50E9 10F4          DJNZ      STRING2 ; Loop 4 Times
0193 50EB FD224B51      LD        (STRGC), IY ; Save pointer
0194 50EF C9            RET                ;
0195                   ;
0196 50F0 2841463D       CSTRG: DB      '(AF= BC= DE= HL= '
0197 5100 2043583D      DB      ' IX= IY= PC= SP= '
0198                   ;
0199                   ;      ( SUBROUTINE HEXOUT IS CALLED WITH A VALUE )
0200                   ;      ( IN REG, A WHICH IS CONVERTED TO HEX-ASCII )
0201                   ;      ( AND WRITTEN TO THE SCREEN.                )
0202                   ;
0203 5111                HEXOUT: EQU     $
0204 5113 LD              LD        (TEMP), A
0205 5114 CB3F           SRL        A
0206 5116 CB3F           SRL        A
0207 5118 CB3F           SRL        A
0208 511A CB3F           SRL        A
0209 511C CD2E51        CALL     HD
0210 511F 3A4A51        LD        A, (TEMP)
0211 5122 CBBF          RES        7, A
0212 5124 CBB7          RES        6, A
0213 5126 CBAF          RES        5, A
0214 5128 CBA7          RES        4, A
0215 512A GD2E51       CALL     HD
0216 512D C9            RET
0217 512E                HD: EQU      $

```

```

0218 512E C630          ADD      A,48
0219 5130 FE3A          CP       58
0220 5132 FA3751        JP       M,H02
0221 5135 C607          ADD      A,7
0222 5137          H02:    EQU     $
0223 5137 CD50BB        CALL     0BB5DH
0224 513A C9           RET
0225                ;
0226                ; [ LOCAL STORAGE AREA FOR CALLERS REGISTERS ]
0227                ; [ ENDBIN STOPS CODE GENERATION ... ]
0228                ; ~~~~~
0229 513B                ENDBIN
0230                ;
0231 513B          DBA:      DS      1
0232 513C          DBBC:    DS      2
0233 513E          DBDE:    DS      2
0234 5140          DBHL:    DS      2
0235 5142          DBIX:    DS      2
0236 5144          DBJY:    DS      2
0237 5146          DBPC:    DS      2
0238 5148          DBSP:    DS      2
0239 514A          TEMP:    DS      1
0240 514B          STRAC:    DS      2
0241                ;
0242                ;.. [ END OF SOURCE PROGRAM (END IS OPTIONAL.) ]
0243                ;
0244 514D                END

```

Number of Errors.: 0000

Number of Symbols.: 0026

Symbol table from.: 4A76 to 4BC8

Macro List from...: 3000 to 30A7

Number of X-refs.: 0065

X-ref table from.: 7EBA to 7FFE

Free Symbol Memory: 22406

File start: 5000 end: 513B length: 013B

CSTRG	50F0	DBA	5138	DBBC	513C
DBDE	513E	DBHL	5140	DBIX	5142
DBIY	5144	DBPC	5146	DBSP	5148
N DEBUG	5000	HEXDUT	5111	HQ	512E
HQ2	5137	N REGOUT	5000	N R00001	5042
N R00002	5051	N R00003	5060	N R00004	506F
N R00005	507E	N R00006	508D	N R00007	509C
STRGC	514B	STRING	50D9	STRING2	50DF
TEMP	514A	WAIT_LOOP	50B0		

No.	Filename	Line #																					
01	program.000	00001																					
CSTRG	50F0	01 0113																					
DBA	513B	01 0078	01 0122	01 0124	01 0165																		
DBBC	513C	01 0078	01 0127	01 0127	01 0166																		
DBDE	513E	01 0080	01 0128	01 0128	01 0167																		
DBHL	5140	01 0081	01 0131	01 0131	01 0168																		
DBIX	5142	01 0082	01 0133	01 0133	01 0169																		
DBIY	5144	01 0083	01 0135	01 0135	01 0170																		
DBPC	5146	01 0091	01 0137	01 0137																			
DBSP	5148	01 0084	01 0138	01 0138	01 0171																		
N DEBUG	5000																						
HEXOUT	5111	01 0125	01 0127	01 0127	01 0129	01 0129	01 0131																
		01 0131	01 0133	01 0133	01 0135	01 0135	01 0137																
		01 0137	01 0139	01 0139																			
HO	512E	01 0209	01 0215																				
HO2	5137	01 0220																					
N REGOUT	5000																						
N RD0001	5042																						
N RD0002	5051																						
N RD0003	5060																						
N RD0004	506F																						
N RD0005	507E																						
N RD0006	508D																						
N RD0007	509C																						
STRGC	514B	01 0114	01 0184	01 0193																			
STRING	50D8	01 0121	01 0127	01 0129	01 0131	01 0133	01 0135																
		01 0137	01 0139																				
STRING2	50DF	01 0192																					
TEMP	514A	01 0204	01 0210																				
WAIT_LOOP	50B0	01 0155																					