## Summary

The following software shows how to suppress the "ok"
prompt.

```
\ Copyright 1998 Mosaic Industries, Inc.  All rights reserved.

\ This source code implements a standard forth interpreter (typically
\ named QUIT), and allows the user to comment out and suppress the "ok" prompt.
\ All multitasking and error trapping functions of the standard interpreter
\ are preserved.  Works for QED kernel versions V3.1x and V4.0x.


4 USE.PAGE                 \ not needed if memory map has already been set
ANEW NOT.OK.STUFF \ forget marker

\ ********* CONSTANT AND VARIABLE DECLARATIONS *************

PARITY 6 XN- XCONSTANT LOCALS.ARE.COMPILING
     \ system variable, zeroed on entry to interpreter

TRANSMITTING 2XN- XCONSTANT ACTIVE.SERIAL
     \ system variable, holds addr of SERIAL1 or SERIAL2 resource variable;
     \ set by USE.SERIALx, used by interpreter to get and release


TRACE DROP (STATUS) - 2+ USER TRACE.INDENT
     \ a byte-sized user variable, = #spaces indented by TRACE; zeroed by QUIT

\ ****** PRIMITIVES **************

: *RELEASE.SERIAL        ( -- )
\ if permission flag in user variable SERIAL.ACCESS is >= RELEASE.AFTER.LINE,
\ then go ahead and release it; otherwise, don't.
\ if resource is released, we PAUSE to give others access to the resource.
\ serial.access values are: release.never {0};release.after.line{1};
\ or release.after.char {2}
     SERIAL.ACCESS @ RELEASE.AFTER.LINE >=
     IF   ACTIVE.SERIAL @ DEFAULT.PAGE RELEASE
          PAUSE                     \ give other tasks a shot at serial line
     ENDIF
;
```

```
: GET.SERIAL.FOR.LINE   ( -- )
       \ if serial.access contains RELEASE.AFTER.LINE,
       \ GETs the in.use serial resource. Otherwise does nothing. Called by QUIT.
       SERIAL.ACCESS @ RELEASE.AFTER.LINE =
       IF     ACTIVE.SERIAL @ DEFAULT.PAGE GET
       ENDIF
;

: *OK?      ( -- ) \ ok prompt is commented out.
       \ at end of interpretation line, prints 'ok' and prints stack if needed
       \ used by QUIT and BREAK;
       \ aborts on dstack underflow, does COLD if S0 isn't in common ram
       \ if allowed by +SERIAL.ACCESS flag, and if we're executing,
       \ releases serial resource variable {SERIAL1.RESOURCE or SERIAL2.RESOURCE,
       \     whichever is in use} at the end of the line
       \ and pauses to give other tasks a chance to use serial.
       STATE @ 0=
       IF                      \ if we're executing...
\          SPACE ." ok"      \ commented out to suppress ok print!!
           DEBUG @                 \ if debug flag is set...
           IF   DEPTH 0<>   \ if items are on dstack...
                  IF .S ENDIF \ ...print stack; aborts on dstack underflow
           ENDIF
           *RELEASE.SERIAL   ( -- ) \ release & pause to give other tasks a chance
       ENDIF
;

\ ********** THE INTERPRETER **************

\ Note: any errors that cause an ABORT will terminate the execution
\ of FORTH.INTERPRETER.  To automatically re-enter the customized interpreter,
\ post a customized abort handler as described in appendix B of the
\ Forth software manual.   For example:
\    : MY.ABORT.ACTION
\        SP!  RP!   FORTH DEFINITIONS  FORTH.INTERPRETER ;
\ CFA.FOR MY.ABORT.ACTION  UABORT X!      \ install new action
\ CUSTOM.ABORT ON                                \ set flag so custom abort is called.



: FORTH.INTERPRETER     ( -- )      \ replacement for QUIT, the standard interpreter.
\ infinite loop, terminated by error-induced ABORTs.
       [COMPILE] [                        \ set execution mode {set state = 0 }
       LOCALS.ARE.COMPILING OFF      \ clear locals.are.compiling
       0 TRACE.INDENT C!                 \ clear #spaces indented by TRACE
       BEGIN
           STATE @ 0=
           IF   RP! ENDIF
           CR                          \ issue carriage return & line feed
           QUERY               \ read a line of text into TIB,init 0= >IN = BLK
           INTERPRET           \ move 1 line to pocket and interpret;check dstack
           *OK?                \ ok prompt is commented out in this routine.
           GET.SERIAL.FOR.LINE \ get if RELEASE.AFTER.LINE is specified
       AGAIN                              \ this is an infinite loop,terminated by errors
;
```

# Mosaic Industries

**5437 Central Ave Suite 1, Newark, CA 94560          Telephone: (510) 790-8222          Fax: (510) 790-0925**