

Special Characters

Certain characters have a special meaning in Con-form. In some cases, you can assign the special meaning to a different character. The special characters and their meanings are described in this chapter.

The following topics are covered below:

- Ampersand (&)
 - Apostrophe (')
 - Circumflex (^)
 - Comma (,)
 - Commercial-at (@)
 - Dollar (\$)
 - Hash (#)
 - Period (.)
 - Semicolon (;)
 - Slash (/)
 - Space
-

Ampersand (&)

Initially, the ampersand is used as the variable character.

You can define a different character. For example, to define the paragraph sign as the variable character, you must specify:

```
.OP VSG=§
```

To cancel the special effect of the variable character, you must specify:

```
.SU OFF
```

Apostrophe (')

If a parameter contains spaces or commas, you must enclose it within apostrophes.

If a constant of a .IF or .WH instruction includes commas or blank spaces, you must enclose it within apostrophes.

Circumflex (^)

Initially, the circumflex is used as the shift character. It is only effective when lower-case is switched on (.LO ON).

You can also define a different character. For example, to define the asterisk as the new shift character, you must specify:

```
.OP SHI=*
```

Comma (,)

The comma is used to separate the parameters of an instruction.

If a macro has more than one parameter, you must insert a comma or space between two parameters.

If a constant of a .IF or .WH instruction includes commas or blank spaces, you must enclose it within apostrophes.

Commercial-at (@)

Initially, the commercial-at sign is used as the shiftlock character. It is only effective when lower-case is switched on (.LO ON).

You can also define another shiftlock character. For example, to define the hash as the new shiftlock character, you must specify:

```
.OP LOC=#
```

Dollar (\$)

Initially, the dollar sign is used as the end-of-line character.

The dollar sign is only interpreted as the end-of-line character when it is the last character in a source text line.

You can define a different character. For example, to define the percent sign as the end-of-line character, you can specify one of the following instructions:

```
.OP END=%
```

```
.EC %
```

Hash (#)

Initially, the hash is used as the page-number character. You can use it in top and bottom titles and with the `.SV` instruction.

The hash always represents the number of the current page. There is one exception: when you use the hash in the bottom title, it represents either the number of the current page (`.OP PAG=EQU`) or the number of the next page (`.OP PAG=DIF`).

You can define a different character. For example, to define the exclamation mark as the page-number character, you must specify:

```
.OP PNS=!
```

Period (.)

A period in the first column of a source text line is used to indicate a Con-form instruction.

To cancel the special effect of the period, so that a period in the first column is not interpreted as a Con-form instruction, you must specify:

```
.IC ON
```

You can also use a period to output another variable or text after the (first) variable without a blank in between (for example "`&var1.&var2`").

Semicolon (;)

Initially, the semicolon is used as the instruction separator character.

You can define a different character. For example, to define the colon as the instruction separator character, you must specify:

```
.OP CSE=:
```

Slash (/)

The slash is used as the separator character in top and bottom titles. It separates the three different areas for left-justified, centered and right-justified text. The text you define for each area must not include the slash.

Space

If you insert one or more blank spaces at the beginning of a line in your source text, you cause a break in filling. However, the formatted text also has the blank spaces in the beginning of the line.

If an instruction or macro has one or more parameters, you must insert a space between the instruction or macro and the first parameter.

If a macro has more than one parameter, you must separate the parameters by inserting a space or comma between them.

If a parameter includes spaces or commas, you must enclose it within apostrophes.

If you use the .IF or .WH instruction, you must include at least one blank space before and after the relational operator.

If a constant of a .IF or .WH instruction includes blank spaces or commas, you must enclose it within apostrophes.