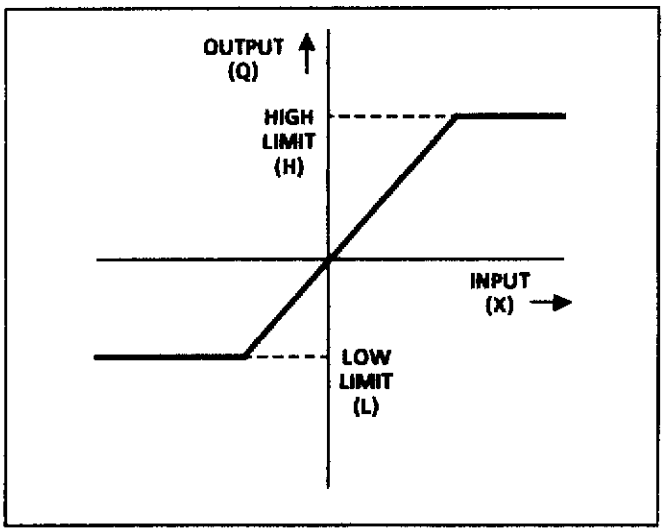
## DESCRIPTION

The LIMIT function provides a unity gain characteristic between input and output, but with the output restricted to a user defined range. Flags are provided to indicate the relationship of the input variable to the high and low limit points.




## FEATURES

- User programmable low and high limit points
- Comprehensive set of flags detecting above, equal or below limits

## SPECIFICATION

Control language:-

```
          LIMIT
X---------SPEC---------VALUE---------Q
          S32            Zm
```

Equation:

For $X \leq L$ (low limit), $Q = L$
$X \geq H$ (high limit), $Q = H$

$L \leq X \leq H$ $Q = X$

Table 1 Program rung data

| Sym | Location | Use | Range |
|---|---|---|---|
| X | Rung input | Input variable | ±32767 |
| Zm | Value location | Start of parameter list (3 locations) | Any write enabled table usable by special functions |
| Q | Rung output | Limited version of input variable | $L \leq Q \leq H$ |

Table 2 Value data

| Sym | Location | Use | Range |
|---|---|---|---|
| F | Zm | Fault code and flags | See tables 3, 4 and 5 |
| L * | Zm+1 | Low limit | Each ±32767 with L < H |
| H * | Zm+2 | High limit | |

* Data set up by user

Table 3 Fault conditions

| Condition | Result (Q) | Code in Zm Numeric Value | Code in Zm States ON |
|---|---|---|---|
| Programming error. No VALUE after SPEC. | 0 | - | - |
| Programming error. VALUE location invalid:- (a) Number instead of address (b) Zm in write protected memory (c) Zm in table not usable by special functions (d) Zm + 2 beyond end of data tables memory area | 0 | - | - |
| Invalid limit values L ≥ H | 0 | -12283 | 0,2,12 14,15 |

Table 4 Conditions indicated by Zm flags

| Condition | Code in Zm Numeric Value | Code in Zm States ON |
|---|---|---|
| Input above high limit X>H | -26112 | 9,11, 12,15 |
| Input equal to high limit X=H | -28672 | 12,15 |
| Input between limits L<X<H | 0 | None |
| Input equal to low limit X=L | 20480 | 12,14 |
| Input below low limit X<L | 22016 | 9,10, 12,14 |

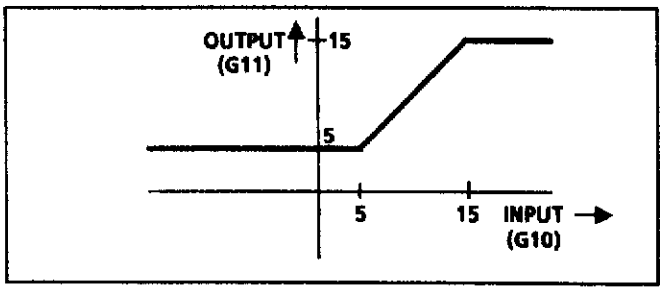
Table 5 Meaning of individual Zm flags

| State ON in Zm | Condition indicated |
|---|---|
| 0 | Fault condition |
| 2 | Low limit not below high limit L⩽H |
| 9 | Input outside range between limits X<L or X>H |
| 10 | Input below low limit X<L |
| 11 | Input above high limit X>H |
| 12 | Input equal or outside range between limits X≤L or X≥H |
| 14 | Input equal or below low limit X≤L |
| 15 | Input equal or above high limit X≥H |

## APPLICATION NOTES

1. Example

(a) Required characteristic




(b) Program rung:

```
|              LIMIT                          |
+---<AND>-----SPEC------VALUE-----<OUT>---+
|    G10       S32        W100        G11    |
```

(c) Value data

W100 flags (output by LIMIT function)
W101 5 (low limit)
W102 15 (high limit)

(d) Typical use of flags


This rung switches on output B0.0 when limits are operative, i.e. G10 < 5 or G10 > 15 and G10 ≠ G11.

2. Preset value data

It is usual for some of the "value" data to be calculated during the design stage of the particular application. Such data would be incorporated in the preset data table so that it can be stored and transported on cassette tape. For controllers with a write protected P table it is necessary for the user to arrange copying of the preset data into a working value table. The data sheet for MOVE function T20 shows a typical program employing LOCATE and MOVE functions for this purpose.


INDUSTRIAL CONTROLS

Kidsgrove, Stoke-on-Trent, ST7 1TW, England
Tel: (0782) 783511 Fax: (0782) 776329 Telex: 36293/4 CICKID G
Registered in England No. 2259095

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