

## GXSBС\$ - Set Button Text on specific GX Window

The GXSBС\$ routine is used to set the text on a button within a specific window.

### 1. Invocation

To set the text on a button code:

```
CALL GXSBС$ USING wd bt text
```

where *wd* is a Window Definition control block; *bt* is a control block of the following format:

```

01      BT
02      BTVERS      PIC 9(4) COMP      * Version number of BT block
                                           * must be set to 1 or 2
02      BTID        PIC 9(4) COMP      * The function number of the button
02      BTINX       PIC 9(2) COMP      * Normally set to zero
                                           * but can be used to indicate
                                           * the button index if two, or more,
                                           * buttons have the same function number

```

and *text* is the zero-terminated text for the button. If the text string just contains a byte of binary-zero the button will be removed.

An alternative, and simpler, way to remove a button is to code:

```
CALL GXSBС$ USING wd bt
```

where *wd* is a Window Definition control block and *bt* is the control block described above.

### 2. STOP Codes and Exception Conditions

The following STOP codes may be generated by GXSBС\$:

STOP code	Description

15251	GXSBC\$ has been called by an application that is not running on GX.
15252	Unknown version of BT control block (i.e. BTVERS does not contain 1 or 2)
15253	Incompatible version of GX
15254	No buttons are displayed on the current window
15255	The specified button is not displayed on the current window
15256	The selected Accelerator Character is already in use
15257	Unspecified error from GX
15258	Button text-string is too long
15259	The target window is not displayed

The following exception conditions may be returned by GXSBC\$:

EXIT code	\$\$COND	Description
15206	6	Unable to allocate temporary work page

### 3. Programming Notes

GXSBC\$ is only available when running on GX. Any attempt to use GXSBC\$ on a non-GX terminal will result in a STOP code. The version of GX must be V3.5s, or later.

GXSBC\$ is only available with GSM SP-17, or later.

#### Version-1 Text Format

The following rules apply when BTVERS=1.

The Accelerator Character is defined as the **first** upper-case letter in the text-string. The text-string is always converted so the first letter in each word that appears on the button is converted to upper-case. For example:

<i>Text string</i>	<i>Button text</i>
example	Example
Example	<u>E</u> xample
eXample	<u>E</u> xample
example two	Example Two
example tWo	Example <u>T</u> wo
eXample tWo	<u>E</u> xample Two

The automatic upper-case conversion of the first letter of each word only occurs until the Accelerator Character is encountered in the string. No characters **after** the Accelerator Character are converted. Thus:

Desired button text	Actual text string
Abcd EfgH	abcd efgH
Abcd efgH	Not possible using the existing methodology
<u>A</u> bcd EfgH	Abcd EfgH
<u>A</u> bcd efgH	Abcd efgH
Ab <u>c</u> d EfgH	abCd EfgH
Ab <u>c</u> d efgH	abCd efgH
Abcd <u>E</u> fgH	abcd EfgH
Abcd <u>e</u> fgH	Not possible using the existing methodology
Abcd Efg <u>H</u>	abcd efGH
Abcd efgH	Not possible using the existing methodology

## Version-2 Text Format

The following rules apply when BTVERS=2. This option is only available with GSM SP-34, or later; and GX V4.7e, or later.

As described above, the version-1 algorithm does not allow some specific button-text formats to be specified (e.g. button-text strings that start with a lower-case letter). If BTVERS is set to 2 the text string is passed **unchanged** to the Windows button control. In this case, the accelerator (access) characters must be passed in the format expected by Windows as explained here:

[https://msdn.microsoft.com/en-us/library/aa983581\(v=vs.71\).aspx](https://msdn.microsoft.com/en-us/library/aa983581(v=vs.71).aspx)

In a nut-shell, prefix the accelerator letter by the ampersand (&) character. To include the ampersand character itself in the string, use a "&&" combination. The results will be

unpredictable if two, or more, non-consecutive "&" characters are included in the text-string; or if a "&" character appears before a non-alphabetic character.

For example:

<b>Text string</b>	<b>Button text</b>
&Print	<u>P</u> rint
E&xit	Ex <u>i</u> t
un&do	un <u>d</u> o
un&Do	un <u>D</u> o
&Jack&&Jill	<u>J</u> ack&Jill
Jack&&&Jill	Jack& <u>J</u> ill

## 4. Examples

[EXAMPLE REQUIRED]

## 5. Copy-Books

None.

## 6. See Also

GXSBT\$ Set Button Text on current window.