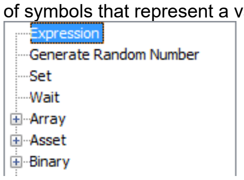
You are here: IoT Gateway User Guide > Projects and triggers > Trigger actions reference > Expression

Topic updated on September 21, 2023

Expression

The **Expression** action is a mathematical operation that takes a variable number of input arguments and returns a single output argument. An expression is any legal combination of symbols that represent a value. The system has rules for what symbols are legal and illegal.



Every expression consists of at least two operands with one or more operators. Operands are values, whereas operators are symbols that represent particular actions. For example, In the expression X + Y - 10. X, Y, and 10 are operands, and + and - are operators.

The following lists the valid operators.

Operator	Description
+	Addition
-	Subtraction
*	Multiplication
/	Division
^	Exponentiation (for example: 2^3 = 8)
%	Modulus (for example: 5%2 = 1)
( )	Force precedence of evaluation
>>	Bitwise shift right
<<	Bitwise shift left
&	Not implemented as an operator, see the and function below.
|	Not implemented as an operator, see the or function below.
^	Not implemented as an operator, see the xor function below.
~	Not implemented as an operator, see the not function below.

The following lists the valid logical operators for conditional operations:

Logical Operator	Description
==	Equal
!=	Not Equal
<	Less Than
<=	Less Than or Equal
>	Greater Than
>=	Greater Than or Equal
&&	And (for example: x==5&&y==10 returns true if x is 5 and y is 10)
||	Or (for example: x==5 || y==10 returns true if x is 5 or y is 10)
!	Not (for example: !(x==5) returns false if x is 5)

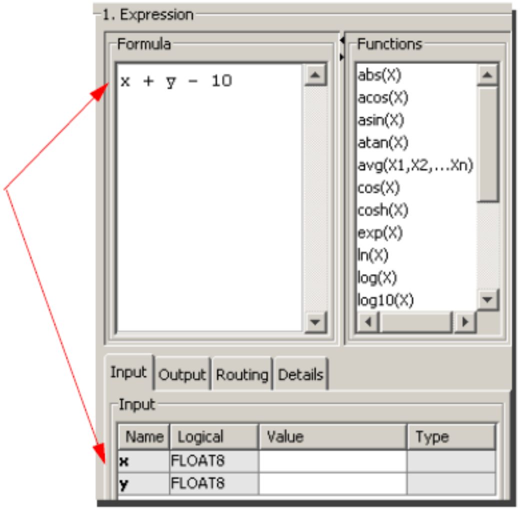
There are also a number of built-in functions used in an expression, such as sin(X) and log(X).

Parameter descriptions

The top of the Expression action provides the **Formula** and **Functions** parameters that accommodate creating an expression.

Formula

Use the **Formula** pane to type an expression.



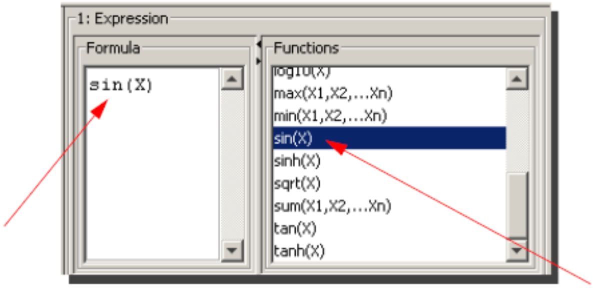
```
x + y - 10
```

Name	Logical	Value	Type
x	FLOAT8		
y	FLOAT8		

As you type in the **Formula** box, the logical variables used in the expression are automatically added to the **Input** tab.
In the example expression, x==y (x is equal to y), the logical variables **x** and **y** are added to the input tab.

