



# Smart Widgets: Compass

DOCUMENT DATE: **9<sup>th</sup> MAY 2020**  
DOCUMENT REVISION: **1.0**



## Description

This application note shows how to create a compass widget for Picaso, Diablo16 and Pixi display modules.

Before getting started, the following are required:

### Hardware

- Any [4D Systems display module](#) powered by any of the following processors:
  - o Picaso
  - o Diablo16
  - o Pixi28/44
- [Programming Adaptor for target display module](#)
- [uSD Card](#)
- USB Card Reader

### Software

- [Workshop4](#)
- This requires the **PRO** version of Workshop4

This application note comes with one (1) ViSi-Genie project.

**Note:** Using a non-4D programming interface could damage the processor and void the warranty.

## Content

<b>Description .....</b>	<b>2</b>
<b>Content .....</b>	<b>2</b>
<b>Application Overview.....</b>	<b>3</b>
<b>Setup Procedure .....</b>	<b>3</b>
<b>Create a New Project .....</b>	<b>3</b>
<b>Design the Project.....</b>	<b>3</b>
<i>Add Smart Gauge to Project .....</i>	<i>4</i>
<i>Open Smart Widgets Editor.....</i>	<i>4</i>
<i>Select Face Image .....</i>	<i>5</i>
<i>Adding Bottom Layer.....</i>	<i>6</i>
<i>Adding Needle Indicator.....</i>	<i>8</i>
<i>Setting Up Needle Layer Parameters.....</i>	<i>9</i>
<i>Add and Configure Input Object .....</i>	<i>12</i>
<b>Run the Program.....</b>	<b>13</b>
<b>Proprietary Information .....</b>	<b>14</b>
<b>Disclaimer of Warranties &amp; Limitation of Liability .....</b>	<b>14</b>

## Application Overview

The Smart Widget tool enables PRO version users to easily create custom widgets of their own design. It allows the user to create Sliders, Knobs and Gauges.

The purpose of this application note is to introduce the PRO version exclusive tool and to discuss how to create a compass widget (smart gauge). To demonstrate, this application note uses the ViSi-Genie environment.

## Setup Procedure

For instructions on how to launch Workshop4, how to open a **ViSi-Genie** project, and how to change the target display, kindly refer to the section “**Setup Procedure**” of the application note

- [ViSi-Genie Getting Started - First Project for Diablo16 Display Modules](#)
- [ViSi-Genie Getting Started - First Project for Picaso Displays](#)
- [ViSi-Genie Getting Started - First Project for Pixxi Display Modules](#)

## Create a New Project

For instructions on how to create a new **ViSi-Genie** project, please refer to the section “**Create a New Project**” of the application note

- [ViSi-Genie Getting Started - First Project for Diablo16 Display Modules](#)
- [ViSi-Genie Getting Started - First Project for Picaso Displays](#)
- [ViSi-Genie Getting Started - First Project for Pixxi Display Modules](#)

## Design the Project


For this application, gen4-uLCD-43DT will be used for the project. Same procedure is applicable for any touch screen Picaso and Diablo16 displays.

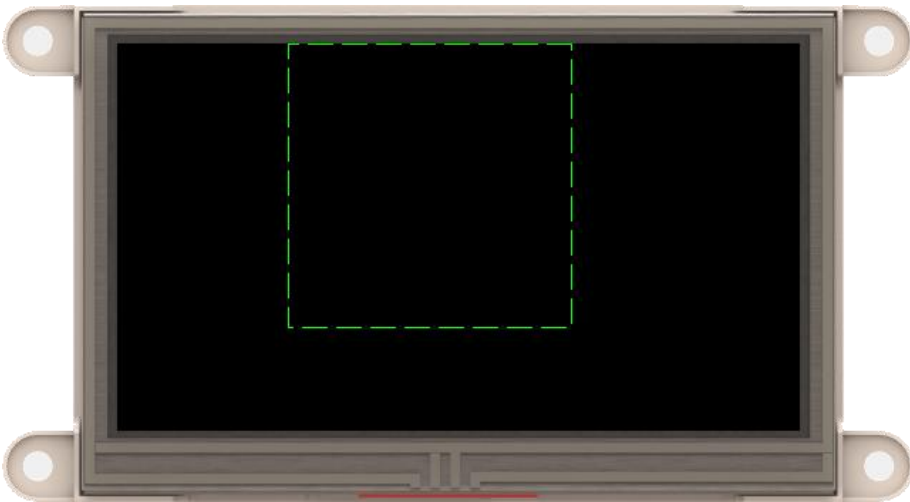
## Add Smart Gauge to Project

A compass widget is obviously an output object. It is best suited that this widget be made as a Smart Gauge.

