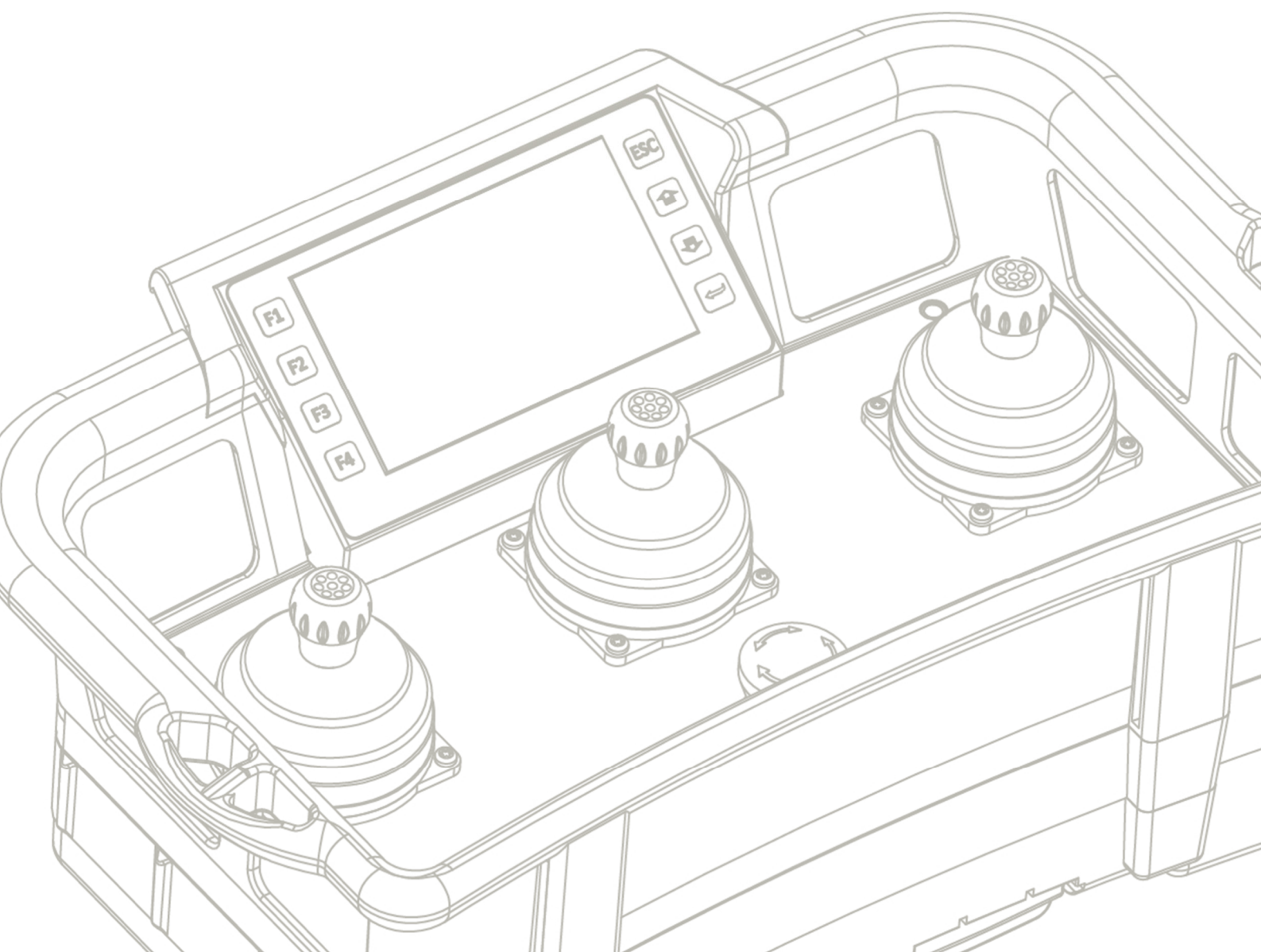


# Programmable Display - User manual

---

Version 0.2 / 17.01.2023



## Disclaimer

The content of this manual was checked for errors. Nevertheless, deviations can't be ruled out, so that we assume no liability for full correctness.

## Trademarks

Registered names, trademarks, etc. used or mentioned in this manual may be trademarks of their respective companies.

## Contact address

NBB Controls + Components GmbH  
Otto-Hahn-Straße 3 – 5  
75248 Ölbronn-Dürrn  
Germany

E-mail: [info@nbb.de](mailto:info@nbb.de)  
Phone: +49-7237 999-0  
Fax: +49-7237 999-199

## Manual versions

Version	Revision	Date	By	Description
0.2	0	17.01.2023	LG	Update of the chapter 1.2
0.1	3	06.09.2022	LG	Added the sales contact
0.1	2	26.07.2022	LG	Added chapter "update mode"
0.1	1	13.05.2022	LG	Pre-release
0.1	0	03.05.2022	LG	Document created

## About this document

### Assumptions

This document assumes that you have a solid knowledge of the following:

- The data which are sent by your machine to the NBB receiver.
- The programming language C.

### Requirements

- Installed and up-to-date version of the GSe-VISU software including the NBB extension.
- Project template from NBB.
- A GSe-VISU licence.
- A computer with a WLAN interface.
- A transmitter from NBB with the programmable display.

