CEGELEC
INDUSTRIAL CONTROLS

Multiplication, Division and Addition Calculations for Arrays of Data

CONARAY S10

DESCRIPTION

The CONARAY function applies the LINCON (S11) algorithm to each element of an input array, and places the results in an output array.

The CONARAY function treats a block of adjacent data table locations as a one dimensional array of numerical values. You can refer to any element within an array using an index that commences at 0 for the first word.

CONARAY allows you to either use:
- individual multiplier, divisor and offset values on each element in the array, or:
- the same multiplier, divisor and offset values on the whole array.

CONARAY produces results that are:
- rounded to the nearest integer:
- limited to be within the range ±32767 if they would otherwise overflow.

Where results are limited, CONARAY indicates this by setting the appropriate error code, and specifying the position of the first overflow on the rung output.

FEATURES

- User-specified input and output Array locations.
- User-programmable Array length.
- Automatic limiting at full scale output.
- Four quadrant operation.
- User-programmable slope and offset.
- Separate scaling values for each element of the array.
- Option of using the same values of multiplier, divisor and offset on the whole array.

SPECIFICATION

Control language:-

```
        CONARAY
X-------SPEC-----VALUE---------Q
          S10          Zm
```

Table 1 **Program rung data**

Sym	Location	Use	Range
X	Rung Input	Internal GEM80 address of input Array	Address of data table location obtained by using the LOCATE (S20) Special Function
Zm	Value location	Start of parameter list (Quantity of locations = 3 + 3GL)	Any write-enabled table usable by Special Functions
Q	Rung output	Fault indication	-1 for no errors

Table 2 **Value data**

Sym	Location	Use	Range
EC	Zm	Error code	See table 3
GL	Zm + 1	Array length. Negative length specifies that same multiplier, divisor and offset values are to be used on the whole array	1 to 4096* data table locations
SA	Zm + 2	Internal GEM80 address of output Array	Address of write-enabled data table location obtained by using the LOCATE (S20) Special Function
Mn	Zm + 3 + 3n	Multiplier for item n	-32767 to +32767
Dn	Zm + 4 + 3n	Divisor for item n	-32767 to +32767
Cn	Zm + 5 + 3n	Offset for item n	-32767 to +32767

* Note: Early implementations of this function were limited to 128 table locations.

Table 3 Fault conditions

Conditions	Result Q	Code in Zm	
		Numeric Value	States ON
PROGRAMMING ERRORS			
Array length invalid	0	3	0 & 1
Invalid array address: (a) Input Array not in table usable by Special Functions; (b) Input Array extends beyond end of data table area	0	5	0 & 2
Invalid array address: (a) Output array not inwrite-enabled table usable by Special Functions; (b) Output array extends beyond end of data table area	0	17	0 & 4
Divide by zero. Q = position of 1st error	word number	1025	0 & 10
STATUS			
LINCON operation causes positive overflow (limit to 32767). Q = position of 1st error	word number	257	0 & 8
LINCON operation causes negative overflow (limit to -32767). Q = position of 1st error	word number	513	0 & 9

PROGRAMMING RULES

1. A and B-tables

On current models of GEM80 Controllers that implement Array functions, you are not permitted to use A and B-table locations with the same number. e.g. you can't have both A7 and B7 in your program. When the Controller compiles your ladder diagram program, it checks it through to make sure that you do not have overlapping A-table and B-table references. If it finds any, it will produce and error message, and your program won't run.

However, the compiler can't do the same sort of checking where you use Group or Array Special Functions, as some of the table locations you are using will be implied, and will not appear in your ladder diagram explicitly.

If you want to use a block of A-table locations as an input or output Array, you must therefore be careful that the block of locations you specify does not overlap any A-table locations in use as the contents of these A-table locations could be affected.

2. Floating Point Tables

You cannot use ADDARAY, SUBARAY, MULARAY or DIVARAY with floating point table locations in Q-table. If you try to use them this way, you will get an error message, and your program won't compile.

APPLICATION NOTES

Program Example 1

You want to apply the CONARAY function (S10) to two adjacent data table locations, G10 and G11. The scaling and offset factors you want to use for G10 and G11 are different. You want to place the respective results into a second array, G200 and G201.

To do this, use the following rungs and associated table values:

```
|          LOCATE                                          |
+--<AND>---SPEC---VALUE-----------------<OUT>---+
|    0       S2      G200                  G2     |
```

```
|          LOCATE          CONARAY                         |
+--<AND>---SPEC---VALUE---SPEC---VALUE---<OUT>---+
|    0       S20     G10     S10     G0      F250    |
```

G0 = Error code
G1 = Array length = 2
G2 = Address of G200 obtained from first rung
G3 = Multiplier for G10 = 4
G4 = Divisor for G10 = 2
G5 = Offset for G10 = 6
G6 = Multiplier for G11 = 12
G7 = Divisor for G11 = 5
G8 = Offset for G11 = 0
F250 = Fault indication.

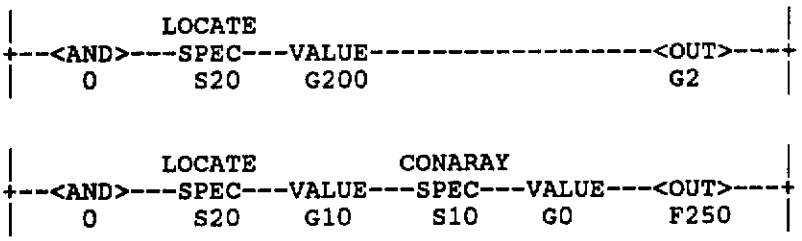
Typical Results:

Inputs		Outputs	
G10	5	G200	16
G11	3	G201	7

Program Example 2

You want to apply the CONARAY function (S10) to two adjacent data table locations, G10 and G11. This time, however, the scaling and offset factors you want to use are the same for both G10 and G11. You want to place the respective results into a second array, consisting of G200 and G201.

To do this, use the following rungs and associated table values:



G0 = Error code
G1 = Array length = -2
(the negative sign indicates common scaling factors for the whole of the array).
G2 = Address of G200 obtained from first rung
G3 = Multiplier for G10,G11 = 4
G4 = Divisor for G10,G11 = 2
G5 = Offset for G10,G11 = 6
F250 = Fault indication.

Typical Results:

Inputs		Outputs	
G10	5	G200	16
G11	3	G201	12

CEGELEC
INDUSTRIAL CONTROLS

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

Publication No. T333 Issue 4