Add a Smart Gauge widget to your ViSi-Genie project. It can be found on the Inputs tab on the Widgets Pane.





Simply click on this icon  to select it. Then place it on the WYSIWYG area.



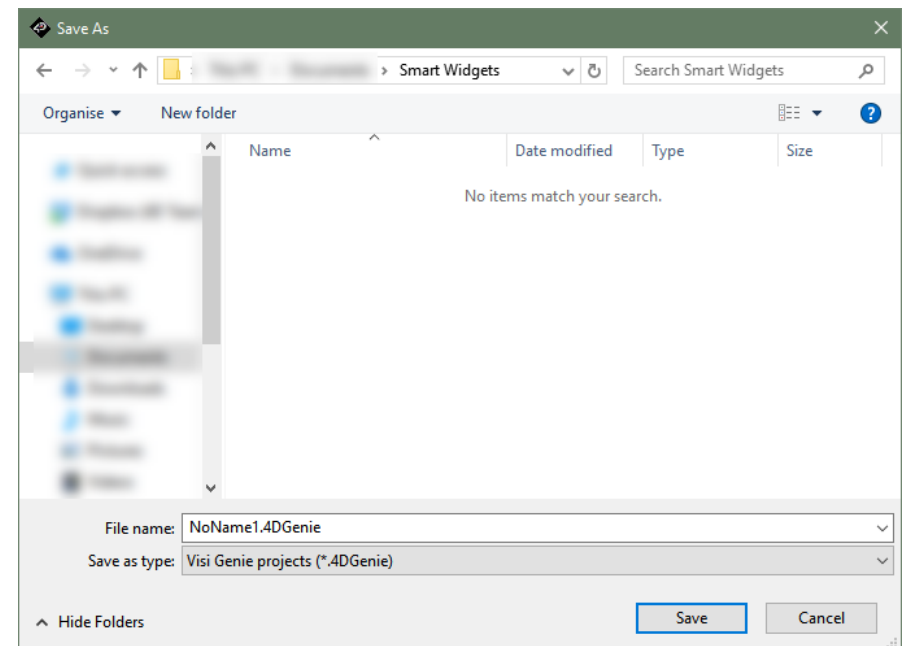
As displayed on the previous image, the widget appears empty when placed in the WYSIWYG.

## Open Smart Widgets Editor

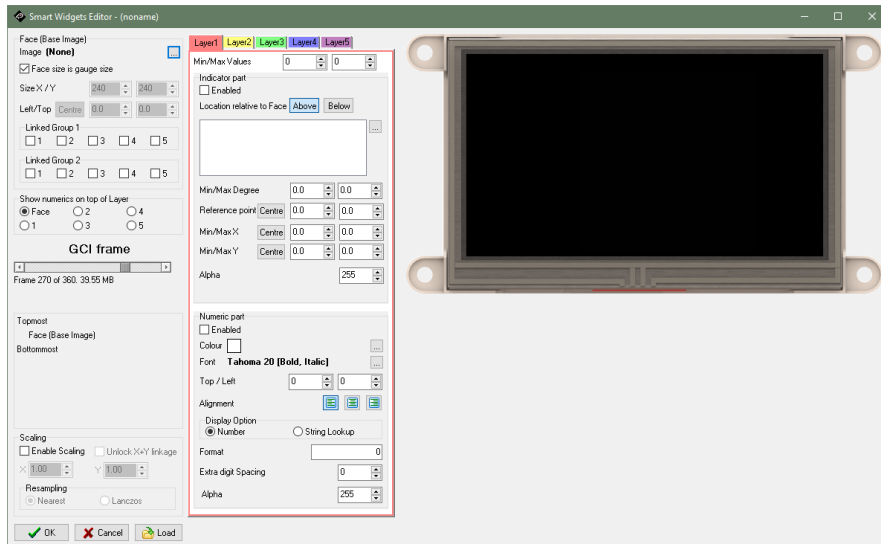
Open the Smart Widgets Editor tool by clicking on  of **Config** in the Object Inspector Properties tab.

