Some Debugging Sourcecode found on a BBC Micro Fig-Forth Disk#

( Toolkit            EEDIT           1 of 2    )
VARIABLE EBLK        ( will hold error value of BLK )
VARIABLE E>IN        ( will hold error value of >IN )

: (EEDIT) ( -- )       ( new error routine to )
  BLK @  EBLK !         ( transfer values of BLK )
  >IN @  WBFR C@ - 1-   E>IN !   ( and >IN to their error)
  $MSG ;                 ( counterparts )

ASSIGN MESSAGE TO-DO (EEDIT)       ( insert new error routine)
                                 ( into error processing procedures)

( Toolkit            EEDIT           2 of 2    )

: EEDIT ( -- )         ( display details of block and )
  ( line where error occured and )
  ( text of line )
  EBLK @  ?DUP
  IF    CR ." Error in block " DUP . DUP SCR !
  E>IN @ 64 / ." line " DUP .
  SWAP CR .LINE
  ELSE ." Last Error was from the Keyboard "
  THEN
  CR   ;

HERE FENCE !

( Toolkit            SHOW           1 of 2    )

: NEWCREATE ( -- )     ( new CREATE which saves )
  BLK @ ,               ( block and offset in header )
  >IN @ ,
  (CREATE) ;

ASSIGN CREATE TO-DO NEWCREATE

: FORGET                FORGET -4 ALLOT ;   ( FORGET 4 extra bytes )
HERE CONSTANT MARKER    ( cannot SHOW dictionary entry before here)
HERE FENCE !            ( set FENCE to prevent inadvertent FORGETting )

( Toolkit            SHOW           2 of 2    )
: SHOW FIND ?DUP
   ( usage :- SHOW <word> displays )
   IF DUP MARKER <
      ( where <word> was defined )
      IF ." Can't SHOW " 2+ NFA ID.
      ELSE
         2+ NFA DUP
         2- 2- @ ?DUP
         IF CR ." Defined in block " DUP . DUP SCR !
         SWAP 2- @ 64 / ." line " DUP .
         SWAP CR .LINE
         ELSE ID. ." was defined from the keyboard"
         THEN
         THEN
      ELSE ." Word not found "
   THEN
   CR ;