

BASIC listing: WINDAB.BAS

```
*****
' *                               W I N D A B . B A S                               *
'-----
' * Task                          : Determines whether Windows is active, and if so, *
' *                               the operating mode.                               *
' *                               QuickBASIC and the QB.LIB must be loaded using    *
' *                               QB /L QB                                           *
' *                               before loading and running this file.             *
'-----
' * Author                        : Michael Tischer                               *
' * Developed on                  : 08/22/91                                         *
' * Last update                   : 04/07/95                                         *
'*****
DECLARE FUNCTION Windows% (MVersion AS INTEGER, SVersion AS INTEGER)

'$INCLUDE: 'Qb.bi'                                'Include file for interrupt calls

CONST MULTIPLEX = &H2F                            'Interrupt number of multiplex interrupt
CONST NOWIN = &H0                                  'Windows not active
CONST WIN386X = &H1                                'Windows/386 V2.x running
CONST WINREAL = &H81                               'Windows running in real mode
CONST WINSTANDARD = &H82                           'Windows running in standard mode
CONST WINENHANCED = &H83                           'Windows running in extended mode

DIM WindowsActive AS INTEGER                       'Windows mode
DIM MVer AS INTEGER                               'Main version of Windows
DIM SVer AS INTEGER                               'Alternate version of Windows

PRINT " WINDAB - (c) 1991 by Michael Tischer "
PRINT                                             'Blank line
WindowsActive = Windows(MVer, SVer)              'Get Windows version and mode
```

```

SELECT CASE WindowsActive
  CASE NOWIN
    PRINT "Windows not active "
  CASE WIN2X
    PRINT "Windows /386 V 2.x active "
  CASE WINREAL
    PRINT "Windows active in real mode "
  CASE WINSTANDARD
    PRINT "Windows active in standard mode "
  CASE WINENHANCED
    PRINT "Windows V"; LTRIM$(STR$(MVer)); ".";
    PRINT LTRIM$(STR$(SVer)); " active in extended mode"
END SELECT

```

```

'*****
'* Windows : Determines whether Windows is active *
'* Input   : MVERSION = Integer variable of main version number *
'*          : SVERSION = Integer variable of sub version number *
'* Output  : Windows status, from constants NOWIN, WIN386X, WINREAL, *
'*          : WINSTANDARD or WINENHANCED *
'* Info    : Version number can only be passed and returned when *
'*          : Windows 3.x is operating in enhanced mode *
'*****

```

```

FUNCTION Windows% (MVersion AS INTEGER, SVersion AS INTEGER)

```

```

  DIM Regs AS RegTypeX          'Processor registers for interrupt call
  DIM VBf AS INTEGER            'Version buffer

```

```

  MVersion = 0                  'Initialize version numbers
  SVersion = 0

```

```

  Regs.ax = &H1600             'Function number: Install test for extended mode

```

```

CALL INTERRUPTX(MULTIPLEX, Regs, Regs)
VBf = Regs.ax                                'Set regs.AX

SELECT CASE VBf MOD 256                        'Compute low byte

'---- Windows /386 running -----

CASE &H1, &HFF                                'Windows /386 running
    MVersion = 2                               'Main version
    SVersion = 0                               'Sub version unknown
    Windows = WIN386X

'---- Windows not running, running in real mode or standard mode -----

CASE &H0, &H80
    Regs.ax = &H4680                            'Identify real or standard mode
    CALL INTERRUPTX(MULTIPLEX, Regs, Regs)
    IF (Regs.ax MOD 256) = &H80 THEN                'Is Windows running?
        Windows = NOWIN                            'No
    ELSE                                            'Windows running in real or standard mode
        Regs.ax = &H1605                            'Emulate installation of DOS Extender
        Regs.bx = &H0
        Regs.si = &H0
        Regs.cx = &H0
        Regs.es = &H0
        Regs.ds = &H0
        Regs.dx = &H1
        CALL INTERRUPTX(MULTIPLEX, Regs, Regs)
        IF Regs.cx = &H0 THEN                        'Windows in real mode?
            Windows = WINREAL                        'Yes
        ELSE                                        'No --> Windows runs in standard mode
            Windows = WINSTANDARD

```

```
END IF  
END IF
```

```
'---- Windows in extended mode, AX contains version number -----
```

```
CASE ELSE  
  MVersion = VBf AND &HF          'Low byte is main version  
  SVersion = VBf \ 256            'High byte is sub version  
  Windows = WINENHANCED  
END SELECT  
  
END FUNCTION
```