## Table of contents

<b>1. Overview .....</b>	<b>6</b>
1.1. What is the "programmable Display"? .....	6
1.1.1. How does it work?.....	6
1.2. The programmable display package.....	7
1.2.1. Download .....	7
1.2.2. Content.....	7
1.2.3. Installation.....	7
1.3. Licencing .....	8
1.3.1. Demo licence.....	8
<b>2. The GSe-VISU Application .....</b>	<b>9</b>
2.1. Getting started.....	9
2.1.1. Open a compressed project .....	9
2.1.2. Open a project.....	10
2.1.3. Building a project .....	11
2.1.4. Project simulation.....	12
2.2. GSe-VISU user manual.....	13
2.3. NBB related parameter.....	13
2.3.1. WLAN settings .....	13
2.3.2. GSe-VISU user manual.....	14
2.3.3. Operating system .....	15
<b>3. NBB template.....</b>	<b>16</b>
3.1. Create a new project from the template .....	16
<b>4. Project export .....</b>	<b>17</b>
4.1. Enabling the update mode .....	17
4.2. Connecting to the display.....	18
4.3. Project upload.....	18
<b>5. Basic examples.....</b>	<b>20</b>
5.1. Keys.....	20
5.2. System language .....	22
5.3. Switching masks.....	23
5.4. Backlight.....	23
5.5. Feedback.....	24

## 1. Overview

### 1.1. What is the “programmable Display”?

With the programmable display, NBB offers a new generation of HMI for the transmitters Nano-media and Nano-magna. In addition to the possibility of freely designing the display, the associated development studio offers a graphic editor that is easy and intuitive to use.

The integrated WLAN module allows a project update directly from the PC via WLAN. In addition, the WLAN interface offers the connection of WLAN cameras, whose video stream can be shown on the display. In this way, blind angles can be easily monitored.

#### 1.1.1. How does it work?

Part of the programmable display is an integrated development environment (GSe-VISU) for creating visualizations and user programs in C/C++.

#### Features of GSe-VISU IDE:

- What you see is what you get (WYSIWYG) editor.
- Simple and intuitive graphic objects for displaying process parameters.
- Resources for fonts, Unicode-capable text objects, colours, bitmaps, touch buttons and process variables
- Integrated scale generator.
- Project uploads over WLAN.

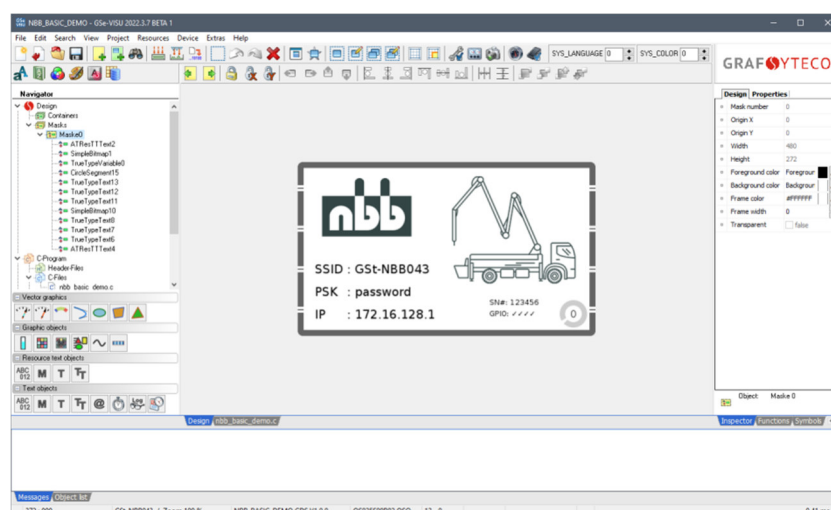


Figure 1 - GSe-VISU designer view

## 1.2. The programmable display package

### 1.2.1. Download

A download link is only available per request. The package is delivered as \*.zip container. Please contact our sales team for more information.

### 1.2.2. Content

Included in the package are the GSe-VISU setup execution file, a project template, the latest OS image (if the GSe-VISU does not contain it) and the user manual.

### 1.2.3. Installation

Run the execution file (\*.exe) to start the installation wizard. Follow the instruction in the wizard to complete the installation.

Important:

In the User Information screen, the serial number *D-NBB0* has to be entered!

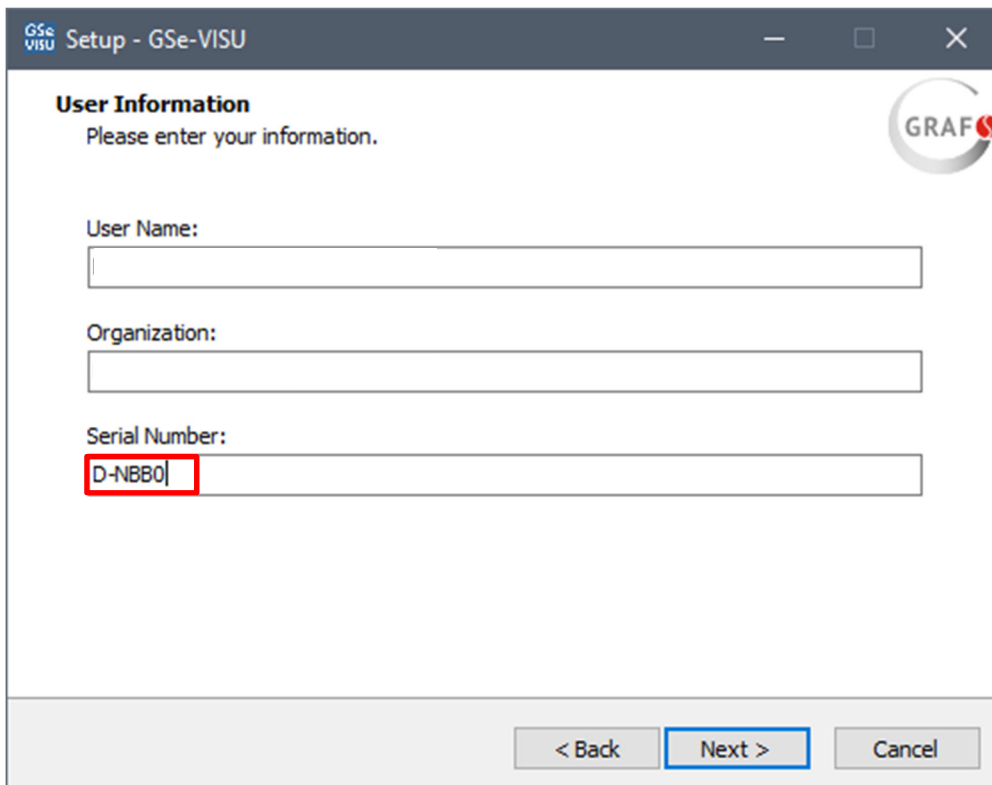


Figure 2 - Serial number

## 1.3. Licencing

After installing the software for the first time on a system, projects cannot be exported because a valid licence file must be loaded. These are available from NBB.

