Defining the Window

To understand how stylebits can modify a window, it's important to first understand what the traditional window looks like and how it behaves. Generally speaking, a window is distinguishable from the rest of the screen by its borders. The borders create a box.

```
Open "My Window" for Window as #1
Print #1, "Trapclose [closeWindow]"

Wait

[closeWindow]
Close #1
End
```

Types of Windows

The Liberty BASIC Help File identifies these four types of windows created using native LB commands:

WINDOW - the most common and the most useful, used to hold controls, can include one or more menus, can trap TAB key to move focus from one control to another.

GRAPHICS - designed to display graphics and sprites, not intended to hold controls, can trap mouse clicks and character key presses.

DIALOG - similar to the traditional window, can contain controls, cannot contain menus, can trap ENTER key as a special button to direct action. When displayed as "modal," the dialog window remain in focus until closed. (The prompt, notice and confirm windows are special dialog modal windows.

TEXT - Limited to writing and editing text, always contain a menubar that contains a read-made File Menu and a ready-made Edit Menu. (The Mainwindow is a Text Window.

There are many native variations of these four windows. See the Helpfile for variations such as full screen (_fs), _nf (no sizing frame), no scroll bars (_nsb), no titlebar (_popup), among others.

Elements of a Window

For an explanation of the parts of a window, visit this tutorial.

As well as the borders of the window box, the typical window contains a blue title bar along its top. The windows constant for the border is WS_BORDER and for the title bar is WS_CAPTION. The window icon, the small picture located in the left portion of the title bar is referred to as WS_SYSMENU. Double-

clicking this icon reveals the window menu (Restore, Move, Size, Minimize, Maximize, Close). In addition, there are three buttons located in the right portion of the title bar. They are

- the minimize button or WS_MINIMIZEBOX
- the maximize button or WS_MAXIMIZEBOX
- the trapclose button.

Using Window Styles Constants in Your Program

Window styling changes have long been available in Liberty BASIC using the API calls **SendMessageLong**. Liberty BASIC v4.x offers the programmer the opportunity to modify these styles using the native command STYLEBITS. Stylebits commands are used to access or change the style and attributes of a window and other controls. There are two categories of styles: standard and extended.

Extended attributes resulted from newer CreateWindwoEX functions so they are more useful in adding qualities rather than removing qualities. Both categories are represented as flags in 32 - bit integers. These styles must be applied to the window and / or controls before the gui is opened.

The stylebits of any particular window and the controls of that window may not be altered after the window is opened except, in some but not all cases, with API calls.

From the Liberty BASIC v4.01 Help File
stylebits #handle, addBits, removeBits, addExtendedBits, removeExtendedBits

STYLEBITS allows you to change the style of a Liberty BASIC window or control. It accepts a handle and four parameters. When the window is opened it checks to see if there are style bits for the window or for any controls. If there is a STYLEBITS command it applies the remove bits first, then applies the add bits. In this way the control is created from the get-go with the desired style. The STYLEBITS command must be issued before the command to open the window.

