### (6) Relationship with Operational Block Skip

The slash "/" character used in the right-hand of an operational expression or in brackets is assumed to be the operator for quotient. It does not mean the optional skip.

### (7) Setting and Parameter of Program Number Classification

A. Disabling of Program Registration, Erase, And Edit

The following setting is permitted to protect the registered user macros and subprograms from inadvertent destruction:

Setting Number

#6004

D4 = 1 ··· The programs of program numbers #8000 through #8999 are disabled for registration, erase, and edit.

D4 = 0 ··· Registration, erase, and edit are enabled.

Parameter Number

#6021

D7 = 1 ··· The programs of program numbers #9000 through #9999 are disabled for registration, erase, and edit.

D7 = 0 ··· Registration, erase, and edit are enabled.

### (8) Effects of Reset Operation

A. A reset operation resets all local variables (#1 through #33) and part of common variables (#100 through #149) to "blank."

B. A reset operation resets the user-macro multiple call state and the multiple DO loop state, making the program pointer return to the program head.

### (9) Special Codes Usable in User Macro Body

A. The special codes listed below may be used in the user macro body:

| | Code | Use | EIA Code 8 | 7 | 6 | 5 | 4 | ∘ | 3 | 2 | 1 | ISO Code 8 | 7 | 6 | 5 | 4 | ∘ | 3 | 2 | 1 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | SP | For comment | | | | ○ | | ∘ | | | | ○ | | ○ | | | ∘ | | | |
| * | ( | For alarm message comment | | | | ○ | ○ | ∘ | | ○ | | | | ○ | | ○ | ∘ | | | |
| * | ) | | | ○ | | | ○ | ∘ | | ○ | | ○ | | ○ | | ○ | ∘ | | | ○ |
| | + | Add | | ○ | ○ | ○ | | ∘ | | | | | | ○ | | ○ | ∘ | | ○ | ○ |
| | - | Subtract | | ○ | | | | ∘ | | | | | | ○ | | ○ | ∘ | ○ | | ○ |
| | : | For comment | | ○ | | | | ∘ | ○ | ○ | | | | ○ | ○ | ○ | ∘ | | ○ | |
| | / | Divide | | | ○ | ○ | | ∘ | | | ○ | | | ○ | | ○ | • | ○ | ○ | ○ |
| | # | Variable | Parameter designation | | | | | | | | | | | ○ | | | ∘ | | ○ | ○ |
| * | * | Multiply | ○ | | | ○ | ○ | ∘ | | | | ○ | | ○ | | ○ | ∘ | | ○ | |
| * | = | Equal | ○ | | | | ○ | ∘ | ○ | | | ○ | | ○ | ○ | ○ | ∘ | ○ | | ○ |
| * | [ | Bracket(open) | ○ | | ○ | ○ | | ∘ | | | | ○ | ○ | | ○ | ○ | ∘ | | ○ | ○ |
| * | ] | Bracket (close) | ○ | | ○ | | | ∘ | | ○ | | ○ | ○ | | ○ | ○ | • | ○ | | ○ |
| | $ | For comment | ○ | | | ○ | | ∘ | ○ | | | | | ○ | | | • | ○ | | |
| | @ | | ○ | | | | ○ | ∘ | ○ | ○ | ○ | ○ | ○ | | | | ∘ | | | |
| | ? | | ○ | | | ○ | ○ | ∘ | ○ | ○ | | | | ○ | ○ | ○ | ∘ | ○ | ○ | ○ |
| | . | Decimal point | | ○ | ○ | | ○ | ∘ | | ○ | ○ | | | ○ | | ○ | • | ○ | ○ | |

Notes:

1. For the hole pattern of EIA code of the character attached with an asterisk, the pattern shown above is standard. However, other patterns may be specified by using the following parameters:

#6110 ··· [
#6111 ··· ]
#6112 ··· *
#6113 ··· =
#6114 ··· (
#6115 ··· )

Read the desired hole pattern in the binary value, convert it into the decimal equivalent, and set it to the parameter. For example, the hole pattern shown below is set as "152":

| 8 | 7 | 6 | 5 | 4 | 0 | 3 | 2 | 1 |
|---|---|---|---|---|---|---|---|---|
| ○ | | | ○ | ○ | ∘ | | | |

When the value of the parameter is "0," the hole pattern listed in the above table is provided.