After starting GSe-VISO, the licence can be loaded via the main menu *Help->License->Load*. If the licence is valid, the full functionality of the GSe-VISU software package is available to you from this point.

### Sales contact

A sales representative for your country can be found on our website:

<https://nbbcontrols.com/en/the-company/sales/>

### 1.3.1. Demo licence

Without a licence, only building and simulation of the project is possible. The C/C++ and visualization editor is not affected and works without restrictions.



## 2. The GSe-VISU Application

### 2.1. Getting started

The software package comes with some example projects. These projects give a good overview over the functionality and capability of the display and IDE.

#### 2.1.1. Open a compressed project

The example projects are delivered as a compressed \*.gdsz file. To open and extract a compressed project, navigate to the main menu bar and select:

*File -> Open Project from GDSZ...*

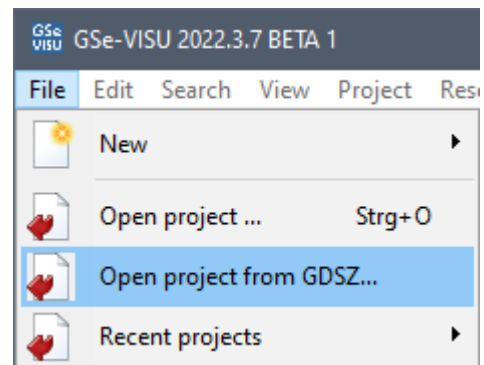


Figure 3 - File context menu

After selecting a project, a new window will pop up. The project will be extracted to the standard location. If you want to choose a location, select the desired directory via the browse button<sup>1</sup> and continue with OK.

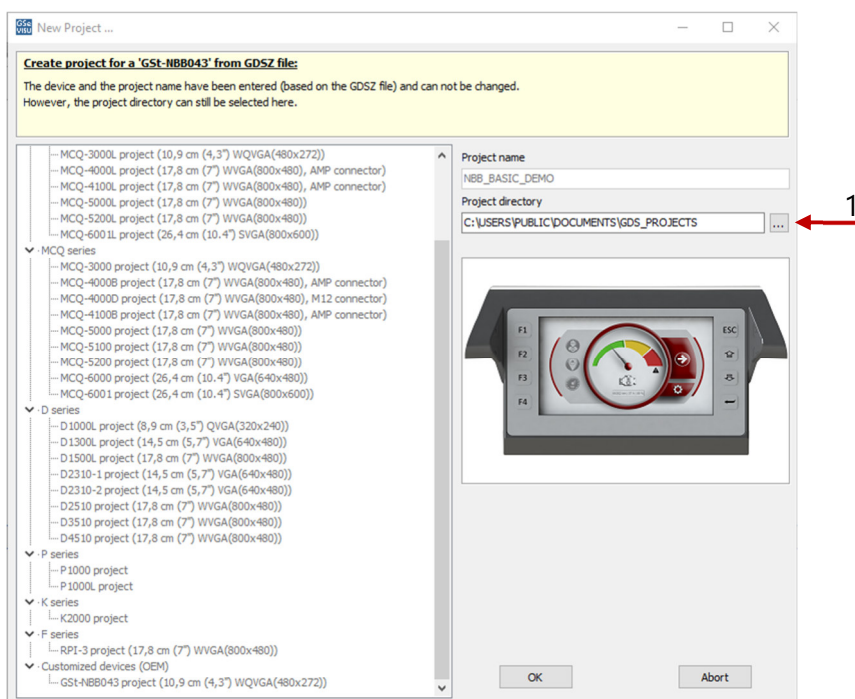


Figure 4 - New project window

### 2.1.2. Open a project

There are several options to open a project. Once the GSe-VISU application is started, you can see the Startup window. Here you can select your recent projects or open a new project over the Open project dialog. Alternatively, the options can also be accessed via the File menu item.

