4D LABS SEMICONDUCTORS

Workshop4

Integrated Development Environment

Workshop4 Widgets Reference Manual

Document Revision: 1.1

Document Date: 23rd July 2020

Workshop4 Widgets Reference Manual

Table of Contents

1. Introduction to Workshop4 Widgets

Workshop4 provides a drag n' drop design tool for most of its powerful development environments. This allows user to easily utilise three types of widgets in their projects:

- GCI Widgets
- Internal Functions (PmmC) Widgets
- Inherent Widgets

This offers increased flexibility in creating applications depending on the hardware requirement. These widgets include the following: buttons, sliders, knobs, gauges, LED indicator, LED digit, strings, static labels, media, and images.

Additional non-GUI widgets are also available that allows the user to add miscellaneous functionalities like sound, hardware I/O, or supporting resources like fonts, depending on the target processor and environment used.

All these widgets are available in the Widgets Menu of the ViSi and ViSi-Genie environment of Workshop4 IDE. Given its high configurability, these widgets can be customized directly on the What You See Is What You Get (WYSIWYG) screen or through the **Object Inspector**.

This document will cover all type of widgets listed above**.**

2. GCI Widgets

The **GCI Widgets** also known as Graphics Composer Image (GCI) widgets, are widgets that are generated by the graphics composer program used in Workshop4 IDE. A GCI widget is composed of a set of frames representing each value.

During project compilation, Workshop4 creates a GCI (*.gci or *.gcf) file which is a collection of GCI widgets sometimes along with a DAT (*.dat) file. These files are copied to a microSD (or external flash for Flash-based Pixxi module) to be accessible to the processor for rendering on the display during runtime. This widget type is ideal for applications that only requires heavy graphics.

These widgets are available for Goldelox, Picaso, Diablo and Pixxi processors.

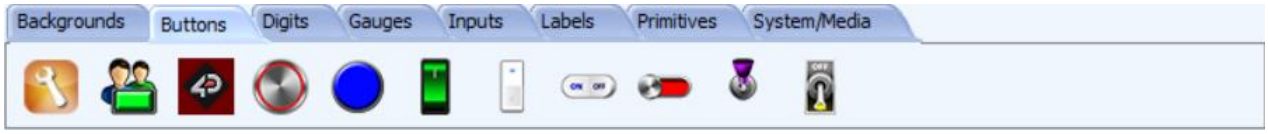
When using Pixxi processor in Flash mode, only a few small GCI widgets can fit in the external flash. Therefore, it is only advisable to be used for icons and small screen backgrounds.

Here are some sample application screens utilizing GCI widgets:

The following subsections show examples of these widgets.

2.1. GCI Button Widgets

These are widgets commonly used for receiving touch inputs from the user.



The following widgets can be selected from the Buttons Pane by selecting their respective icon.

Fancy Button

User Button

Animated Button

Button01

Button02

Rocker01

Rocker02

Rocker03

Slider01

Toggle01

Toggle02

2.1.1. GCI Fancy Button

The Fancy Button widget is shown in the diagram below.

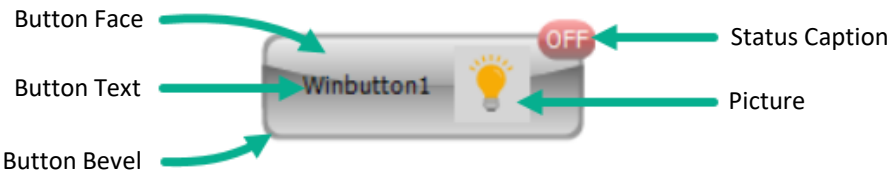


Figure 1: Fancy Button Diagram

Property	Description
Name	Widget name
Alias	Widget alias
Appearance	Button appearance options
Alignment	Button text position
Layout	Picture location
PictureAlignment	Picture alignment
SimpleLayout	Disable button face colour gradient
Spacing	Button text spacing from picture
AutoSizeToPicture	Resize button to the picture size
Bevel	Enable or disable Button Bevel
Bevel Color	Button Bevel
Caption	Button text string
Color	Button main colour
Font	Button text font options
Height	Widget height
Left	Top-left X-position
Matrix	Button grouping (ViSi-Genie only)
Momentary	Button type
Picture	Image selection for button picture
StatusWhenOff	Status caption options for the initial state
BGcolor	Status caption background colour
Caption	Status caption text string
Glow	Enable or disable glow effect
Font	Status caption text font options
Visible	Show or hide status caption
StatusWhenOn	Status caption options for the second state
BGcolor	Status caption background colour
Caption	Status caption text string
Glow	Enable or disable glow effect
Font	Status caption text font options
Visible	Show or hide status caption
Top	Top-left Y-position
Width	Widget width

2.1.2. GCI User Button

The User Button widget properties are listed below.

Figure 2: User button widget example

Property	Description
Name	Widget name
Alias	Widget alias
Height	Widget height
Images	Button states image selection (Opens Image List Editor)
Left	Top-left X-position
Matrix	Button grouping (ViSi-Genie only)
Momentary	Button type
Stretch	Enable or disable widget scaling
Top	Top-left Y-position
Width	Widget width

Note: When this widget is added, it will be initially empty. The user can add images for the button state through the Images property in the Object Inspector pane.