Property	Value
Name	SmartGauge0
Alias	SmartGauge0
Config	(None) 
Left	116
Top	20

The tool requires that the project is already saved before the tool opens. Therefore, since on this case, it hasn't been saved yet, Workshop4 will automatically prompt to save.



Save the project to desired location. The tool will open after the project has been saved.




As shown in the image, this tool has a lot of parts. The next steps will focus only on the minimum tool functionalities required to make a basic slider.

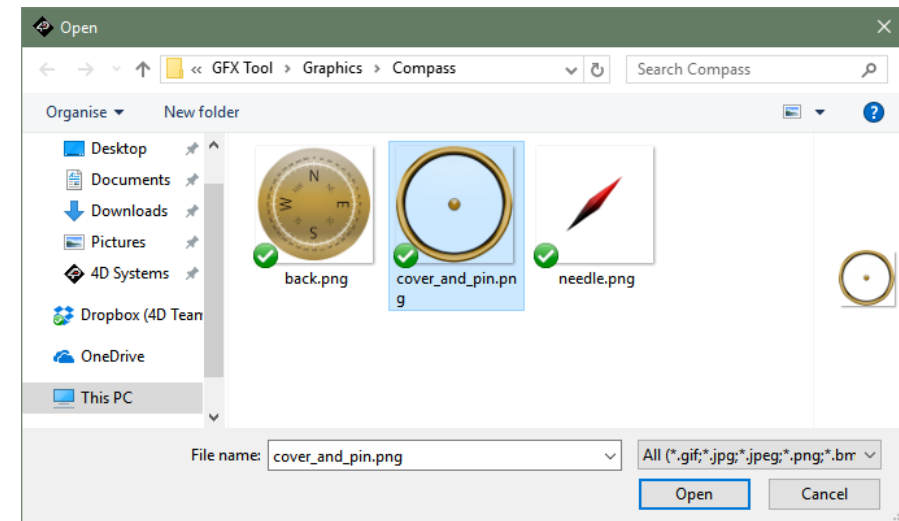
For detailed discussions on how each part works, please refer to the **Smart Widgets Editor User Guide**.

## Select Face Image

The first step when creating a smart widget is to select the Face (base image).



Click  to select an image.



After selecting a base image, it will be displayed in the preview area.



The size of the output widget depends on the size of this image by default.



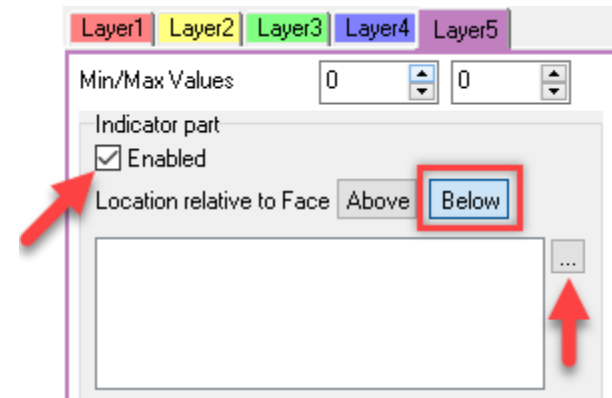
Any part of any image used in a smart widget will not be displayed if outside this **working area**.

## Adding Bottom Layer

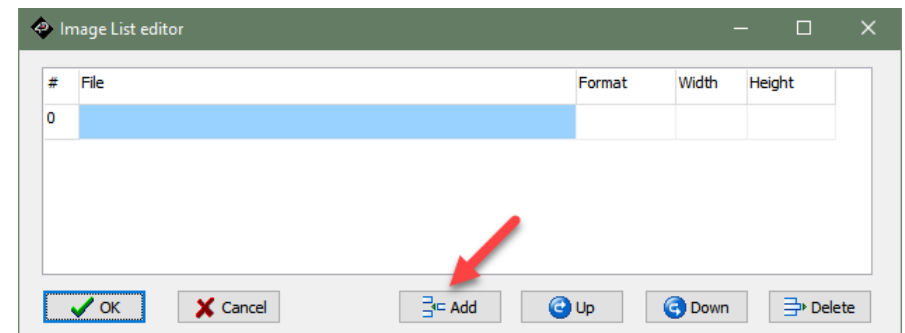
In case the form has a background image or color, it will also be seen through the any hollow part. If this is not desirable, a bottom layer is required.