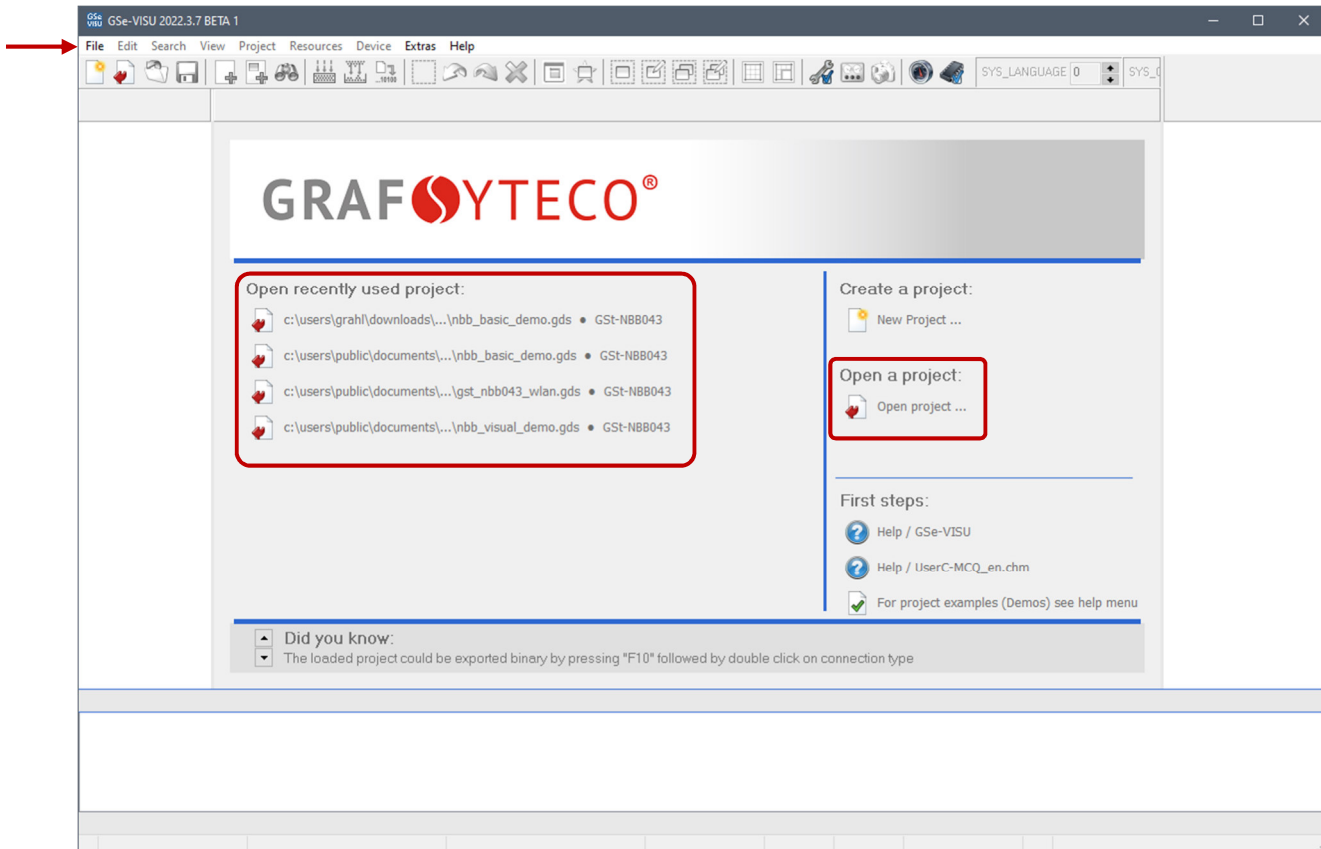


Figure 5 - GSe-VISU start page

### 2.1.3. Building a project

To check for errors the project can be compiled. The compilation can be initiated through the main menu *Project -> Make* or *Project -> Make rebuild*.

If one or more errors occur during the compilation, it will be displayed in the Message box at the bottom.

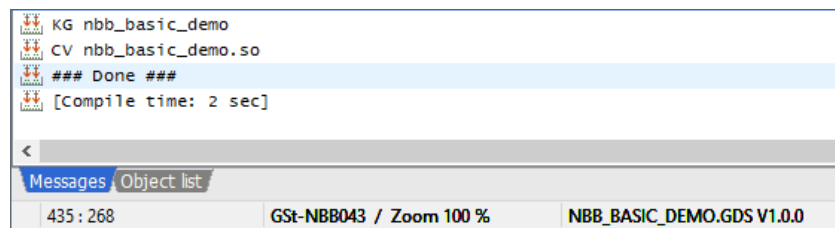


Figure 6 - Message box

### 2.1.4. Project simulation

After a compilation with no errors, the project can be uploaded to the display or the application can be simulated on the computer.

For an upload, a valid licence is required. More about the licence can be found in chapter 1.3. The simulation can be started with the Export dialog: *Project -> Export binary*

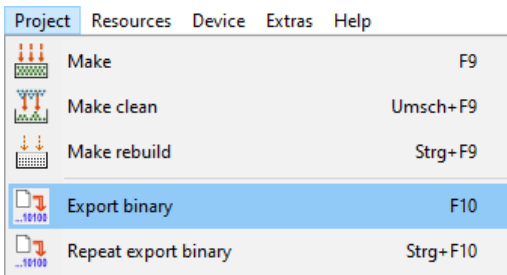


Figure 7 - Project context menu

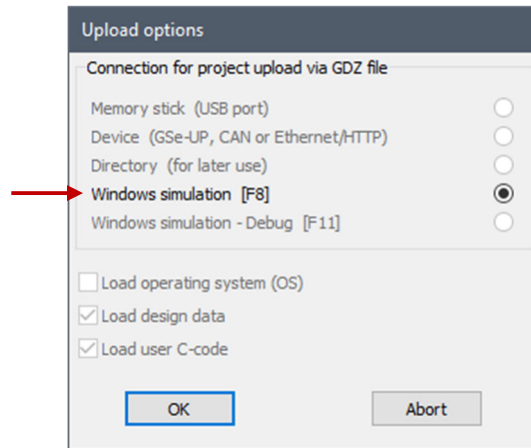


Figure 8 – Project export dialog

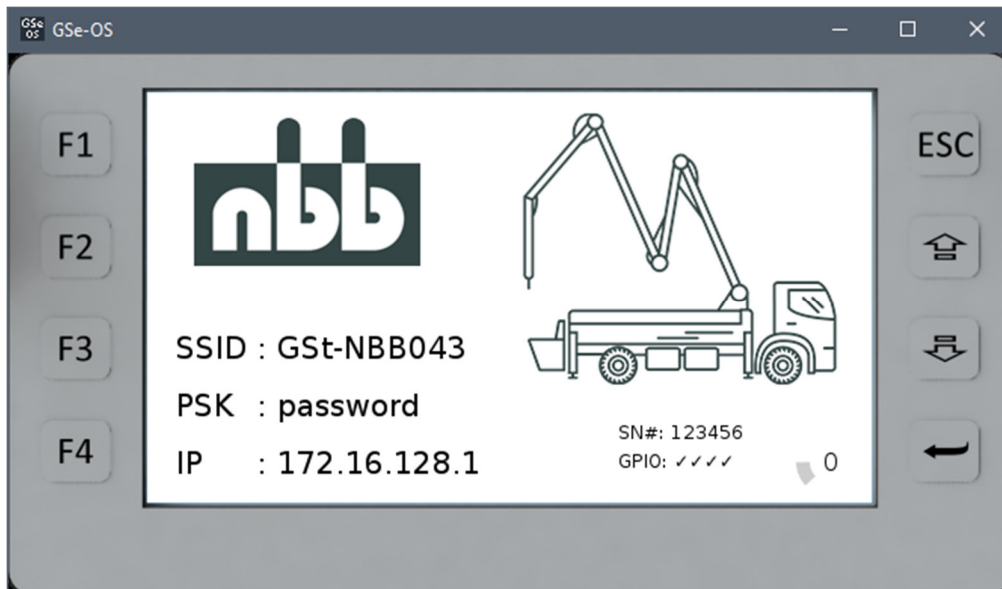


Figure 9 - Windows simulator

## 2.2. GSe-VISU user manual

The GSe-VISU IDE includes its own documentation. These documents can be accessed through the main menu *Help -> Help*.

