

StructCalcOp

%StructCalcOp function

Function The function performs a given mathematical operation over a part of local structures. This function belongs to the control functions.

Declaration

```
%StructCalcOp(  
    in _recA[fromRowA]^fromColA, IN _recA[toRowA]^toColA,  
    in _recB[fromRowB]^fromColB,  
    in _recC[fromRowC]^fromColC,  
    INT in _operation [, IN BOOL _bInvalidIsZero = @TRUE],  
)
```

Parameters	_recA[fromRowA]^fromColA, _recA[toRowA]^toColA	Reference to the value of local structured variable field.
	_recB[fromRowB]^fromColB	Reference to the value of local structured variable field.
	_recC[fromRowC]^fromColC	Reference to the value of local structured variable field.
	INT IN _operation	Type of performed mathematical operation.
	IN BOOL _bInvalidIsZero	Optional parameter. Determines a behavior of mathematical operation in case that some of operands has an invalid value.

Description The value of parameter **_operation** determines the mathematical operation according to this table:

Value	Mathematical operation
0	+
1	-
2	*
3	/

The mathematical operation is performed as follows:

$$C = A + B$$

where:

Symbol A represents an area of local structured variable values which is given by its left upper corner (field **_recA[fromRowA]^fromColA**) and right lower corner (field **_recA[toRowA]^toColA**). The structured variable which is used in the first and second parameter must be the same. The dimensions (number of rows and columns) are also defined by this way. They are used in the definition of B and C areas that are defined only by their right upper corner in the parameters.

The mathematical operation (defined by a parameter) is performed among the individual appropriate fields of A and B areas. A result is written into the C area.

A set parameter **_bInvalidIsZero** (zeroing of invalid value) is interpreted as follows:

- using the operators **+** and **-**, the invalid values are, before a calculation, considered as zero values
- using the operators ***** and **/**, the result is 0

Example

In the following example there is shown an addition of two matrixes.

The first matrix is delimited by fields **_A[1]^firstColNr** (where **_firstColNr** has value =1) and **_A[2]^lastColNr** (where **_lastColNr** is set on consecutive number of the last column of relevant structure). The second matrix is defined by the field **_B[1]^firstColNr** and consists of two rows and of the same number of columns like matrix in variable **_A**.

The result is written to appropriate fields in variable **_C**.

```

BEGIN
  RECORD (SD.BIG) _A
  RECORD (SD.BIG) _B
  RECORD (SD.BIG) _C

  REDIM _A[20]
  REDIM _B[20]
  REDIM _C[20]

  INT _firstColNr, _lastColNr

  _firstColNr := 1
  _lastColNr := _A\CNR
  _A[1]^R1 := 1
  _A[2]^R1 := 2
  _B[1]^R1 := 1
  _B[2]^R1 := 2
  %StructCalcOp(_A[1]^_firstColNr,_A[2]^_lastColNr,_B[1]^_firstColNr,_C[1]
  ^_firstColNr,0)

  END

```



Related pages:

[ESL extended functions](#)
[Function Arguments - Types](#)