When adding the bottommost layer, it is easiest to use Layer 5.

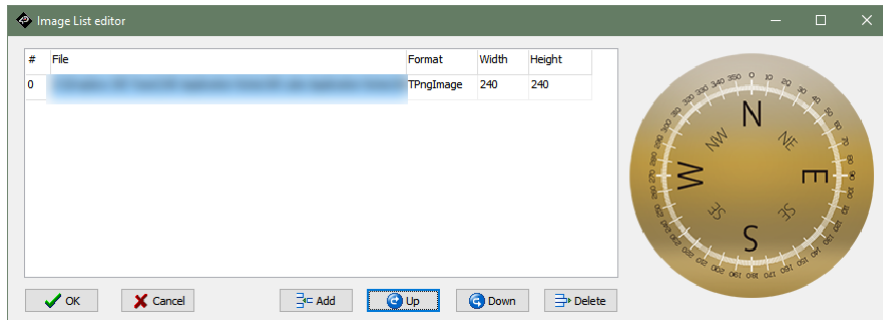
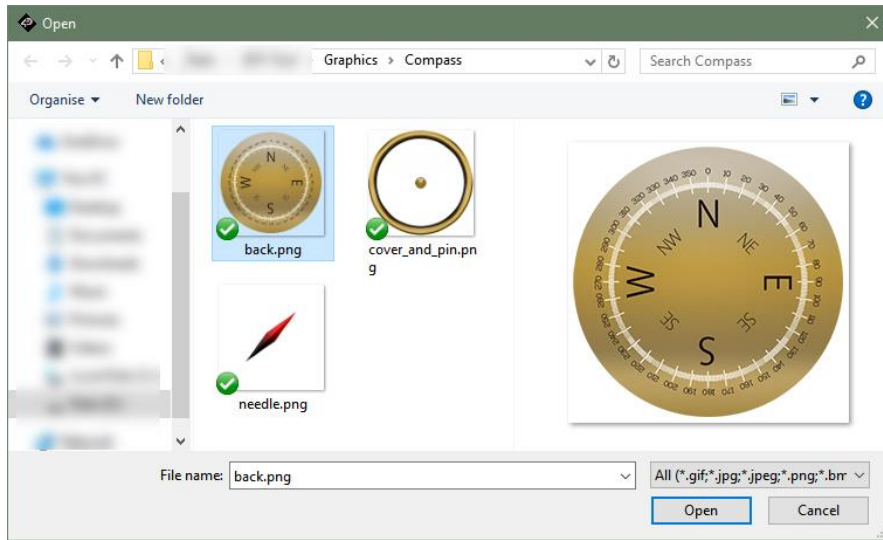
Enable Layer 5 and add the bottom layer image similar to how the needle layer was added.



Obviously, the bottom layer is below the face image.



Select the image and click **Open**



Click **OK**

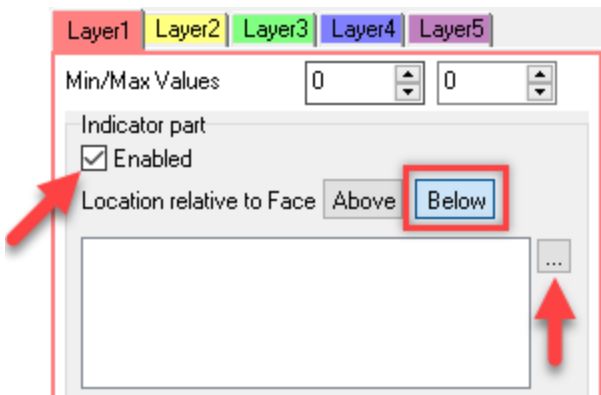


After selecting the face image and placing the bottommost image, a compass still needs a needle.