## 2.3. NBB related parameter

### 2.3.1. WLAN settings

The display has an integrated WIFI module which acts as an access point. Connected clients can then upload new projects.

The settings can be changed at: *Device -> Device parameters -> Interfaces -> WLAN*

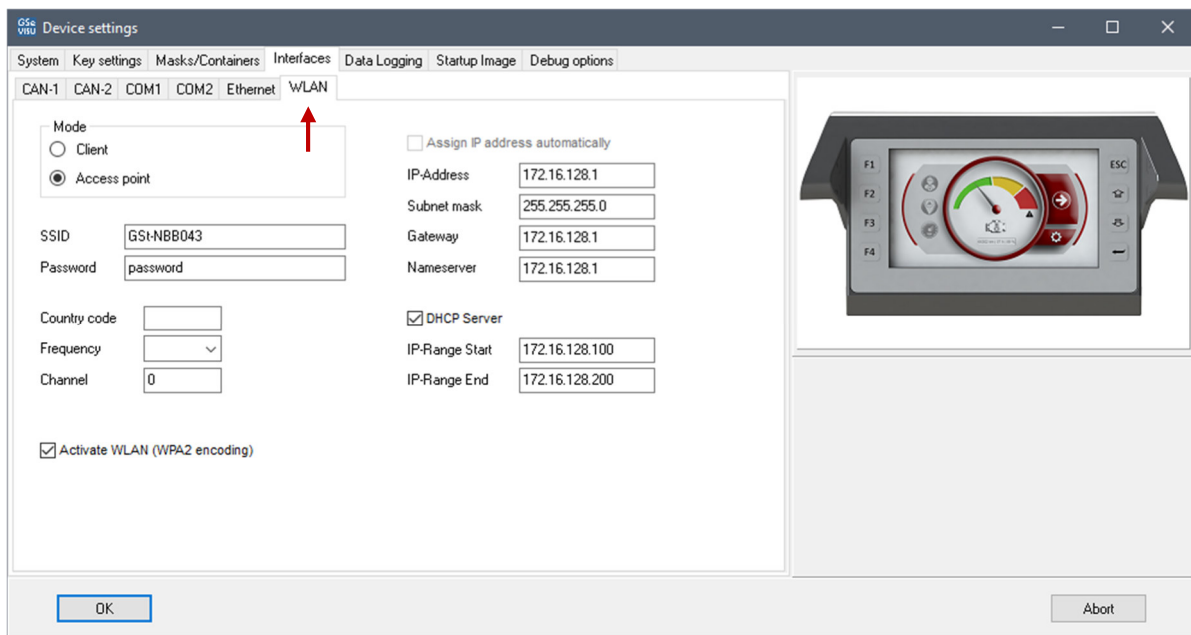


Figure 10 - WLAN settings

**Note:**

The factory settings of the WLAN network, unless specifically specified by the customer, are:

- SSID = GSt-NBB043
- Password = password
- IP-Address = 172.16.128.1

### 2.3.2. GSe-VISU user manual

An image is displayed while the display boots up.

This image can be exchanged. For example, your own company logo can be displayed.

To select your own image, open the device parameters under the menu item Device.

Navigate to the tab Startup Image and choose the image you like.

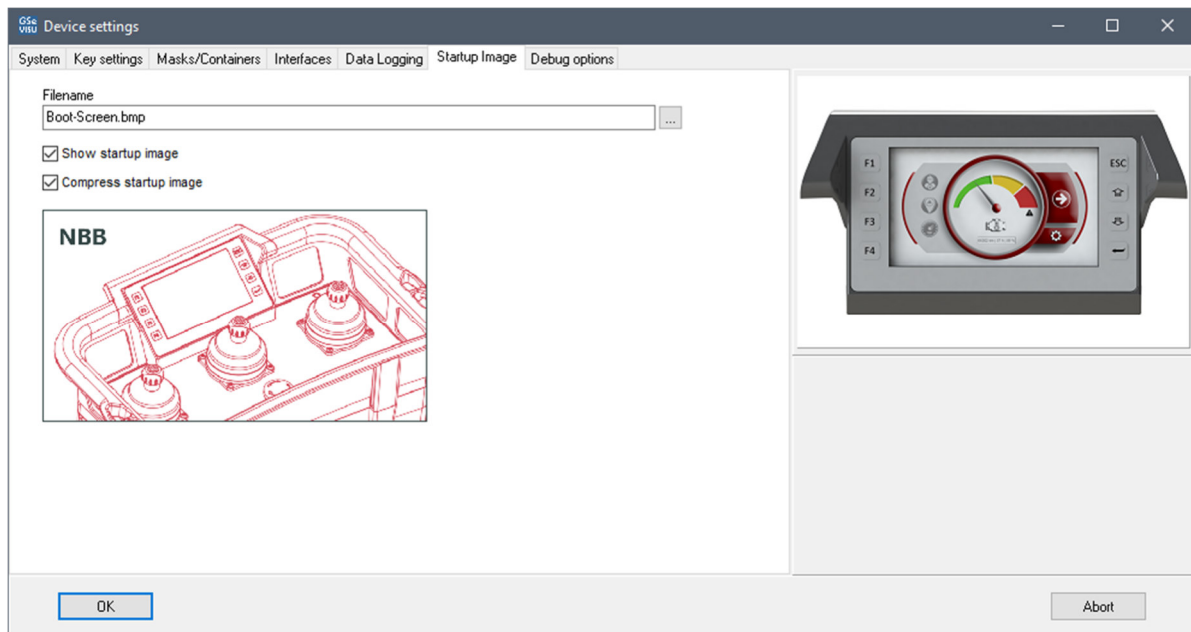


Figure 11 - Startup image selection

**Note:**

The image must be a BMP file. An image size of 480x272 is recommended.

