

# ZCALL FUNCTIONS

The following functions are intended for use with Eason's WinBuild and DOSBuild program generation products. Use in programs, which are customer generated, is **NOT RECOMMENDED**. All of the calls below have optional parameters and minimal error checking. WinBuild and DOSBuild are aware of the restrictions of these calls and do not expose them to extraneous ranges or extra parameters. Use extreme care when using these commands to insure that parameters are correctly specified.

---

## CALL ZA

---

Put normal/large/huge text or number with reverse as an option

Usage:

CALL ZA(Size, Row, Column, String [, Number ])

Used for all PUT TEXT and PUT NUMBER pseudocodes.

Size:           0 - Normal Text  
                 1 - Large Text  
                 2 - Huge Text  
                 4 - Reverse Normal  
                 5 - Reverse Large  
                 6 - Reverse Huge

Row: 1-8

Column: 1-40

String:

In the absence of the Number parameter, the String is printed as a literal.

With the Number parameter, the String is a format string as per the following format string specification:

var can be any valid numeric value or expression.

format string consists of a string up to 20 characters long.

It can be either a constant or a variable. The programmer can select from the following options:

# reserves space for a digit.

+ or - allows the operator to enter a sign.

. allows the operator to enter a decimal point.

e or E allows the operator to enter an exponent

Number:

Any valid BASIC numeric variable.

---

# CALL ZB

---

Set up program - called on program initiation

Usage:

CALL ZB()

Sets up and clears hotkey flags and softkey flags.

---

# CALL ZC

---

Initialize a screen

Usage:

CALL ZC( Type )

Type: 1 - Normal Screen, clears graphic and text screens  
2 - Quick Screen, clears only text screen

This function clears the screen according to the parameter list, turns off all soft keys and restores any hot keys that are active. It also clears the keyboard buffer.

---

# CALL ZD

---

Display hot key text

Usage:

CALL ZD( Keynum, OnOff[, KeyText] )

KenNum - 1 thru 10 key number to vector on

OnOff - 1 turn on key text (use text pre-stored for key if KeyText parameter is not included).

0 turn off key text

KeyText - String variable (5 characters max) for the text to place over the key (only available on keys 1 through 6).

---

# CALL ZE

---

Place text for a softkey

Usage:

CALL ZE( Keynum% )

Keynum% - Variable for returned parameter of softkey that has been activated

- 0 means no softkey down.
- 1 - 10 means a softkey has been pressed, and this is the number of the key that is down.

---

# CALL ZF

---

Display the softkey at the desired position

Usage:

CALL ZF( KeyNum, KeyText\$ )

KeyNum - Number of the key to place text over (1 through 6) numbers exceeding 6 are ignored.

KeyText\$ - Text to place over the softkey. First 5 characters only are displayed.

---

# CALL ZG

---

Perform Communications Function

Usage:

CALL ZG( Port, Echo, text\$ [, OutNumber[, NumResponse[, StrResponse\$]]) )

Port - Specifies the serial communications port:

- 1 - Com 1
- 2 - Com 2

Echo - Specifies the serial echo control:

0 - Do not read and destroy echo

1 - Read and destroy echo

text\$ - Text string to be output on the serial port

Outnumber - If this parameter is present, concatenate the number with the text\$ parameter in a string fashion and send it out the serial port.

NumResponse - If this parameter is present, wait for a numeric response on the specified serial port. When received, translate into a number and place the numeric result into NumResponse. Leading spaces and one non-numeric character is ignored. Note that the parameter StringResponse\$ MUST NOT be present, otherwise unpredictable results may occur.

StringResponse\$ - If this parameter is present, wait for a string response on the specified serial port. When received, the characters will be placed into the specified StringResponse\$ variable. Note that the parameter NumResponse MUST NOT be present, otherwise unpredictable results may occur.

---

## CALL ZH

---

Get Number setup

Usage:

CALL ZH( Row, Column, Format\$, Video%, Default )

Must be used with the CALL NKEY function as follows:

```
CALL ZH(y,x,fmt$,ZBLDVID%,default)
CALL NKEY(ZBLDN$,ZBLDN%):IF ZBLDN%<13THEN%%0
CALL ZI():number=VAL(ZBLDN$)
```

Row - 1-8

Column - 1-40

Format\$: - A format string as per the following format string specification:

var can be any valid numeric value or expression.  
format string consists of a string up to 20 characters long.  
It can be either a constant or a variable. The programmer can select from the following options:

# reserves space for a digit.

+ or - allows the operator to enter a sign.

. allows the operator to enter a decimal point.  
e or E allows the operator to enter an exponent

Video% - 0 specifies no background  
1 specifies reverse video  
2 specifies box

Default - Any valid BASIC numeric variable to be used as the initial  
or default number.

See also:

CALL NKEY()  
CALL ZI()

---

## CALL ZI

---

Post numeric entry cleanup

Usage:

CALL ZI()

This function clears the reverse video behind the numeric entry  
(even if none were selected!). It retains the cursor and length  
information used by the previous CALL ZH or CALL ZJ instruction.  
DO NOT USE this command without first performing either of these  
functions.

---

## CALL ZJ

---

Get String Setup

Usage:

CALL ZJ( Row, Column, Video%, String\$ )

Must be used with the CALL SINPUT function as follows:

```
ZBLDS%=1:ZBLDS$=default$:CALL ZJ(y,x,ZBLDVID%,ZBLDS$)
CALL SINPUT(ZBLDS$,ZBLDS%):IF ZBLDS%<>13THEN ZBLDS%=0:GOTO%%0
CALL ZI():text$=ZBLDS$
```

Row - 1-8

Column - 1-40

Video% - 0 specifies no background  
          1 specifies reverse video  
          2 specifies box

String\$ - String\$ is the default string used by the CALL SINPUT function  
          Note that the CALL ZJ command uses the String\$ parameter to  
          determine the length of the string input area.

See also:

CALL SINPUT  
CALL ZI

---

## CALL ZK

---

Bar graph set up function #1

Usage:

CALL ZK( Uleftx, Ulefty, Lrightx, Lrighty ,Orient )

Uleftx - The upper left X position (0-239) for the graph

Ulefty - The upper left Y position (0-63) for the graph

Lrightx - The lower right X position (0-239) for the graph

Lrighty - The lower right Y position (0-63) for the graph

Orient - The orientation: 0 for horizontal, 1 for vertical

Warning: omitting any of these parameters may have unpredictable results.

---

## CALL ZL

---

Bar graph set up function #2

Usage:

CALL ZL(Lowlim,Uplim)

Lowlim - The minimum limit for the graph data

Uplim - The upper limit for the graph data

Warning: CALL ZK MUST have been executed prior to calling CALL ZL.  
Omitting any of these parameters may have unpredictable results.

Note this function located in bank 0 to utilize the variable scaling floating point routines.

---

## CALL ZM

---

Display Bar Graph Data

Usage:

CALL ZM(Data)

Data - Variable to bargraph

Warning: CALL ZK and CALL ZL must have been executed prior to calling CALL ZM.

---

## CALL ZN

---

Update an I/O bit on the selected I/O BANK

Usage:

CALL ZN(Number,Bitval)

Number - The bit position number (0 thru 23, or a variable)

Bitval - The bit value (0 or 1, or a variable)

THIS PAGE INTENTIONALLY BLANK