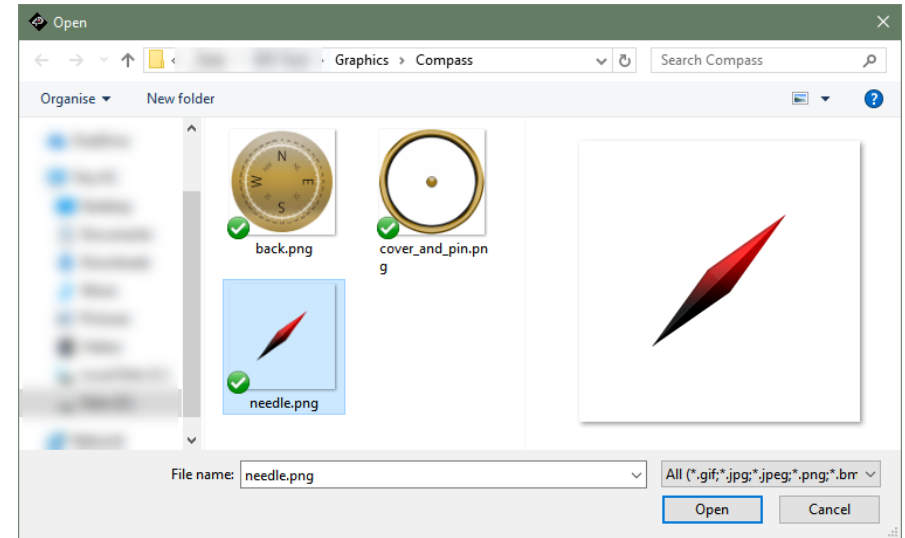
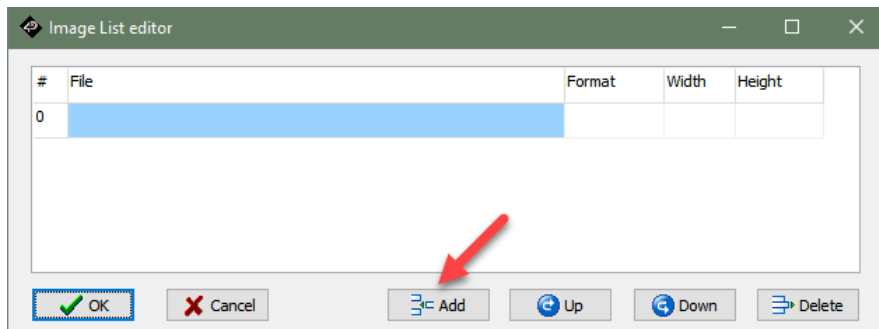
## Adding Needle Indicator

In this project, a Layer 1 will manipulate the needle.

Enable Layer 1.



Add compass needle



Select the needle image and click **Open**





## Setting Up Needle Layer Parameters



After setting up the layers, give a range of values to Layer 1 (needle layer). For this compass, Layer 1 will be given 0 to 359 to cover for a whole 360 degrees.

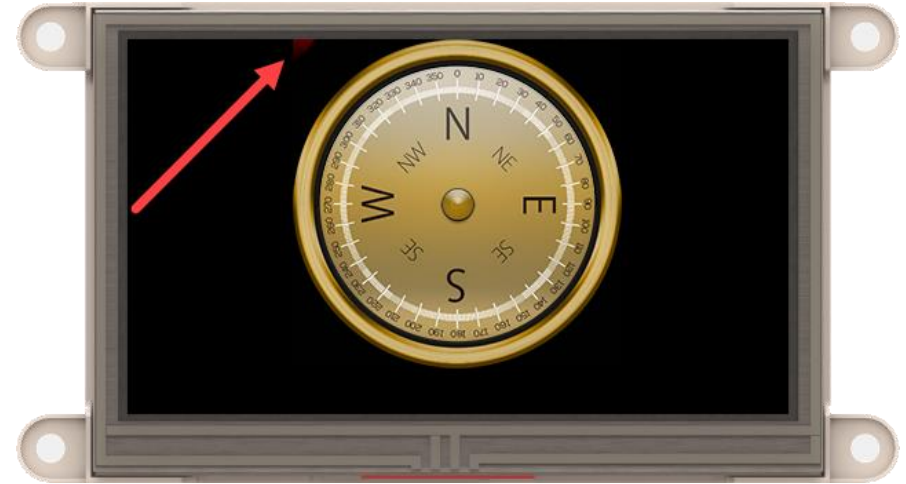
Min/Max Values

Afterwards, we need to set the point of rotation.

