

GETTEXT S21

DESCRIPTION

The GETTEXT function allows the user to program to access text messages and accumulate the strings in working data tables for display and printing, via S38, PRITEXT. The text strings can also be serially transmitted to other controllers.

FEATURES

- Text message number and length definable by user.

Table 1 Program rung data

Sym	Location	Use	Range
X	Upper rung input	Number of text message	0 to 32,767
Zm	Value location	Start of parameter list (2 locations)	Any write enabled table usable by special functions
Q	Rung output	Fault indication	0 for new errors

Table 2 Value data

Sym	Location	Use	Range
TS	Zm	Start of text string read area	Address of data table location obtainable by LOCATE
TL	Zm + 1	Length of text string to be sampled	1 to 255 characters. See Note 1

SPECIFICATION

Control language:-

```

      GETTEXT
X-----SPEC-----VALUE-----Q
      S21           Zm
  
```

The mnemonic GETTEXT is not automatically recognised by existing GEM80 programming units, but can be added as a user-defined mnemonic in the System Programmer.

Note: The user must ensure that sufficient data tables are allocated for reading in the message string, i.e. 2 characters per data table.

Table 2 Fault Conditions

Condition	Rung Output Q	
	Numeric Value	States ON
Successful execution	0	none
Message number accessing undefined null message	3	0 & 1
Message length greater than allocated table area	5	0 & 2
Programming error Invalid string read address	9	0 & 3

APPLICATION NOTES

If it is required to use S21 GETTEXT in conjunction with S38 PRITEXT to print out text from message editor, the following needs to be noted:-

S38 PRITEXT requires the length of the message in the low byte of Zm. This could be achieved by:-

- Supplying all messages of the same fixed length

OR

- Calculating each individual message length

1. The fixed length set would not have to be exceeded by any of the messages. Any messages which are shorter than this length would need to be filled up with either spaces or zeros.
2. Calculating each individual message length could be achieved by defining a block of data tables for the messages. Comparing this block of tables with zeros would determine where the message ended. Taking the result of when the comparison was true i.e. is equal to zero and therefore the end of the message, would be the length of the message. This value can then be placed in the first word as the message length for S38 PRITEXT.



Kidsgrove, Stoke-on-Trent, ST7 1TW, England
Tel: (0782) 783511 Fax: (0782) 776329

Note: The Company's policy is one of continuous development. Products supplied may differ from those described herein