

Basic Fast Stack and Fast Jump#

from ANTIC [Issue February 1988](#)

Table of Contents

- [Basic Fast Stack and Fast Jump](#)
- [Fast Stack](#)
- [Fast Jump](#)

Fast Stack#

```
0100 ; FAST STACK
0110 ; BY BILL BODENSTEIN
0120 ; (c)1987, ANTIC PUBLISHING
0130 ;
0140 ;EQUATES
0150 ;
0160 PORTB = $D301 ;Toggle ROM
0170 BASIC.ON = 253
0180 BASIC.OFF = 255
0190 ;
0200 LDA = 165 ;Decimal opcode
0210 STMCUR = $8A ;BASIC line ptr
0220 FORLN = $A0 ;Line # put here
0230 ;
0240 PUT.PATCH1 = $A071 ;Patch after
0250 ; STOP/END
0260 PUT.PATCH2 = PUT.PATCH1+5
0270 ;
0280 ; Relocatable, but
0290 *= $0600 ;could be called
0300 ; via USR(1536)
0310 STARTCODE
0320 PLA ;Remove # args
0330 COPY.BASIC
0340 LDA #$A0 ;Start of BASIC
0350 STA $E1
0360 LDA #$00
0370 STA $E0
0380 TAY
0390 LOOP1
0400 LDX #BASIC.ON
0410 STX PORTB ;BASIC ROM on
0420 LDA ($E0),Y ;Get a byte
0430 LDX #BASIC.OFF
0440 STX PORTB ;BASIC RAM on
0450 STA ($E0),Y ;Copy byte
0460 INY
0470 BNE LOOP1 ;And loop
0480 ;
0490 INC $E1
0500 LDA $E1
0510 CMP #192 ;Until all moved
0520 BNE LOOP1
```

```

0530 ;
0540 MODIFY.BASIC
0550     LDA #STMCUR+1 ;LDA ($8A),Y=>
0560     STA $B6C6     ; LDA $8B
0570     LDA #LDA     ; and LDA $8A
0580     STA $B6C0
0590     STA $B6C5
0600 ;
0610     LDA # <PUT.PATCH2
0620     STA $BDCC     ;Change JSR from
0630     LDA # >PUT.PATCH2
0640     STA $BDCD     ;$B816 to patch2
0650 ;
0660 ;Install patch to re-enable ROM
0670 ;at STOP or END, and patch to
0680 ;change line pointer.
0690     LDX #ENDCODE-PATCH1-1
0700     LDY #ENDCODE-STARTCODE-1
0710 LOOP2
0720     LDA ($D4),Y ;Move bytes from
0730     STA PUT.PATCH1,X ;USR code
0740     DEY
0750     DEX
0760     BPL LOOP2     ;Done when patch
0770 ;
0780     RTS           ;installed
0790 ;
0800 ;Patches to be placed in code
0810 ;after STOP/END. Note: once
0820 ;BASIC ROM is enabled by patch1,
0830 ;patch2 won't be executed.
0840 PATCH1
0850     LDX #BASIC.ON
0860     STX PORTB
0870 ;
0880 PATCH2
0890     LDA FORLN     ;Ln addr is here
0900     STA STMCUR   ;Point to it
0910     LDA FORLN+1
0920     STA STMCUR+1
0930     LDY #2       ;(Rest is the
0940     LDA (STMCUR),Y ; same)
0950     STA $9F
0960     CLC
0970     RTS
0980 ENDCODE

```

Fast Jump#

```

0100 ; FAST JUMP
0110 ; BY BILL BODENSTEIN
0120 ; (c)1987, ANTIC PUBLISHING
0130 ;
0140 ; EQUATES
0150 ;
0160 PORTB = $D301     ;Toggle ROM here
0170 BASIC.ON = 253
0180 ;
0190 JSR = 32         ;Decimal opcodes

```

```

0200 NOP = 234
0210 STMTAB = $88 ;Start of prog
0220 STMCUR = $8A ;Current line
0230 FORLN = $A0 ;Lnno saved here
0240 ;
0250 PUT.PATCH = $A000 ;Mem for patch
0260 JSR.HERE = PUT.PATCH+6 ;Actual
0270 ; code
0280 ;
0290 *= $0600 ;Relocatable but
0300 ; could be called
0310 ; via USR(1536)
0320 STARTCODE
0330 PLA ;Remove # args
0340 ;
0350 ;Before searching for line,
0360 ;jump to patch.
0370 MODIFY.BASIC
0380 LDA #JSR ;JSR PATCH
0390 STA $A9AA
0400 LDA # <JSR.HERE
0410 STA $A9AB
0420 LDA # >JSR.HERE
0430 STA $A9AC
0440 LDA #NOP ;NOP
0450 STA $A9AD
0460 ;
0470 ;Install patch in unused (except
0480 ;with NEW) BASIC RAM.
0490 LDX #ENDCODE-PATCH-1
0500 LDY #ENDCODE-STARTCODE-1
0510 LOOP
0520 LDA ($D4),Y ;Move bytes from
0530 STA PUT.PATCH,X ;USR code
0540 DEY
0550 DEX
0560 BPL LOOP ;Done when patch
0570 RTS ;installed
0580 ;
0590 ;Patch to be installed in BASIC
0600 ;RAM. If NEW occurs, BASIC ROM
0610 ;is enabled so that patch isn't
0620 ;accidentally executed.
0630 ;(Remember: With FAST-STACK, ROM
0640 ;is always enabled in direct
0650 ;mode.)
0660 PATCH
0670 NOP
0680 LDX #BASIC.ON
0690 STX PORTB
0700 ;
0710 LDY #1 ;Is lnno >= curr
0720 LDA FORLN+1 ; lnno???
0730 CMP (STMCUR),Y
0740 BCC NORMAL
0750 BNE FAST
0760 ;
0770 DEY
0780 LDA FORLN

```

```
0790     CMP (STMCUR),Y
0800     BCC NORMAL
0810 ;
0820 FAST
0830     LDA STMCUR+1 ;Yes,start from
0840     LDY STMCUR  ; current line
0850     RTS
0860 NORMAL
0870     LDA STMTAB+1 ;No,start from
0880     LDY STMTAB  ; first line
0890     RTS
0900 ENDCODE
```