Rotation point

For this, the point of rotation of the image is at (120, 120) this is the pixel position on the needle image itself, not on the preview area.

By setting the rotation point, it can be noticed that the needle is almost entirely outside the working area.

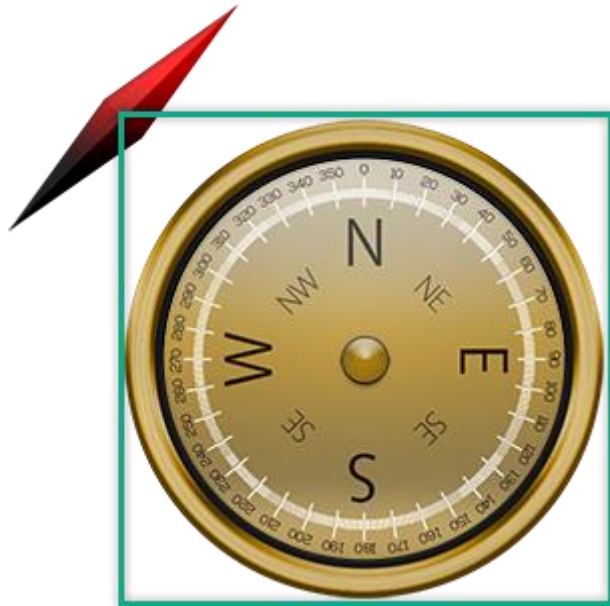


This is because **Min/Max X** and **Min/Max Y** are both set to 0.

Min/Max X     
 Min/Max Y

This makes the rotation point of the needle image (120, 120) be located at (0, 0) of the preview area.

To illustrate:



This needs to be located on the right position on the preview area. In this application note, the position of the rotation point with respect to the preview area is also (120, 120).

**Min/Max X** and **Min/Max Y** needs to be set to constant values.

Min/Max X	Centre	120.0	120.0
Min/Max Y	Centre	120.0	120.0

Both the **Min/Max X** and while **Min/Max Y** need to be at 120. This will make the X and Y positions of the needle's rotation point be constant from minimum value to maximum value.

You'll notice that the needle is back on its original position.



Now, to create frames that rotates the needle, an initial angular position (**Min Degree**) and final angular position (**Max Degree**) must both be specified.

To initially point to the North (upwards), since the needle is off by 45 degrees clockwise, the **Min Degree** needs to be set to -45. Since this is a 360 degrees needle rotation. **Max Degree** needs to be set to:

$$360 - \text{Min Degree} - 1$$

Therefore:

Min/Max Degree	-45.0	314.0
----------------	-------	-------

Here are some frames from the widget:

At 1<sup>st</sup> frame (index 0):



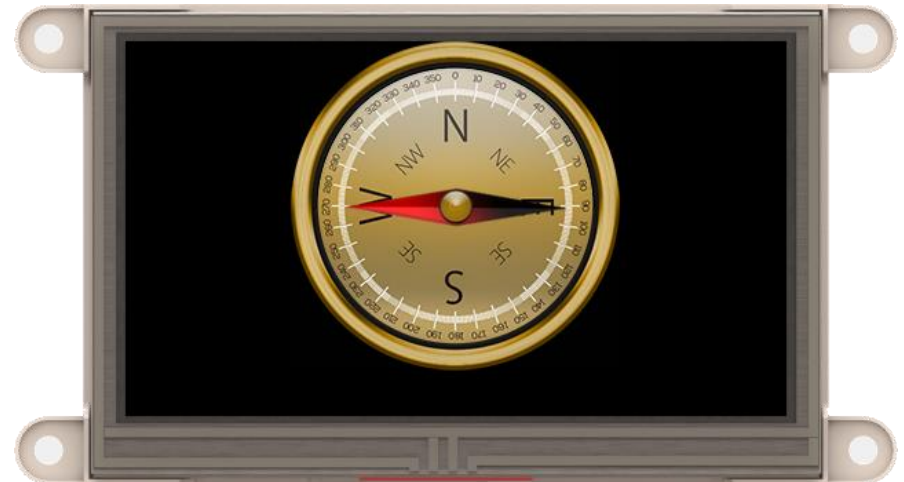
At 181<sup>st</sup> frame (index 180):



