**CONTROL STRUCTURES**

- **DO...LOOP**
  - **DO**
    - (.addr) start - 1
  - **DE**
    - -
  - **INDEX**
    - -
  - **J**
    - -
  - **JUMP**
    - -
  - **LABEL**
    - -
  - **FRAME**
    - -
  - **IF**
    - -
  - **THEN**
    - -
  - **ELSE**
    - -
  - **ENDIF**
    - -

**Terminal Input - Output**

- **PRINT**
  - -
  - **READ**
  - -
  - **INPUT**
  - -
  - **WHILE**
  - -

**Stack Manipulation**

- **DUP**
  - (n) (n + 1)
- **DROP**
  - (n) (n - 1)
- **OVER**
  - (n1 n2 n3)
- **ROT**
  - (n1 n2 n3)
  - (n1 n2 n3)
  - (n1 n2 n3)
- **OR**
  - (n 0)
- **AND**
  - (n n)
- **NOT**
  - (n ~n)

**Number Bases**

- **DECIMAL**
- **BASE**
  - (addr)

**Arithmetic and Logical**

- **ADD**
  - (n1 n2)
- **SUB**
  - (n1 n2)
  - (n1 n2)
  - (n1 n2)
- **MOD**
  - (n1 n2)
  - (n1 n2)
  - (n1 n2)
- **AND**
  - (n1 n2)
  - (n1 n2)
- **AND**
  - (n1 n2)
  - (n1 n2)
  - (n1 n2)
  - (n1 n2)
  - (n1 n2)
  - (n1 n2)
- **NOT**
  - (n ~n)

**Comparison**

- **LESS**
  - (n1)
  - (n2)
  - (n3)
  - (n4)
- **LESS**
- **GREAT**
  - (n1)
  - (n2)
  - (n3)
  - (n4)
- **EQUAL**
  - (n1)
  - (n2)
  - (n3)
  - (n4)

**Memory**

- **PRINT**
  - (addr)
  - (n)
  - (addr)
  - (n)
  - (addr)
  - (b)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
  - (addr)
HANDY REFERENCE CARD
valFORTH 1.1

Vocabularies

CONTEXT ( -- addr )
Returns address of pointer to context vocabulary
(searched first).

CURRENT ( -- addr )
Returns address of pointer to current vocabulary
(whose new definitions are put).

FORTH ( -- )
Main FORTH vocabulary (execution of FORTH sets
CONTEXT vocabulary).

EDITOR ( -- )
Editor vocabulary; sets CONTEXT.

ASSEMBLER ( -- )
Assembler vocabulary; sets CONTEXT.

VOCABULARY ( -- )
Sets CURRENT vocabulary to CONTEXT.

XXX ( -- )
Create new vocabulary named xxx.

Miscellaneous and System

( ( -- )
Begin comment, terminated by right paren on same
line, space after .

FORGET xxx ( -- )
Forget all definitions back to and including xxx.

ABORT ( -- )
Error termination of operation.

HERE ( -- addr )
Find the address of xxx in the dictionary; if used
in definition, compile address.

PAD ( -- addr )
Returns address of next unused byte in the
dictionary.

IN ( -- addr )
Returns address of scratch area [usually 128 bytes
beyond HERE].

OUT ( -- addr )
System variable containing offset into input buffer.

GET, e.g., by WORD.

OVER ( -- addr )
Returns address of top stack item.

SWAP ( -- )
Leave a gap of n bytes in the dictionary.

Compile a number into the dictionary.

SOFTWARE AND DOCUMENTATION

©Copyright 1982
Valper International

Aller is a trademark of Aller Inc., a division of Warner Communications.
DEBUGGING UTILITIES

DECOMP xxx
COMPP ( addr -- )
"two" ( addr -- )
FREE ( -- )
STACK ( -- Flag -- )
S ( -- ... -- )
U.S. ( -- )
BT ( -- )
CALL ( -- Flag (executing) -- )
MS ( -- )

FLOATING POINT

VARINFOM ( xx:ff -- )
VARINFOM ( xx:addr -- )
FIPS ( fp1 <-- fp2, fp1 <-- fp2 )
FIPSLOCATE ( -- )
FERVER ( fp1 <-- fp2, fp1 <-- fp2, fp2 <-- fp1 )
FLOATING ( -- )
FP ( -- )
FP ( -- )
FP ( -- )
FP ( -- )
FLOAT ( n -- fp )
FIX ( fp (non-nag, less then 37256.5) -- n )
LOG ( fp1 <-- fp2 )
LOG10 ( fp1 <-- fp2 )
EXP ( fp1 <-- fp2 )
EXP10 ( fp1 <-- fp2 )

OPERATING SYSTEM

OPEN ( addr :m n :n2 -- n3 )
CLOSE ( n -- )
PUT ( b1 < b2 )
GET ( b1 < b2 )
GETREC ( addr n1 :n2 <-- n3 )
PUTREC ( addr n1 :n2 <-- n3 )
RESTART ( n -- )
RESTART ( n -- b1 b2 b3 )
SPECIAL ( b1 b2 b3 b4 b5 b6 b7 b8 b9 )
RES232 ( -- )

This word opens the device whose name is at addr. The device is opened on channel n and is available for input or output.

The device status byte is returned as n3.

Outputs byte b1 on channel n, returns status byte b2.

Gets byte b1 from channel n, returns status byte b2.

Inputs record from channel n up to length n1.

Outputs status n1 bytes from channel n.

Returns status byte b from channel n.

Opens channel with device status byte b1 and b2 and returns normal status byte b3.

