BTICN$ - Set GX Button Icon

The BTICN$ routine is used to set, or update, the icon associated with a GX "Icon Button".

1. Invocation

To set the icon for an Icon Button code:

```
CALL BTICN$ USING ic icon [hover]
```

where *icon* is a LOW-VALUES terminated string which specifies the name of the icon file; *hover* is an optional LOW-VALUES terminated string which specifies the hover text associated with the icon. For GSM SP-29, and earlier, *ic* is a control block of the following format:

```
01    IC
  02  ICFTN       PIC 9(4) COMP          * Button function number (1 to 255)
  02  ICINX       PIC 9(2) COMP          * Button index, this would
                                         * normally be set to 1 unless
                                         * there is more than one button
                                         * with a particular function number
                                         * For GSM SP-19, and higher, the top
                                         * bit of this field is used as a
                                         * flag (see below)
  02  ICID4C                             * 4 character window-id for use when
                                         * BTICN$ is called before the window
                                         * is displayed on the screen
   03 ICID        PIC X(2)               * 2 character Window-id for use when
                                         * BTICN$ is called before the window
                                         * is displayed on the screen
                                         * This field must be set to SPACES
                                         * to indicate the current window
   03 FILLER      PIC X(2)               * Must be set to spaces
  02  ICACC       PIC X                  * Accelerator character for the
                                         * button text
                                         * This field must always be spaces
                                         * when BTICN$ is called before a
                                         * window is displayed otherwise, if
                                         * set, it will override the
                                         * accelerator defined in the text of
                                         * the BTN label
```

For GSM SP-30, and later, *ic* has been extended to the following format:

```
01    IC
  02  ICFLAG      PIC 9(2) COMP          * Extension flag
                                         * 0 = No ICEXTN present
                                         * 1 = ICEXTN extension present
                                         * All other values reserved
  02  ICFTN       PIC 9(2) COMP          * Button function number
  02  ICINX       PIC 9(2) COMP          * Button index, this would
                                         * normally be set to 1 unless
                                         * there is more than one button
                                         * with a particular function number
                                         * For GSM SP-19, and higher, the top
                                         * bit of this field is used as a
                                         * flag (see below)
  02  ICID4C                             * 4 character window-id for use when
                                         * BTICN$ is called before the window
                                         * is displayed on the screen
   03 ICID        PIC X(2)               * 2 character Window-id for use when
                                         * BTICN$ is called before the window
                                         * is displayed on the screen
                                         * This field must be set to SPACES
                                         * to indicate the current window
   03 FILLER      PIC X(2)               * Must be set to spaces
  02  ICACC       PIC X                  * Accelerator character for the
                                         * button text
                                         * This field must always be spaces
```

```
                                               * when BTICN$ is called before a
                                               * window is displayed otherwise, if
                                               * set, it will override the
                                               * accelerator defined in the text of
                                               * the BTN label
        02  ICEXTN
         03 ICVERS      PIC 9(4) COMP          * Version number must be 1
            VALUE       1
         03 ICMODE      PIC 9(2) COMP          * 0 = The window-ID represents a
                                               * window that is about to be created
                                               * 1 = The window-ID represents an
                                               * existing window
```

For GSM SP-32, and later, *ic* has been further extended to the following format:

```
      01    IC
        02  ICFLAG      PIC X                  * Bit flag:
                                               * #01 0 = No ICEXTN present
                                               * #01 1 = ICEXTN extension present
                                               * #02 0 = Normal button
                                               * #02 1 = Interrupt button
                                               * #04 Reserved for future use
                                               * #08 Reserved for future use
                                               * #10 Reserved for future use
                                               * #20 Reserved for future use
                                               * #40 Reserved for future use
                                               * #80 Reserved for future use
        02  ICFTN       PIC 9(2) COMP          * Button function number
        02  ICINX       PIC 9(2) COMP          * Button index, this would
                                               * normally be set to 1 unless
                                               * there is more than one button
                                               * with a particular function number
                                               * For GSM SP-19, and higher, the top
                                               * bit of this field is used as a
                                               * flag (see below)
        02  ICID4C                             * 4 character window-id for use when
                                               * BTICN$ is called before the window
                                               * is displayed on the screen
         03 ICID        PIC X(2)               * 2 character Window-id for use when
                                               * BTICN$ is called before the window
                                               * is displayed on the screen
                                               * This field must be set to SPACES
                                               * to indicate the current window
         03 FILLER      PIC X(2)               * Must be set to spaces
        02  ICACC       PIC X                  * Accelerator character for the
                                               * button text
                                               * This field must always be spaces
                                               * when BTICN$ is called before a
                                               * window is displayed otherwise, if
                                               * set, it will override the
                                               * accelerator defined in the text of
                                               * the BTN label
        02  ICEXTN
         03 ICVERS      PIC 9(4) COMP          * Version number must be 1
            VALUE       1
         03 ICMODE      PIC 9(2) COMP          * 0 = The window-ID represents a
                                               * window that is about to be created
                                               * 1 = The window-ID represents an
                                               * existing window
```

2. STOP Codes and Exception Conditions

The following STOP codes may be generated by BTICN$:

STOP code	**Description**

23946	BTICN$ has been called by an application that is not running on GX.
23947	The total length of the GX command block has exceeded an internal limit.
23948	An attempt has been made to call BTICN$ on an incompatible version of GX. The version of GX must be V2.8g, or later
23949	The window specified is not currently displayed (for GSM SP-19, and later, this STOP code will only be generated by the synchronous version of BTICN$ - see below)
23950	The button specified is not found (for GSM SP-19, and later, this STOP code will only be generated by the synchronous version of BTICN$ - see below)
23951	The Accelerator letter is already in use (for GSM SP-19, and later, this STOP code will only be generated by the synchronous version of BTICN$ - see below)
23952	BTVERS does not contain 1.