At 91<sup>st</sup> frame (index 90):



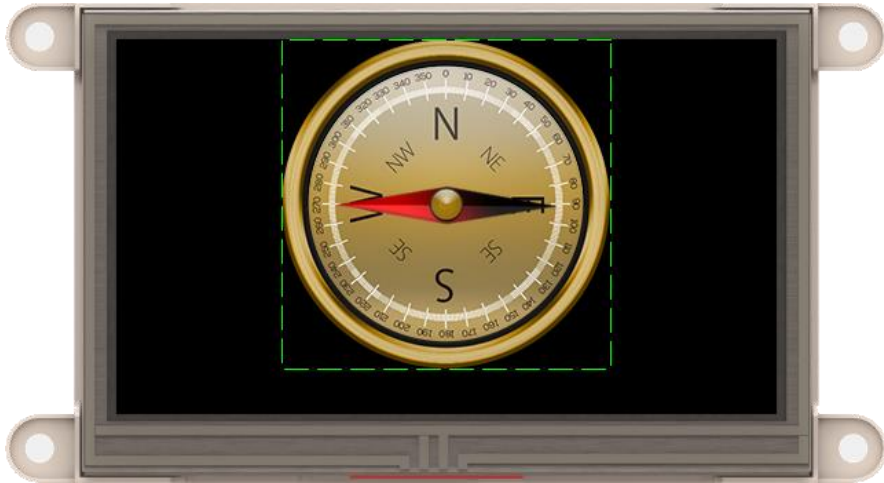
At 271<sup>st</sup> frame (index 270):



**Note:** Notice that an increase from Min to Max Degrees results to a clockwise rotation. Obviously, a decrease would result in a counterclockwise rotation

Press **OK** to continue.

The widget will be displayed on the WYSIWYG area. Drag the widget to your desired position.



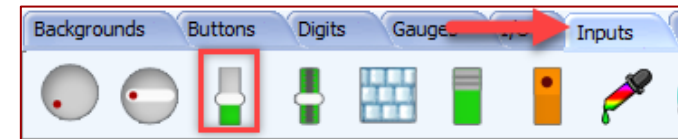
The compass is now ready to be used in the project. This simply acts like a regular gauge.

## Add and Configure Input Object

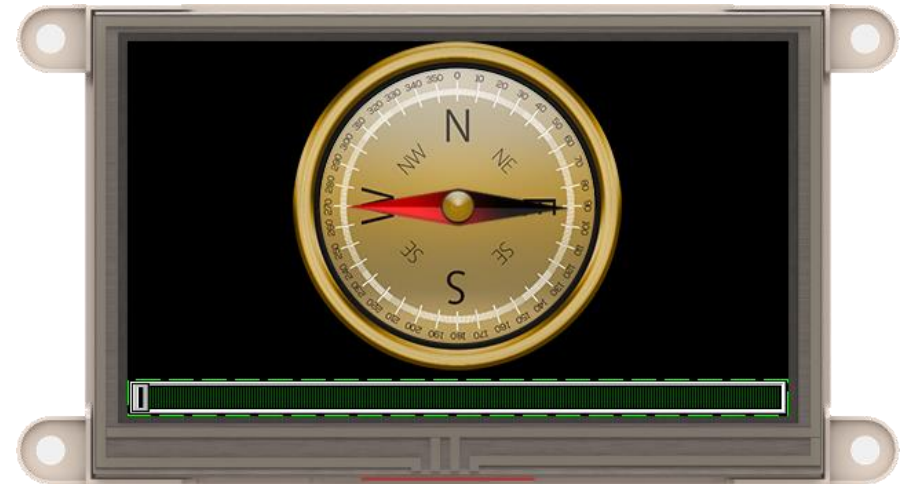
In this application note project, an input object will be used to display the individual frames of the compass for simplicity.

Take note that the purpose of this appnote is only to discuss how to make a compass widget.

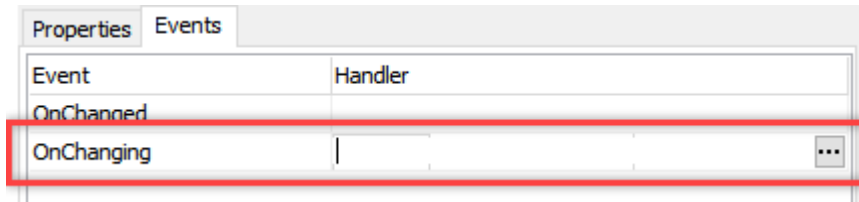
Add a Slider to the project.