### 2.3.3. Operating system

On the display runs an operating system which is responsible for the hardware and basic functionality of the display.

Select only the latest OS version for the project. The OS selection can be found in the device settings under *System* -> *System settings*.

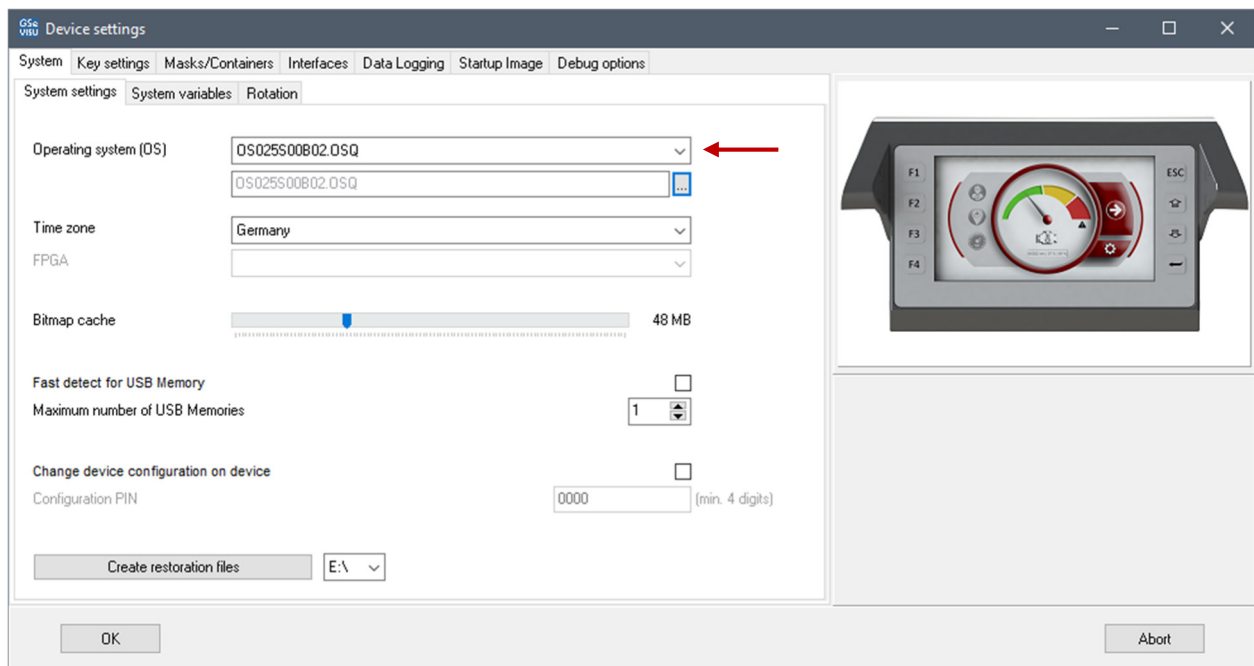


Figure 12 - OS selection

To select a newly released OS from the drop-down menu, it has to be placed in the folder:  
 C:\Program Files (x86)\GRAF-SYTECO\GSe-VISU\OS\

**Note:**

The path may vary depending on where GSe-VISU was installed.

### 3. NBB template

Included in the download is a project template from NBB. This template is preconfigured for the display. Therefore, it is highly recommended to create new projects from this template.

#### 3.1. Create a new project from the template

1. Open the template \*.gds file.
2. Save the template as a new project with the menu item *Save project as ...* .

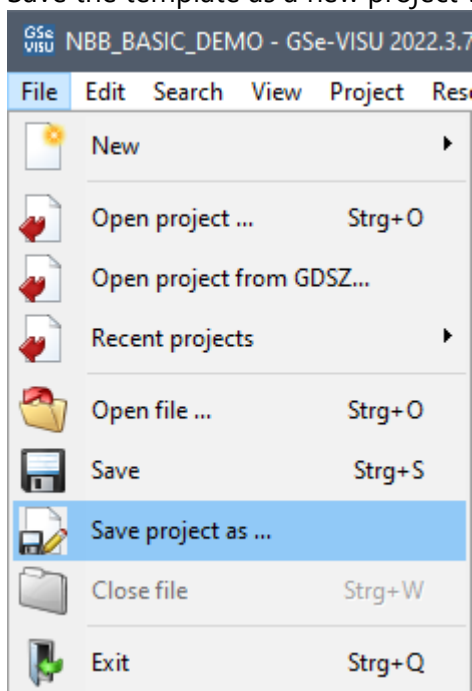


Figure 13 - File context menu

3. Select the new location and assign a name to the project.  
All necessary files and folders will be created.
4. After the new project is created, it will be loaded automatically.



## 4. Project export

### 4.1. Enabling the update mode

In normal operating mode, the WLAN is deactivated. To switch to update mode, the *ESC* and *Enter* keys must be pressed simultaneously at startup.

When the update mode is successfully activated, a pop-up is displayed and after 5 seconds the display restarts with active WLAN.

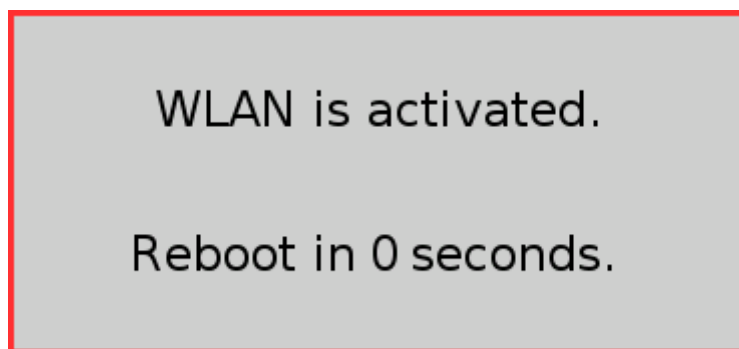


Figure 14 - Info pop-up

**Note:**

The update mode is automatically deactivated when shutting down or restarting the display. After transferring a new project, however, the WLAN remains active.