Functions

You can use built-in functions within an expression. To add a built-in function, double-click the function in the **Functions** list.



```
sin(X)
```

The function is added to the **Formula** box. Note that the default input operand for the sin( ) function is **X**. You can change this to any value you want.
When using the trig functions, the angle is specified in radians. For example, sin(1.0) = 0.841471. The 1.0 is equivalent to 1 radian: where 2 pi radians equals 360 degrees

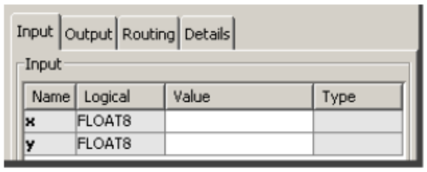
The following table lists the built-in functions that you can use within an Expression action.

Function	Description
abs(X)	Absolute value
acos(X)	Trigonometric arcosine
and(X,Y)	Bitwise AND "&"
asin(X)	Trigonometric arcsine
atan(X)	Trigonometric arctangent
avg(X1, X2...,Xn)	Average of a set of values
ceil(X)	Ceiling (round up)
cos(X)	Trigonometric cosine
cosh(X)	Hyperbolic cosine
exp(X)	e to the power X
floor(X)	Floor (round down)
ln(X)	Natural log (base e)
log(X)	Natural log (base e)
log10(X)	Log base 10
max(X1, X2...,Xn)	Maximum of a set of values
min(X1, X2...,Xn)	Minimum of a set of values
not(X)	Bitwise NOT "~"
or(X,Y)	Bitwise OR "|"
sin(X)	Trigonometric sine
sinh(X)	Hyperbolic sine
sqrt(X)	Square root
sum(X1, X2...,Xn)	Sum of a set of values

Function	Description
tan(X)	Trigonometric tangent
tanh(X)	Hyperbolic tangent
xor(X,Y)	Bitwise XOR "^"

Input tab

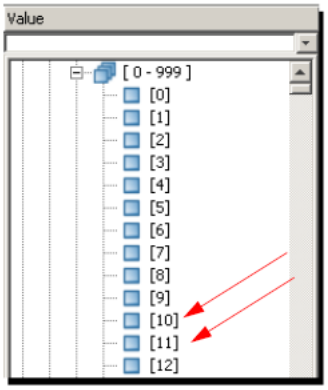
The following describes the **Input** tab available from the bottom of the **Expression** action pane. In the example expression, X + Y - 10 will be evaluated. However, before the trigger can evaluate this expression, values need to be assigned to both **X** and **Y**. The **Input** tab lets you assign values to the expression using trigger variables.



Input

Name	Logical	Value	Type
x	FLOAT8		
y	FLOAT8		

The following describes the columns on the **Input** tab.

Column	Description
Name	This column provides the expression variables. The expression is automatically added when you type the expression in the **Formula** box or double-click a built-in expression function from the **Functions** list. The variables used in the expression are just placeholders. They are not local variables, static variables, or device variables.
Logical	By default, all variables used in an expression have the data type **FLOAT8** which is an 8 byte floating point number. Depending on your system, you might see **LREAL** as the default data type. LREAL is also an 8 byte floating point number. When you assign the target value to hold the result of the expression, you can convert this FLOAT value to a different data type.
Value	The column lets you assign the trigger variable you want to use as input for the expression variable. Select the **Value** column to select from a list of device variables, or a user defined local or static variable. For this example, the actual PLC device variable Local CPU 1.D[10] is assigned to **X,** and Local CPU 1.D[11] is assigned to **Y**  Value [0 - 999] [0] [1] [2] [3] [4] [5] [6] [7] [8] [9] [10] [11] [12]
Type	When you specify **Value**, the default data type of the variable is automatically added to the **Type** column. You can override the default data type by selecting a different data type from the list.

When the trigger executes and evaluates this expression, the following actions will occur:

- Get the value of the device variable data type INT2 Local CPU 1.D[10].
- Get the value of the device variable data type INT2 Local CPU 1.D[11].
- Add these 2 values together.
- Subtract 10 from the value.