Most often you can recognize whether a stylebit is standard (dwStyle) or extended (dwExStyle) by its Window Constant. The stylebit WS_CAPTION can be passed as the first (addBits) or second (removeBits) parameter, while the stylebit WS_EX_TOPMOST is passed as either the third (addExtendedBits) or fourth (removeExtendedBits) parameter. Liberty BASIC recognizes window constants by placing an additional underscore immediately in front of the constant, e.g, _WS_CAPTION. Run this demo to see how window attributes can be both added and removed.

```
    'This Demo opens up to five independent windows.  The windows

'will be placed side by side for visual comparison.  The left (primary
)
    'window is opened 'as is' with no Stylebits assigned.  The
    'right windows show windows with attributes either added
```

```
    'or removed by the use of the Stylebits command.
    'This demo requires Liberty BASIC v4.x or greater.

    Nomainwin
    If Val(Version$) < 4 Then
        Notice "This program requires Liberty BASIC V4.0 or greater"
        End
    End If

    'The main (left) window - No stylebits assigned
    WindowWidth = 300
    WindowHeight = 350
    UpperLeftX = 50
    UpperLeftY = 100

    Statictext #main.0a,
"No Stylebits commands have been used in this window", 20, 10, 260, 40
    Statictext #main.0b,
"Click on any
of the buttons to see Stylebits in action", 20, 50, 260, 40
    Button #main.1,
"addBits: Vertical Scroll Bar", [parameter1], UL, 15, 120, 270, 30
    Button #main.2,
"removeBits: Maximize Button", [parameter2], UL, 15, 160, 270, 30
    Button #main.3,
"addExtendedBits: Sunken Edge Bor
der", [parameter3], UL, 15, 200, 270, 30
    Button #main.4,
"removeExtendedBits: No Demo", [parameter4], UL, 15, 240, 270, 30
    Button #main.5,
"See All 3 Stylebits Commands", [parameterAll], UL, 15, 280, 270, 30

    Open "The Window - AS IS" for Window as #main
    Print #main, "Trapclose [endDemo]"
    Print #main, "Font Times_New_Roman 12 Bold"

    Wait

[parameter1]
    'addBits: Add a vertical scroll bar to the window
    WindowWidth = 300
    WindowHeight = 350
    UpperLeftX = 400
    UpperLeftY = 100

    Stylebits #1, _WS_VSCROLL, 0, 0, 0
```

```
    Statictext #1.0a,
"A
regula
r window doe
s not normally include a vertical scroll bar.", 20, 10, 260, 40
    Statictext #1.0b,
"The Stylebits
(1st parameter) has ADDED this feature.", 20, 50, 260, 40
    Statictext #1.1, "addBits: Vertical Scroll Bar", 15, 120, 270, 20
    Statictext #1.2,
"Stylebits #1, _WS_VSCROLL, 0, 0, 0", 15, 160, 270, 40
    Button #1.3, "Close", [close1], UL, 120, 250

    Open "addBits: Vertical Scroll Bar" for Window as #1
    Print #1, "Trapclose [close1]"
    Print #1, "Font Times_New_Roman 12 Bold"
    window1 = 1

    Wait

[close1]
    Close #1
    window1 = 0
    Wait

[parameter2]
    'removeBits: Remove the maximize button
    WindowWidth = 300
    WindowHeight = 350
    UpperLeftX = 400
    UpperLeftY = 100

    Stylebits #2, 0, _WS_MAXIMIZEBOX, 0, 0

    Statictext #2.0a,
"A regular window contains a maximize button", 20, 10, 260, 40
    Statictext #2.0b,
"The Stylebit
s (2nd parameter) has REMOVED this feature.", 20, 50, 260, 40
    Statictext #2.1, "removeBits: Maximize Window", 15, 120, 270, 20
    Statictext #2.2,
"Stylebits #2, 0, _WS_MAXIMIZEBOX, 0, 0", 15, 160, 270, 40
    Button #2.3, "Close", [close2], UL, 120, 250

    Open "removeBits: Maximize Window" for Window as #2
```

```
    Print #2, "Trapclose [close2]"
    Print #2, "Font Times_New_Roman 12 Bold"
    window2 = 1

    Wait

[close2]
    Close #2
    window2 = 0
    Wait

[parameter3]
    'addExtendedBits: Add Sunken Edge Border
    WindowWidth = 300
    WindowHeight = 350
    UpperLeftX = 400
    UpperLeftY = 100

    Stylebits #3, 0, 0, _WS_EX_CLIENTEDGE, 0

    Statictext #3.0a,
"A regular window has smooth resizable edges.", 20, 10, 260, 40
    Statictext #3.0b,
"The Stylebits (
3rd parameter) has altered the edges.", 20, 50, 260, 40
    Statictext #3.1,
"addExtendedBits: Sunken Edge Border", 15, 120, 270, 20
    Statictext #3.2,
"Stylebits #3, 0, 0, _WS_EX_CLIENTEDGE, 0", 15, 160, 270, 40
    Button #3.3, "Close", [close3], UL, 120, 250

    Open "addExtendedBits: Sunken Edge Border" for Window as #3
    Print #3, "Trapclose [close3]"
    Print #3, "Font Times_New_Roman 12 Bold"
    window3 = 1

    Wait

[close3]
    Close #3
    window3 = 0
    Wait


[parameter4]
    'removeExtendedBits: No Demo
```

```
    WindowWidth = 300
    WindowHeight = 350
    UpperLeftX = 400
    UpperLeftY = 100

    Stylebits #4, 0, 0, 0, 0 'No Extended Bits to remove

    Statictext #4.0a,
"The extended bits are generally added.", 20, 10, 260, 40
    Statictext #4.0b,
"There is no demo for removing extended bits.", 20, 50, 260, 40
    Statictext #4.1, "addExtendedBits: No Demo", 15, 120, 270, 20
    Statictext #4.2, "Stylebits #3, 0, 0, 0, 0", 15, 160, 270, 40
    Button #4.3, "Close", [close4], UL, 120, 250

    Open "removeExtendedBits: No Changes" for Window as #4
    Print #4, "Trapclose [close4]"
    Print #4, "Font Times_New_Roman 12 Bold"
    window4 = 1

    Wait

[close4]
    Close #4
    window4 = 0
    Wait

[parameterAll]
    'All 3 Stylebits Combined in the Same Window
    WindowWidth = 300
    WindowHeight = 350
    UpperLeftX = 400
    UpperLeftY = 100

    Stylebits #5, _WS_VSCROLL, _WS_MAXIMIZEBOX, _WS_EX_CLIENTEDGE, 0

    Statictext #5.0a,
"This window has all 3 stylebits applied.", 20, 10, 260, 40
    Statictext #5.0b,
"No removeExtendedBits has been applied.", 20, 50, 260, 40
    Statictext #5.1,
"Stylebits: #1, #2, and #3 combined", 15, 120, 270, 20
    Statictext #5.2,
"Sty
lebits #
5, _WS_VSCROLL,
```

```
_WS_MAXIMIZEBOX, _WS_EX_CLIENTEDGE, 0", 15, 160, 270, 60
    Button #5.3, "Close", [close5], UL, 120, 250

    Open "removeExtendedBits: No Changes" for Window as #5
    Print #5, "Trapclose [close5]"
    Print #5, "Font Times_New_Roman 12 Bold"
    window5 = 1

    Wait

[close5]
    Close #5
    window5 = 0
    Wait

[endDemo]
    If window1 = 1 Then Close #1
    If window2 = 1 Then Close #2
    If window3 = 1 Then Close #3
    If window4 = 1 Then Close #4
    If window5 = 1 Then Close #5
    Close #main
    End
```