Loss of power does not count as a shutdown.

## 4.2. Connecting to the display

After the display has booted into update mode, it provides a WLAN access point. A connection to the WLAN runs as usual in the Windows network settings. The WLAN SSID and the password can be changed as shown in Chapter 2.3.1.

## 4.3. Project upload

If there is a connection between computer and display, the upload can be initialized via GSe-VISU. To do this, start the project export via the menu. (*Project -> Export binary*)

Select the Device option and continue with OK.

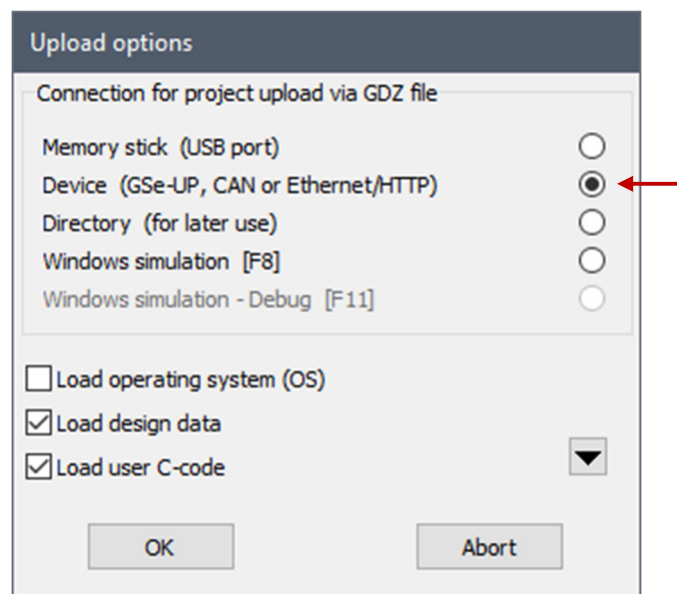


Figure 15 - Project export dialog

The project is now ready for upload. If no error occurs a new window appears. Here the HTTP tab is selected and the IP of the device is entered.

The project is transferred to the display with the Upload button.

If the connection is successful, the update status appears on the display and a restart is carried out automatically.

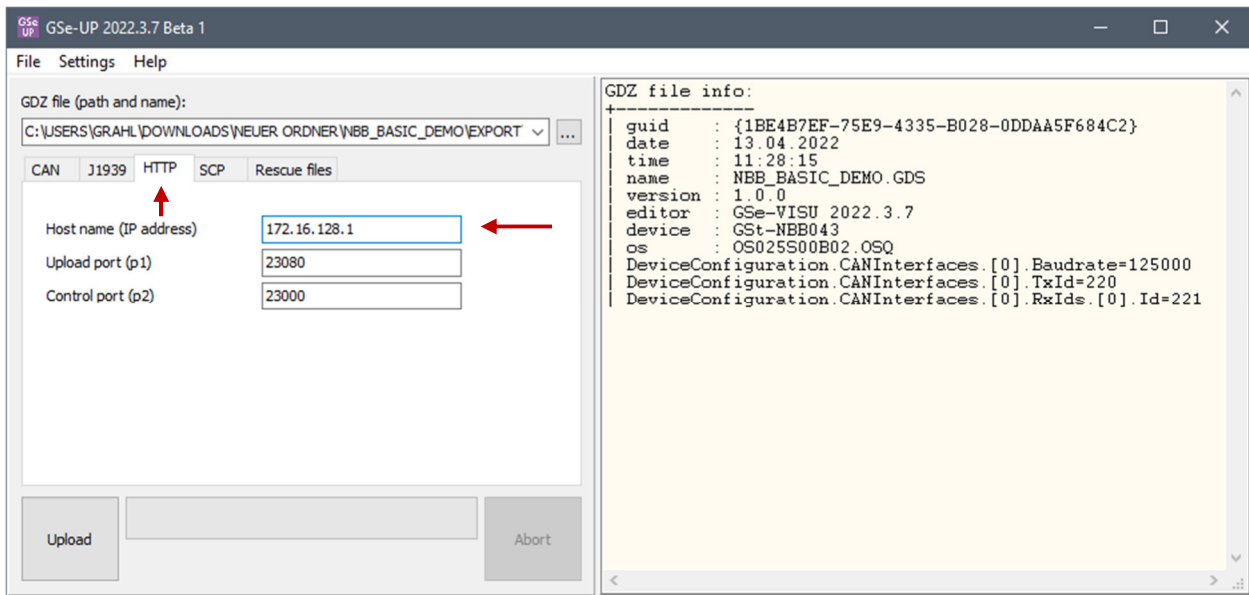


Figure 16 - Project upload window

## 5. Basic examples

### 5.1. Keys

The 8 keys on the display can be read with two functions.

```
int32_t IsKeyPressedNew(int32_t keyNum)
```

Returns != 0 if the key with the given number was pressed between the last UserC cycle and the current one. It is only save to call this function in the UserCCycle function.

Parameter

keyNum : Number of the key to be queried.

Return

> 0 if the requested key was newly pressed.

```
int32_t IsKeyDown(int32_t keyNum)
```

Returns > 0 if the key with the given number is pressed. The number returned represents the time from the last key press. This can be used to easily detect longer key presses.

Parameter

keyNum : Number of the key to be queried.

Return

> 0 if the key with the given number is pressed.

