

_Bit32

Um im LSE Bitvariablen für Checkboxes verwenden zu können, muss ein Objekt dieser Klasse instanziiert werden. Diese Variablen können dann im LSE als Checkbox platziert werden.

Um ein einzelnes Bit in einem 4-Byte Datenblock zu setzen oder zu löschen, muss folgendes Format beim Schreiben auf den Server **Data** eingehalten werden:

Auf dem höchstwertigen Bit (Bit 31) steht die Information, ob das gewünschte Bit gesetzt (1) oder gelöscht (0) wird. Die Bits 0 bis 30 definieren das gewünschte Bit. Dieses wird mit einer 1 gekennzeichnet. Alle anderen Bits bleiben auf 0.

Beispiel: Wird der Wert 16#80000001 auf diesen Server geschrieben, so wird lediglich das Bit 0 auf 1 gesetzt, alle anderen Bits bleiben unverändert.

Damit alle 32 Bits verwendet werden können, gibt es 2 Sonderfälle. Wird dem Server der Wert 16#80000000 übergeben, so kann dies nur bedeuten, dass das Bit 31 (msb) auf 1 gesetzt werden soll. Wird 16#00000000 übergeben, so wird nur Bit 31 (msb) gelöscht.

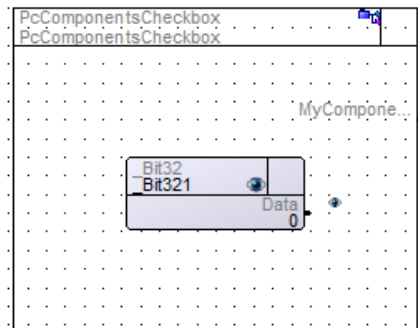
Schnittstellen

Server

Data	Um die Eingaben aus der Visualisierung auf einzelne Bits zu „übersetzen“, muss in der Writemethode des platzierten Servers die Writemethode des Servers Data aufgerufen werden. (siehe Codebeispiel, die Klasse _Bit32 wird dazu in die Klasse, deren Server zu platzieren ist, eingebettet).
-------------	---

Beispiel

Objektnetzwerk der komplexen Klasse PcComponentsCheckbox:



Writemethode des Servers MyComponents:

```

FUNCTION VIRTUAL GLOBAL PcComponentsCheckbox::MyComponents::Write
VAR_INPUT
    input (EAX)    : PcComponentsRadio::_PC_COMPONENTS;
END_VAR
VAR_OUTPUT
    result (EAX)   : PcComponentsRadio::_PC_COMPONENTS;
END_VAR

MyComponents := _Bit321.Data.Write(input:=input);
result := MyComponents;
END_FUNCTION

```

Die Write-Methode wird also mit einem Wert, wie weiter oben erklärt, von der Visualisierung aufgerufen. Die Umsetzung auf das einzelne Bit wird dann von der Methode `_Bit32::Data::Write` übernommen.

Voraussetzung dafür, dass der Wert wie erwartet von der Visualisierung übergeben wird, ist, dass ein BDINT-Typ dem Server zugewiesen ist. Im LASAL SCREEN Editor können dann einzelne Bits des Servers als Checkbox platziert werden.