Identification

Symbolic reference to unavailable graphic character constants upper_case_char, punctuation_char.
Charles Garman

Purpose

This section describes two library data segments, upper_case_char and punctuation_char, for symbolic reference to graphic characters currently unavailable in character-string constants; it is to be withdrawn when all ASCII graphics are available in EPL, or when full PL/I arrives.

Background

Although the data character set for EPL is full 7-bit ASCII, at this writing the 26 upper-case alphabetic graphics and 12 of the punctuation-type graphics may not appear in EPL character-string literals. Specifically the upper-case alphabet maps into its lower-case counterpart, while the 12 non-language-character-set graphics map into the percent sign (%, octal 045). Until this situation is changed, either with later versions of EPL or the appearance of full PL/I, these graphics will be available symbolically in a similar manner to those in the data segment ctl_char (BY.8.01).

Usage

For each character which a program wishes to reference, the following declaration would appear

    dcl upper_case_char$lower_case_graphic char(1) ext;

or

    dcl punctuation_char$character_name char(1) ext;

For the upper-case characters, lower_case_graphic is simply the single lower-case counterpart of the desired character, e.g. "a". For punctuation characters, the graphics, names, and octal values are given below.
Graphic | Name                | Octal Value |
---------|---------------------|-------------
!        | exclamation_point   | 041         |
#        | number_sign         | 043         |
%        | percent             | 045         |
'        | acute_accent        | 047         |
@        | commercial_at       | 100         |
[        | open_bracket        | 133         |
\       | left_slant          | 134         |
]        | close_bracket       | 135         |
{        | grave_accent        | 140         |
}        | open_brace          | 173         |
}        | close_brace         | 175         |
~        | tilde               | 176         |

**Implementation**

These segments are created in essentially similar fashion as `ctl_char` (BY.8.01).