



Summary

The following software shows how to install fast non-multitasking routines as serial primitives KEY EMIT and ?KEY.

```
\ This code installs fast non-multitasking routines as the serial primitives
\ KEY EMIT and ?KEY. Three functions named KEY.FASTER EMIT.FASTER and ?KEY.FASTER
\ are defined. These routines can be called directly for maximum speed,
\ or they called via the vectored serial routines KEY EMIT and ?KEY
\ which are in turn called by higher level functions such as ." " and TYPE.
\ Unlike the standard serial primitives, these routines do not call
\ PAUSE while waiting for the character buffer to fill or empty,
\ and they do not call GET and RELEASE. Avoiding these multitasking calls
\ saves time. NOTE: use caution if your system uses multitasking.

\ Those who need to optimize serial performance should also consider
\ using interrupt-based queued serial on the primary serial port.
\ This maximizes efficiency because the processor does not idly wait
\ for the transmit buffer to empty or the receive buffer to fill;
\ rather, the processor only reads/writes the serial buffer when action
\ is required. This frees up more "foreground" time for the processor.
\ The "Queued Serial" source code file available from Mosaic Industries
\ implements this buffered interrupt-driven serial I/O efficiently.

\ Copyright 1997 Mosaic Industries, Inc. All Rights Reserved.
\ Disclaimer: THIS SOFTWARE IS PROVIDED ON AN "AS IS" BASIS, WITHOUT ANY
\ WARRANTIES OR REPRESENTATIONS EXPRESS OR IMPLIED, INCLUDING, BUT NOT
\ LIMITED TO, ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS
\ FOR A PARTICULAR PURPOSE.
```

HEX

```
4 USE. PAGE          \ comment this out if you have already set up the memory map.
ANEW FAST. SERIAL. CODE
```

```
802E CONSTANT SCI. STATUS. REG          \ SCSR
802F CONSTANT SCI. DATA. REG           \ SCDR
20 CONSTANT DATA. RECEIVED. MASK      \ bit mask for SCI. STATUS. REG
80 CONSTANT DATA. XMITTED. MASK       \ bit mask for SCI. STATUS. REG
```

```

CODE ?KEY.FASTER      ( -- flag ) \ flag = true if char has been received
\ this ?KEY routine does not use GET or RELEASE, and does not PAUSE.
\ use with care in multitasking systems.
0 IMM LDX              \ X <- false flag
SCI.STATUS.REG EXT LDAA
DATA.RECEIVED.MASK IMM BITA \ test bit 5, set upon receive
NE IF,
    DEX                \ make flag true if char was received
THEN,
DEY DEY
0 IND, Y STX          ( -- flag )
RTS
END. CODE

CODE KEY.FASTER      ( -- b ) \ waits for char, puts it on dstack
\ this KEY routine does not use GET or RELEASE, and does not PAUSE.
\ use with care in multitasking systems.
BEGIN,
    SCI.STATUS.REG EXT LDAA
    DATA.RECEIVED.MASK IMM BITA \ test bit 5, set upon receive
NE UNTIL,
    SCI.DATA.REG EXT LDAB        \ wait until char received
CLRA
DEY DEY
0 IND, Y STD                    ( -- b )
RTS
END. CODE

CODE EMIT.FASTER     ( b -- ) \ waits for prev char output, then sends
\ this EMIT routine does not use GET or RELEASE, and does not PAUSE.
\ use with care in multitasking systems.
1 IND, Y LDAB          \ B <- char
INY INY                ( -- ) \ drop
BEGIN,
    SCI.STATUS.REG EXT TST      \ test bit 7, xmit buffer empty
MI UNTIL,
    SCI.DATA.REG EXT STAB      \ wait until previous char is xmitted
RTS                        \ send char
END. CODE

: INSTALL.FASTER.SERIAL ( -- )
\ installs above routines so they are automatically called
\ by the printing routines such as . F. ." " and TYPE
CFA.FOR ?KEY.FASTER U?KEY X!
CFA.FOR KEY.FASTER UKEY X!
CFA.FOR EMIT.FASTER UEMIT X!
;

```

```
: TYPE.FASTER      ( xaddr\count -- )
  \ this demo routine calls EMIT.FAST to print the specified string;
  \ xaddr is the first byte in the string, not the address of the count.
  0 DO  XDUP C@ EMIT.FASTER
      1XN+
  LOOP
  XDROP
  ;

\ To use the vectored serial drivers, execute:
\   INSTALL.FASTER.SERIAL  ." Hi There"

\ To use direct call to EMIT.FASTER, execute:
\   ." Hi there everyone!" COUNT TYPE.FASTER
```

The information provided herein is believed to be reliable; however, Mosaic Industries assumes no responsibility for inaccuracies or omissions. Mosaic Industries assumes no responsibility for the use of this information and all use of such information shall be entirely at the user's own risk.

Mosaic Industries

5437 Central Ave Suite 1, Newark, CA 94560

Telephone: (510) 790-8222

Fax: (510) 790-0925