The following exception conditions may be returned by BTICN$:

EXIT code	**$$COND**	**Description**
23926	26	Insufficient memory to complete the operation
23927	27	The Icon File is open by an another process
23928	28	The Icon file could not be found on the PC running GX (for GSM SP-19, and later, this exception will only be returned by the synchronous version of BTICN$ - see below)

3. Programming Notes

BTICN$ is only available when running on GX. Any attempt to use BTICN$ on a non-GX terminal will result in a STOP code. The version of GX must be V2.8g, or later. The version of GSM must be GSM SP-9, or later. The optional hover text is only support with GSM SP-18, or later.

An Icon Button is defined as a normal Speedbase button (i.e. defined by the BTN option) with button text of the following format:

"~I*awwwddd*"

where:

I	Icon Button indicator
a	accelerator character
www	Button width (in pixels)
ddd	Button depth (in pixels)

Note that Icon Buttons do **not** conform to the normal practice in GX where window items are sized in characters. In particular, it is **not** assumed that each button will occupy a single line as Icon Buttons may span several lines. This means care must be taken when positioning these buttons so as not to interfere with other items on screen, and if the buttons are placed at the bottom of the window some dummy labels must be included to increase the depth of the window. Once the buttons have been configured on the window the icons to be used are set up using the BTICN$ call.

Note that an Icon File (not a bitmap) **must** be specified for the buttons. Icon Files (32 * 32 pixels), unlike normal bitmap files, have a special invisible colour associated with them which allows the icon to blend into any background colour.

During the lifetime of a window that contains one, or more, Icon Buttons the BTICN$ routine is used in two phases. Before the window has been displayed, or entered, the BTICN$ routine must be used to establish the default icons. In the "before" phase, BTICN$ must be called with ICID set to the window-id of the target window; and ICACC set to SPACE. Note that ICACC is meaningless during the "before" phase because the Accelerator character is defined in the text of button label.

The length of the window-id can be either 2 characters (ICID) or 4 characters (ICID4C).

During the lifetime of a window an Icon Button can be dynamically changed by calling BTICN$ with ICID set to SPACES (i.e. to indicate the "current" window). In the "current window" phase ICACC must be set to the Accelerator letter of the new icon.

For each Icon Button, calling BTICN$ in the "before" is mandatory to establish the initial icon. However, calling BTICN$ in the "current window" phase, is optional and is only required if the logic of the window processing changes the meaning of a button(s).

The text of the Icon Button must be included in the Icon File itself. Thus, developers must be very careful to ensure that the Accelerator character defined in the Global program (either in the label text or in the ICACC field) agrees with the underlined character in the button text within the Icon File.

The *icon* string can be either an absolute pathname or a relative filename. If a relative filename is specified the Icon File is expected to be present in the Windows folder defined by the following setting in the GX.INI file:

```
[icons]
Folder=
```

The default file extension for Icon Files is .ICO.

For GSM SP-18, and lower, BTICN$ only operates in (slow) synchronous mode. In this mode, GX returns a result packet in response to the command packet sent by BTICN$. The results returned by GX allow BTICN$ to generate the appropriate STOP code or exception when GX detects an error condition.

For GSM SP-19, and higher, by default, BTICN$ operates in (fast) asynchronous mode. In this mode, GX does **not** return a result packet in response to the command packet sent by BTICN$. Thus, some error conditions, reflected in STOP codes and exceptions will not be detected (see the tables in section 2 for further details).

For GSM SP-19, and higher, the (slow) synchronous mode of BTICN$ that may return all possible STOP codes and exceptions must be enabled explicitly by setting the top bit (#80) of ICINX. It may be convenient to redefine ICINX as a PIC X field for this purpose. For example:

```
77    ICINXX REDEFINES ICINCX PIC X
...
MOVE #81 TO ICINXX                  * Set sync mode and index 1
```

In the (slow) synchronous mode of BTICN$ all the STOP codes and exceptions listed in the tables in section 2 may be generated.

For GSM SP-30, and later, it is possible to use BTICN$ to set the button item of an existing window, that is not the current window. This feature requires GX V4.5e, or later. To achieve this effect ICFLAG, ICVERS and ICMODE (all new for GSM SP-30) must all be set to 1.

For GSM SP-32, and later, it is possible to use BTICN$ to transform an “interrupt button” to an iconic button. To achieve this, set ICFLAG to #02 (if ICEXTN is not present in the control block) or #03 (if an ICEXTN extension is present in the IC block); and ICFTN to a value between 1 (for U-1) to 7 (for U-7).

4. Examples

[EXAMPLE REQUIRED]

5. Copy-Books

None.

6. See Also

None.