It is sometimes desirable to add two or more stylebit parameters. The window constants can be combined using the logical boolean OR conjunction. This next demo first opens a normal window, then opens a second window that

- cannot be maximized: uses the removeBits (2nd parameter) _WS_MAXIMIZEBOX
- cannot be minimized: uses the removeBits (2nd parameter) _WS_MINIMIZEBOX
- has sunken edge borders: uses the addExtendedBits (3rd parameter) _WS_EX_CLIENTEDGE
- stays on top even when not active window: uses the addExtendedBits (3rd parameter) _WS_EX_TOPMOST

```
    'This Demo opens up to two independent windows.  The windows

'will be placed side by side for visual comparison.  The left (primary
)
    'window is opened 'as is' with no Stylebits assigned.  The
    'right window demonstrates

'cannot be maximized: uses the removeBits (2nd parameter) _WS_MAXIMIZE
BOX

'cannot be minimized: uses the removeBits (2nd parameter) _WS_MINIMIZE
```

```
BOX

'has sunken edge borders: uses the addExtendedBits (3rd parameter) _WS
_EX_CLIENTEDGE

'stays on top even when not active window: uses the addExtendedBits (3
rd parameter) _WS_EX_TOPMOST
    'This demo requires Liberty BASIC v4.x or greater.

    Nomainwin
    If Val(Version$) < 4 Then
        Notice "This program requires Liberty BASIC V4.0 or greater"
        End
    End If

[noStylebits]
    WindowWidth = 300
    WindowHeight = 350
    UpperLeftX = 50
    UpperLeftY = 100

    Statictext #1.1,
"No Stylebits commands have been used in this window", 20, 10, 260, 40
    Statictext #1.2,
"
Dr
ag t
his wind
ow 'over' the se
cond window to see the effects of the", 20, 110, 260,40
    Statictext #1.3,
"_WS_EX_TOPMOST stylebits command", 20, 150, 260, 40

    Open "The Window - AS IS" for Window as #1
    Print #1, "Trapclose [close1]"
    Print #1, "Font Times_New_Roman 12 Bold"
    window1 = 1

[multipleStylebits]
    WindowWidth = 300
    WindowHeight = 350
    UpperLeftX = 400
    UpperLeftY = 100

    Stylebits #2, 0, _WS_MAXIMIZE
BOX or _WS_MINIMIZEBOX, _WS_EX_CLIENTEDGE or _WS_EX_TOPMOST, 0
```

```
    sbText$ =
"_WS_MAXIMIZEBOX or _WS_MINIMIZEBOX, _WS_EX_CLIENTEDGE or _WS_EX_TOPMO
ST, 0"
    Statictext #2.1,
"Try to Maximize or Minimize this Window", 20, 10, 260, 40
    Statictext #2.2,
"Click and drag this window around.", 20, 50, 260, 40
    Statictext #2.3,
"Notice the sunken border edges.", 20, 90, 260, 40
    Statictext #2.4, "Stylebits #2, ";sbText$, 15, 160, 260, 80
    Statictext #2.5,
"2 or more parameters joined with OR", 15, 260, 260, 40

    Open "Combining Stylebits Parameters Using OR" for Window as #2
    Print #2, "Trapclose [close2]"
    Print #2, "Font Times_New_Roman 12 Bold"
    window2 = 1

    Wait

[close1]
    Close #1
    window1 = 0
    If window2 = 0 Then End
    Wait

[close2]
    Close #2
    window2 = 0
    If window1 = 0 Then End
    Wait
```

Stylebits, of both dwStyle and dwExStyle, can be assigned to other types of windows and most controls (buttons, textboxes, listboxes, statictext, etc.). Texteditors are not readily modifiable with Stylebits. The helpfile states that *since the texteditor is not a native Windows control you will only be able to do things like tweak its border and perhaps a few other things.*

A List of Stylebits

You can get a list of all dwStyles and dwExStyles available with the Stylebits command at the MSDN Library.