IMAGE$ - Display Image File

The IMAGE$ routine allows you to display an Image File.

1. Invocation

To display an Image File using IMAGE$ code:

```
CALL IMAGE$ USING st
```

where *st* is a control block defined as follows:

```
01 ST
   02    STLEN      PIC 9(4) COMP    * Length of path-name
   02    STPATH     PIC PTR          * Pointer to path-name
                                     * (e.g. d:\images\image1.bmp)
```

If STLEN is 0, IMAGE$ expected the path-name, in STPATH, to be a zero-terminated string.

2. STOP Codes and Exception Conditions

The following STOP codes may be generated by IMAGE$:

STOP code	Description
13603	IMAGE$ has been called by an application that is not running on GX.EXE or GSMWIN32.EXE.

The following EXIT codes may be returned by IMAGE$:

EXIT code	$$COND	Description
23907	7	Command failed ($$CRES contains the result of the command)

3. Programming Notes

IMAGE$ is only available when running on GX.EXE or GSMWIN32.EXE. Any attempt to use IMAGE$ on any other terminal will result in a STOP code.

IMAGE$ uses the SHCMD$ routine to invoke the Windows program, **gsmbmp32.exe**, to display the Image File. Note that gsmbmp32.exe was developed using Visual Basic-98 and the Windows Control it incorporates only supports the .BMP and "simple" .JPG file formats. It may fail with a "Run-time error '481' Invalid Picture" with complex .JPG files (e.g. if the .JPG file includes certain Exif information). The IMAG2$ routine, which uses SHCMD$ to invoke, **gsmjpg32.exe**, should be used for "modern" .JPG file formats.

4. Examples

None.

5.   Copy-Books
None.

6.   See Also
Imag2$.docx   Display Image File using gsmjpg32.exe
Shcmd$.docx   Execute Windows command