Ordinarily, the device is opened by the word "OPEN" in the System "Special" command. AUS1 through AUS6 are b1 through b6 respectively. The device status byte is placed in the channel number b7. Returns status byte b9.

Opens the AR-I 800 drivers into the dictionary (approx 16k).
valFORTH 6502 Assembler

ASSMELER ( --- )

CODE xxx ( --- )

Cl ( --- )

END-CODE ( --- )

SUBROUTINE xxx ( --- )

LCODE ( --- )

Control Structures

IF, ( flag --- addr 2 )

ELSE, ( addr 2 --- addr 3 )

ENDIF, ( addr 2/3 --- )

BEGIN, ( --- addr )

UNTIL, ( addr 1 flag --- )

WHILE, ( addr 1 flag --- addr 4 )

REPEAT, ( addr 4 --- )

AGAIN, ( addr 1 --- )

Parameter Passing (These routines must be jumped to.)

NEXT ( --- addr )

PUSH ( --- addr )

PUSHA ( --- addr )

PUT ( --- addr )

PUTA ( --- addr )

BINARY ( --- addr )

POP and POPTH

POP ( --- addr )

POPND ( --- addr )

SETUP ( --- addr )

N ( --- addr )

Opcodes (various --- various)

Aliases

NEXT, = NEXT JMP, POPZ, = POP2 JMP, POP, = POP JMP, PSW, = PUSH JMP, CL, = WSAVE LDW, PUT, = PUT JMP, PUSHA, = PUSHA JMP, PUTA, = PUTA JMP, POP, = POP JMP, END, = UNTIL

Software and Documentation ©Copyright 1982 Valpar International
HANDY REFERENCE CARD

vlFORTH

SOFTWARE SYSTEM

EDITOR 1.1 COMMAND SUMMARY

Below is a quick reference list of all the commands which the video editor recognizes.

Enter the Edit Mode (executed outside of the edit mode)

V       (scrn---)  #Enter the edit mode and view the specified screen.
L       (---)     #Review the current screen.
WHERE    (---)     #Enter the edit mode and position the cursor over
                  #the word that caused a compilation error.
LOCATE   cccc  (---) #Enter the edit mode and position the cursor over
                      #the word "ccc" where it is defined.
LOCATOR  (ON/OFF) #When ON allows all words compiled until the next
                   #OFF to be loadable using the LOCATE command above.
                   #Sets the length (in lines) of the storage buffer.
                   #The default is five.

Cursor Movement (issued within the edit mode)

ctrl ↑  #Move cursor up one line, wrapping to the bottom line
        #if moved off the top.
ctrl ↓  #Move cursor down one line, wrapping to the top
        #line if moved off the bottom.
ctrl ←  #Move cursor left one character, wrapping to the
        #right edge if moved off the left.
ctrl →  #Move cursor right one character, wrapping to the
        #left edge if moved off the right.
RETURN  #Position the cursor at the beginning of the next
        #line.
TAB     #Advance to next tabular column.

Editing Commands (issued within the edit mode)

ctrl INS  #Insert one blank at cursor location, losing the
           #last character on the line.
ctrl DEL  #Delete character under cursor, closing the line.
shift INS #Insert blank line above current line, losing the
           #last line on the screen.
shift DEL #Delete current cursor line, closing the screen.
ctrl I    #Toggle insert mode on and off.
           #(see full description of ctrl-[).  
BACKS    #Delete last character typed, if on the same line
           #as the cursor.
ctrl H    #Erase to end of line (backspace).

Buffer Management (issued within the edit mode)

ctrl T    #Delete current cursor line sending
           #it to the edit buffer for later use.
ctrl F    #Take the current buffer line and insert it
           #above the current cursor line.
ctrl K    #Copy current cursor line sending it to the
           #edit buffer for later use.
ctrl U    #Take the current* buffer line and copy it
           #to the current cursor line.
ctrl R    #Roll the buffer making the current buffer
           #line current.
ctrl B    #Roll the buffer backwards making the fourth
           #buffer line on the current screen.
ctrl C    #Clear the current* buffer line and performs
           #a ctrl-D.

*Note: The current buffer line is bottommost on the video display.

Changing Screens (issued within the edit mode)

ctrl P    #Display the previous screen saving all
           #changes made to the current screen.
ctrl N    #Display the next screen saving all changes
           #made to the current screen.
ctrl S    #Save changes made to the current screen
           #and exit the screen.
ctrl Q    #Quit the edit session forgetting all changes
           #made to the current screen.

Special Keys (issued within the edit mode)

ESC      #Do not interpret the next key typed as any
           #of the commands above. Send it directly to
           #the screen instead.
ctrl A    #Put the arrow "->" ("next screen") in the
           #lower-right-hand corner of the screen unless
           #it is already there, in which case remove it.
ctrl J    #Split the current line into two lines at the point
           #where the cursor is.
ctrl D    #Corrects any major editing blunders.

Screen Management (executed outside of the edit mode)

#FLUSH    (---)  #Save any updated FORTH screens to disk.
#EMPTY    (---)  #Forget any changes made to any screens not yet
                   #flushed to disk.
COPY      (from to--)  #Copies screens from to to screen #.
CLEARS    (scrn--)    #Blanks all specified screens.
#CLEARS   (scrn--)    #Blank fills specified number of screens
                   #starting with screen #.
#SMOVE    (from to scrn--)  #Duplicates the specified number of screens
                   #Starting with screen number "from".
                   #Note: Screen 0 is the current screen.
                   #Screen 1 is the next screen.
                   #Screen 2 is the next after next screen.

#EDITOR 1.0 COMMAND

Software and Documentation ©Copyright 1982
Valper International