Introduction

Some Iddero products support **complex logic and arithmetic expressions**. This includes the Iddero HC3-KNX and HC3L-KNX touch panels, and Iddero Home Server 3.

This functionality requires firmware version **4.7** or later.

Functional description

Overview

Logic and arithmetic expressions are defined as a text string (single line). They can use input values from up to four communication objects (A, B, C, D). The expression is reevaluated each time one of the input objects is updated. The value resulting from the evaluation of the expression is then written to the output communication object.

Each input object and the output object can be configured as DPT 1 or as any numeric DPT (either integer or floating point).

Syntax

Expressions support both logic and arithmetic operations. The syntax is **case sensitive**.

Values:

- Internally, all numeric values are floating point values (IEEE 754 double-precision)
- Boolean values are either `true` (also, `on` / `ON`) or `false` (also: `off` / `OFF`)
- The special value `nil` is used to represent “no value”
- Values from the input objects are represented as `A`, `B`, `C`, `D`

Operators:

- Arithmetic: `+` (addition), `-` (subtraction), `*` (multiplication), `/` (division)
- Relational: `==` (equal), `~=` or `!=` (not equal), `<`, `<=`, `>`, `>=`
- Logical: `and` (also: `&&`), `or` (also: `||`), `not` (also: `!`)

Functions:

- The special function `IF(cond, a, b)` can be used to select a value (either “a” or “b”) depending on a logic condition (“cond”).

Examples

Simple temperature comparator with boolean output:

> Inputs: A, B as DPT 9
> Output as DPT 1
> Expression: `A > B`
>
> Description: if A > B, “1” is written to the output object, otherwise, “0” is written

Select a numeric value (for example, light intensity), depending on a logic condition (for example, presence detection coming from two different sensors):

> Inputs: A, B as DPT 1
> Output as DPT 5.001 (percentage)
> Expression: `IF(A || B, 100, 10)`
>
> Description: if presence is detected by any of the two sensors (A and/or B are “true”), then write 100% to the output, otherwise write 10% (note that the % symbol should NOT be included in the expression)

Conditionally trigger a scene:

> Inputs: A, B, as DPT 1
> Output as DPT 5
> Expression: `IF(A || B, 0, nil)`
>
> Description: if either A or B is “true”, then trigger scene 1 (write value “0” to output object). Otherwise, do nothing. Note how `nil` is used so that the output is not updated in this case.

Simple fancoil control:

> Input: A as DPT 5.001 (percentage)
> Output as DPT 5
> Expression: `IF(A > 66, 2, IF(A > 33, 1, 0))`
>
> Description: writes 0-1-2 to the output object, depending on the input value:

- If input value > 66%, write “2” to the output
- Otherwise: if input value > 33%, write “1”
- Otherwise: write “0”

Note the use of nested IF function calls. Also, note that for percentage values, the % symbol should NOT be included in the expression.

Convert temperature from ºC to ºF for visualisation:

Input: A as DPT 9
Output as DPT 14 (for better precision of the displayed value)
Expression: `(A * 9 / 5) + 32`

Description: converts a temperature value received from the bus (DPT 9) to Fahrenheit. Using DPT 14 for the output object allows to display the converted value without loss of precision.

Simple logic expression involving multiple inputs and multiple operators:

All inputs and outputs configured as DPT 1
Some expression examples:

```
A && B
A && (B || C || D)
(A || B) && C && !D
```

Detailed evaluation process

Each time the expression is triggered, it is evaluated as follows:

1. Input values are processed:
   - For objects configured as DPT 1, the current object value is converted to an internal boolean value (1 → true, 0 → false)
   - For objects configured with any other DPT, the current object value is converted to a floating point value (IEEE 754 double-precision)
2. The expression is evaluated
3. The result of the expression is processed:
   - If the result is “nil”, then the output is not updated
   - If the result is a numeric value:
     - If the output object is configured as DPT 1, the object is updated as follows: 0 is written to the output unchanged; any other value is written as 1
     - If the output object is configured with any other DPT, the result is converted to the actual output DPT:
       - For integer DPTs, the value is rounded to the nearest integer
       - Out of range values are limited to the actual range of the output DPT

        - Special values (NaN, positive/negative infinity) are only written to the output object for DPT 14. For other DPTs, the output is not updated.
    - If the result is a boolean value:
        - If the output object is configured as DPT 1, the result is converted to 1/0 (true → 1, false → 0) and written to the object
        - If the output object is configured with any other DPT, this is considered an error and the object is not updated