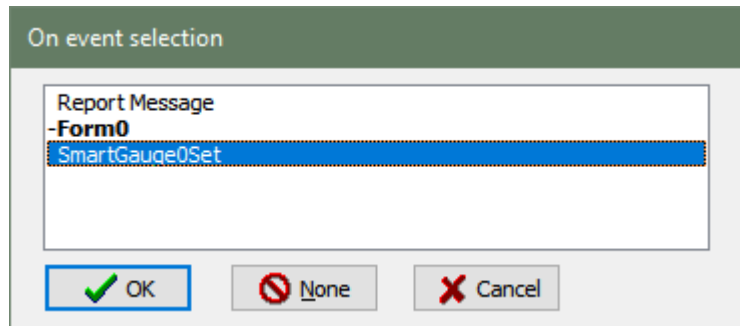
It is important that the Slider has the same number of frames as the compass. With values 0 to 359 for both the slider and compass, they have same number of frames.



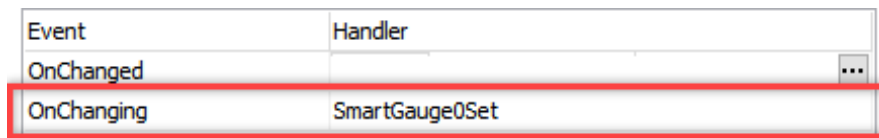
Then for the Slider, under Events tab of the Object Inspector:



Set the event for **OnChanging**:



Select **SmartGauge0Set** and click on **OK**.



The project is now ready for upload.

## Run the Program

For instructions on how to save a **ViSi-Genie** project, how to connect the target display to the PC, how to select the program destination, and how to compile and download a program, please refer to the section “**Run the Program**” of the application note

- [ViSi-Genie Getting Started - First Project for Diablo16 Display Modules](#)
- [ViSi-Genie Getting Started - First Project for Picaso Displays](#)
- [ViSi-Genie Getting Started - First Project for Pixxi Display Modules](#)

## Proprietary Information

The information contained in this document is the property of 4D Systems Pty. Ltd. and may be the subject of patents pending or granted, and must not be copied or disclosed without prior written permission.

4D Systems endeavours to ensure that the information in this document is correct and fairly stated but does not accept liability for any error or omission. The development of 4D Systems products and services is continuous and published information may not be up to date. It is important to check the current position with 4D Systems.

All trademarks belong to their respective owners and are recognised and acknowledged.

## Disclaimer of Warranties & Limitation of Liability

4D Systems makes no warranty, either expresses or implied with respect to any product, and specifically disclaims all other warranties, including, without limitation, warranties for merchantability, non-infringement and fitness for any particular purpose.

Information contained in this publication regarding device applications and the like is provided only for your convenience and may be superseded by updates. It is your responsibility to ensure that your application meets with your specifications.

In no event shall 4D Systems be liable to the buyer or to any third party for any indirect, incidental, special, consequential, punitive or exemplary damages (including without limitation lost profits, lost savings, or loss of business opportunity) arising out of or relating to any product or service provided or to be provided by 4D Systems, or the use or inability to use the same, even if 4D Systems has been advised of the possibility of such damages.

4D Systems products are not fault tolerant nor designed, manufactured or intended for use or resale as on line control equipment in hazardous environments requiring fail – safe performance, such as in the operation of nuclear facilities, aircraft navigation or communication systems, air traffic control, direct life support machines or weapons systems in which the failure of the product could lead directly to death, personal injury or severe physical or environmental damage ('High Risk Activities'). 4D Systems and its suppliers specifically disclaim any expressed or implied warranty of fitness for High Risk Activities.

Use of 4D Systems' products and devices in 'High Risk Activities' and in any other application is entirely at the buyer's risk, and the buyer agrees to defend, indemnify and hold harmless 4D Systems from any and all damages, claims, suits, or expenses resulting from such use. No licenses are conveyed, implicitly or otherwise, under any 4D Systems intellectual property rights.