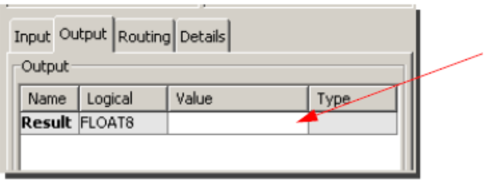
The last step is to specify the location to store the result of the expression.

Output tab

The **Output** tab for an **Expression** action lets you specify the location to store the result of the expression as follows:

Variables

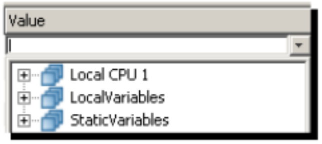
Use the **Variables** parameter to assign the variable that will store the result of the expression evaluation.



Output

Name	Logical	Value	Type
Result	FLOAT8		

Select the **Value** column and then use the down-arrow to display a list of trigger variables.

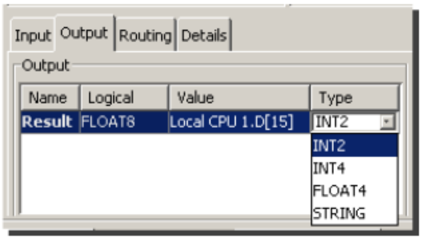


The list includes all started devices (for example Local CPU1), Local Variables, Static Variables, and Event Variables.

When you assign a device variable, the **Type** drop-down list becomes available with the associated data type of the variable.

Type

Use the **Type** drop-down list if the data type of the device variable needs to be changed.



Name	Logical	Value	Type
Result	FLOAT8	Local CPU 1.D[15]	INT2

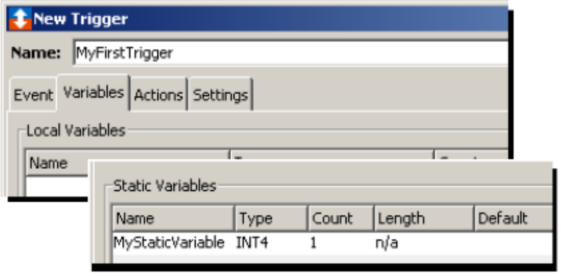
Only valid data types are listed. For this example, PLC device variable Local CPU1.D[15] is assigned to hold the result of the expression.

To review. When the trigger executes, the expression is evaluated and the following will occur:

- Add Local CPU1.D[10] to Local CPU1.D[11]
- Subtract 10 from the value
- Store the result in Local CPU1.D[15]

Specifying a static variable to store the result of the expression

You can also specify a static variable as the recipient of an expression evaluation. A static variable retains its value while the trigger is in a started state. It is assumed that you have defined the static variable using the **Static Variables** tab on the New Trigger window.

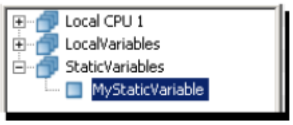


Name: MyFirstTrigger

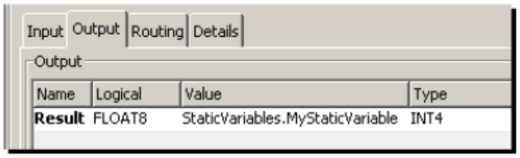
Name	Type	Count	Length	Default
MyStaticVariable	INT4	1	n/a	

When the static variable is defined, its initial value is created when the trigger is started. The value (and possibly modified value) continues to exist while the trigger is in a started state and is lost when the trigger is stopped.

The steps to add a static variable to the **Output** tab for the expression are the same as a device variable, except you select a static variable from the list.



The **Value** column will contain the static variable.



Name	Logical	Value	Type
Result	FLOAT8	StaticVariables.MyStaticVariable	INT4

Notice the **Type** drop-down list is unavailable. You cannot change the data type of a static variables from this list.

Related topics

If

About Telit | Contact Us | Legal Notices | Terms of Service | Privacy Policy

Copyright © 2025, Telit IoT Solutions Holding Ltd.. All rights reserved.