#### Note:

The numerical assignment of the keys is as followed:

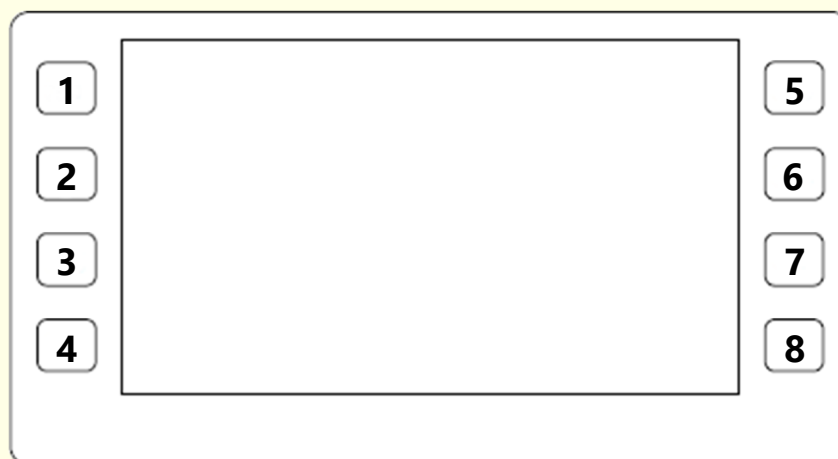


Figure 17 - Key to number assignment

Example to test if a key is pressed and execute some code, but only once:

```
#define KEY_F1 1

if (IsKeyPressedNew(KEY_F1))
{
    /* execute if the key F1 is pressed */
}
```

Example to detect a long key press:

```
#define KEY_F1 1

if (1000U > IsKeyDown(KEY_F1))
{
    /* do something on long press */
}
```

## 5.2. System language

The system language can be changed on the fly. To switch between the pre-defined languages, a system variable needs to be updated.

SYS_COLOR	65529				---	0	0
SYS_ENC_LEFT	65526				---	0	0
SYS_ENC_RIGHT	65525				---	0	0
SYS_LANGUAGE	65530	←			---	0	0
SYS_OPERATING_TIME	65531				---	0	0
SYS_PRIORITY	65528				---	0	0
SYS_TEMPERATURE	65527				---	0	0
SYS_U_SUPPLY	65524				---	0	0

Figure 18 - Variable table

### Note:

More information about text resources can be found in the GSe-VISO manual in chapter: Project resource -> Text resource.

Example:

```
#define LANG_GERMAN 0 /* default */
#define LANG_ENGLISH 1

SetVar(HDL_SYS_LANGUAGE, LANG_ENGLISH);
```

### 5.3. Switching masks

Switching between masks is done with the function `PrioMaskOn()`. Every mask that is created gets an object name like `MSK_<MASK_NAME>`.

Example to display the mask with the Name Mask\_F4:

```
PrioMaskOn(MSK_MASK_F4);
```

### 5.4. Backlight

To change the display backlight the function `SetDisplayBacklight()` has to be called.

```
int32_t SetDisplayBacklight(uint32_t n, uint32_t intens)
```

Sets the state of the backlight of a display.

Parameter

n : Number of the display. (0 for devices with only one display)

intens : A value between 0 and 1000 where 0 is lowest intensity and 1000 is highest.

Return

0 if the backlight was set successfully or < 0 if there was an error.

#### Note:

It is not possible to read the display backlight. Therefore it is recommended to save the backlight value in a static variable or even in a global variable.

Example:

```
/* read the current backlight value */
int32_t backlight = GetVar(HDL_DSPBACKLIGHT);

/* modify the backlight */

/* save the new backlight value */
SetVar(HDL_DSPBACKLIGHT, backlight);
/* apply the new backlight value */
SetDisplayBacklight(0, backlight);
```

### 5.5. Feedback

Depending on the project scope, feedback data from the receiver are automatically sent to the display. The data are then stored in global variables and can be accessed with the `GetVar()` function from the c code. These global variables can also be used directly in the graphic designer.

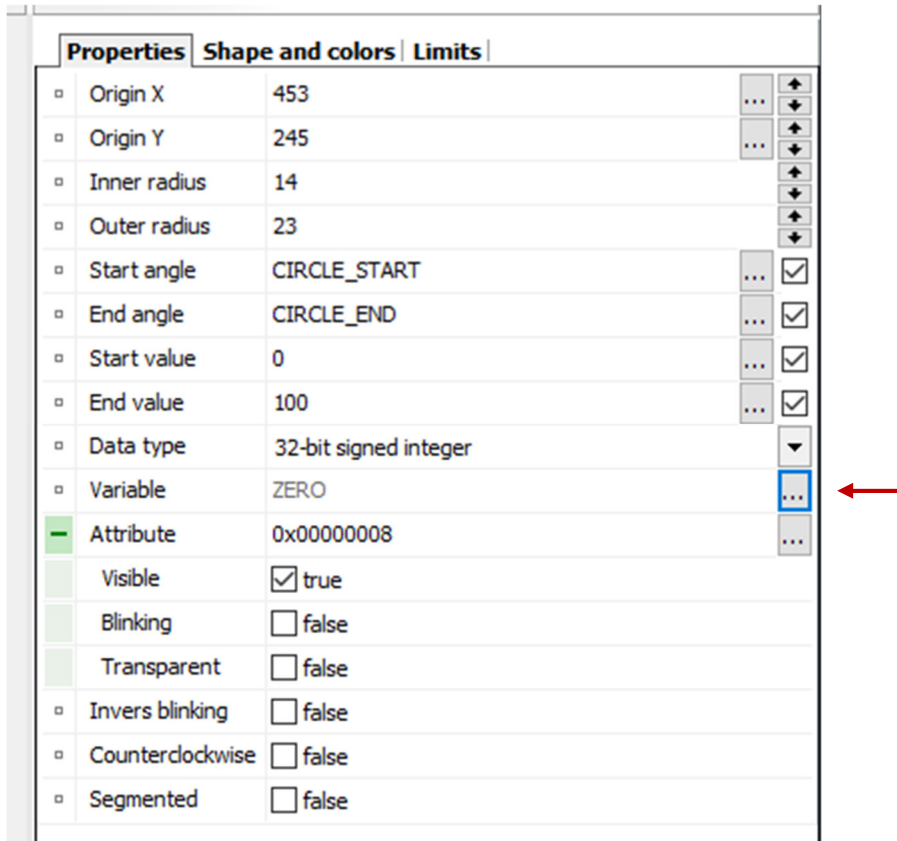


Figure 19 - Object properties