Complex-function elements

The use of the qualifying symbols and dependency notation as described in A00269 through A00291, A00293, A00296, A00303, A00304, A00306, A00315, A00338, A00348 and relating graphical symbols may become impracticable for more complex circuit assemblies such as large-scale and very-large-scale integrated circuits. In these cases, the following techniques may be used.

1 General

All of the above-mentioned rules and concepts may be employed. However, dependency notation may be used only if no confusion with other labelling is likely.

2 Input and output designation

Inside the symbol outline, all inputs and outputs should be designated with the terminal names appearing on the selected data sheet or other documentation referenced in the description of symbol S01731 (12-54-01). This data sheet or other documentation should preferably be one that uses terminal names from a terminal-naming standard. Further abbreviation of these names should be considered only if these names are inconveniently long. For clarity, terminal names may be expanded or supplemented.

If labels defined in A00272 through A00291, A00293, A00296, A00303, A00304, A00306, A00315, A00338, A00348 and A00269xxx are used on the data sheet with a meaning other than that defined in those chapters, such labels shall be expanded to prevent confusion (for example DBUS instead of D).

In cases where the manufacturer's labelling prevents the use of the bit-grouping symbol for a clear representation of a bus, these labels may be modified provided correspondence with the data sheet is still possible.

3 Negated terminal names

Negated terminal names may be converted to the un-negated form inside the symbol by using the negation or polarity symbols, depending upon the convention in force. If an input or output serves two functions that are activated at opposite polarities, a branch on the connecting line may be used to permit two separate labels to be shown, thus avoiding a negation bar.

For an illustration, see A00317_Illustration_a_EN.pdf below.

4 In-line negation indication

If only an in-line notation can be used, the symbol ¬ (symbol 11-2.3 of ISO 31-11) shall be used instead of the negation bar defined in section 2 of A00276.

If this symbol is immediately followed by an identifying number in the sense of dependency notation, it applies to that identifying number only.

If this symbol is not immediately followed by an identifying number in the sense of dependency notation, the negation applies to the string to the right of the symbol up to the first of the following:

- an unmatched closing parenthesis, or

- a solidus that is itself not enclosed within a matching set of parentheses to the right of the symbol, or

- the end of the string.

For illustrations, see A00317_Illustration_b.pdf below.

5 Functional grouping

The connecting lines should be functionally grouped and, where appropriate, be partitioned into control and data lines. The control lines may appear on the "control block outline" for

which the common control block outline as described in A00270 is used.

6 Long character strings

Long character strings associated with input or output lines may be narrowed (at the expense of height) by inclusion in an open box as shown A00317_Illustration_c_EN.pdf below. The box shall open away from the input or output line. The broken character string shall be justified flush against the closed side, taking into account embedded spaces. To avoid ambiguity as to the continuity of a negation bar, multiple characters under a single bar shall not be split. Strings should not be broken in such a way that readability is impaired or an intended space is lost.

7 Consecutive labels and terminal designations

If both the internal labels and the (external) terminal designations are consecutive, then grouping of inputs [outputs] may be simplified by showing only the first and last connecting lines and their respective labels, the connecting lines being separated by dots or short strokes. Dots or short strokes may also be shown inside the symbol outline.

For an illustration, see A00317_Illustration_d.pdf below.

8 Function tables and truth tables

When tables are used to provide additional information about the behaviour of the circuit, the table entries should refer to logic levels or to external logic states.

If the table entries refer to external logic states on a theoretical logic diagram or when using a single logic convention, then, in the table, any label derived from one appearing inside the symbol at an input or output bearing a negation symbol must be modified by adding (or removing) a negation bar. All other labels should appear on the table without modification. For an example of use, see symbol S001735 (12-56-02).

9 Internal diagrams

To depict the functional behavior of a complex-function element, a diagram inside the outline of a symbol (referred to as an internal diagram) may be used. In this case, the following rules shall be observed:

- the negation or polarity indicator shall be shown at the symbol outline at those inputs [outputs] to which it applies to indicate the relationship between the internal logic state of the input [output] and its external logic state or logic level;

- by definition, logic States, and not logic levels, exist within the outline of a symbol. Therefore the symbol for logic polarity cannot be used on an internal diagram, and the symbol for logic negation shall be applied where appropriate;

- input and output labels shall be shown inside and adjacent to the symbol outline and/or inside and adjacent to the outlines of the symbols appearing on the internal diagram. Labels containing identifying numbers in the sense of dependency notation shall only be shown inside the outlines of the internal symbols to which they apply. When labels are repeated after the application of logic negation, they shall be modified by adding (or removing) a negation bar;

- connections solely between elements of the internal diagram need not be labelled.

For illustrations, see A00317_Illustration_e_EN.pdf below.