2.1.3. GCI Animated Button

The Animated Button widget properties are listed below.

Figure 3: Animated button widget example

Property	Description
Name	Widget name
Alias	Widget alias
Height	Widget height
Images	Button states images (Opens Image List Editor)
Interval	Animation frame animation speed (ViSi-Genie only)
Left	Top-left X-position
Matrix	Button grouping (ViSi-Genie only)
Momentary	Button type
Stretch	Enable or disable widget scaling
Top	Top-left Y-position
Width	Widget width

Note: When this widget is added, it will be initially empty. The user can add images for the button state through the Images property in the Object Inspector pane.

2.1.4. GCI 4D Button

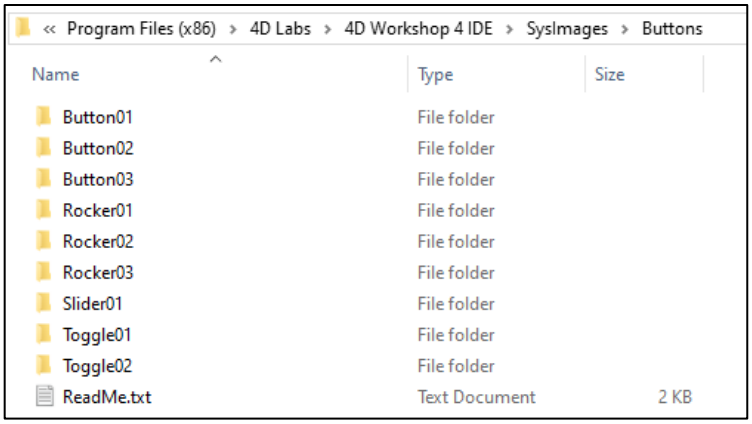
The default 4D Button widgets are shown below. Users can add custom buttons under this category.

Figure 4: 4D Buttons

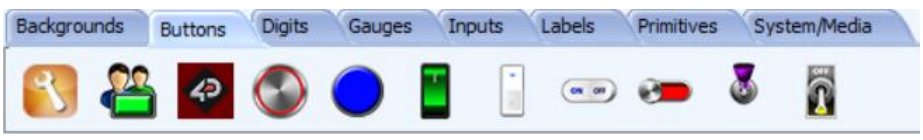
Property	Description
Name	Widget name
Alias	Widget alias
Height	Widget height
Left	Top-left X-position
Matrix	Button grouping (ViSi-Genie only)
Momentary	Button action type
Size	Set button size
Style	Button styles
Top	Top-left Y-position
Type	Button design type
Width	Widget width

Adding Custom GCI Buttons in Workshop4

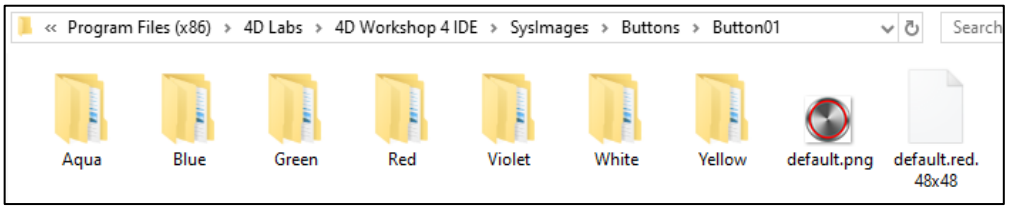
New buttons for the Buttons Pane can be created by adding files with the specified formatting into the dedicated directory inside the Workshop4 program located at \4D Labs\4D Workshop 4 IDE\SysImages\Buttons\



Within this directory are the button '**types**' folder. All **'types'** folders containing valid contents will appear in the Buttons pane in Workshop4.

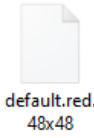


Inside the '**types**' folder are the following: preview image, config file and style folders

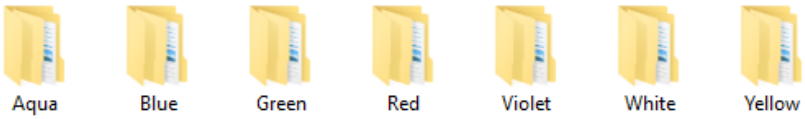


The preview image in the Button tab for the new button is provided by the image file named **'default.png'**. The image file can only have a maximum dimension of 32 by 32 pixels.

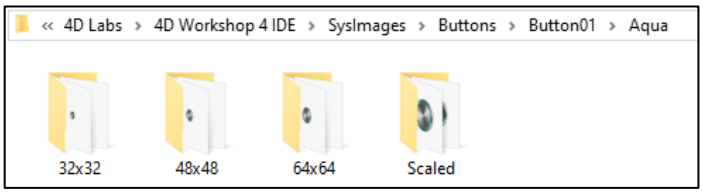
The config file specifies the default style and size of the button widget when it is added to the WYSIWYG screen. This file has a filename format: **default.[style].[size]**, (i.e. "default.blue.32x32", this sets the default button to have the blue style and use the images in the 32x32 size folder). If this file is not present, Workshop4 will use the first style and size as the default.



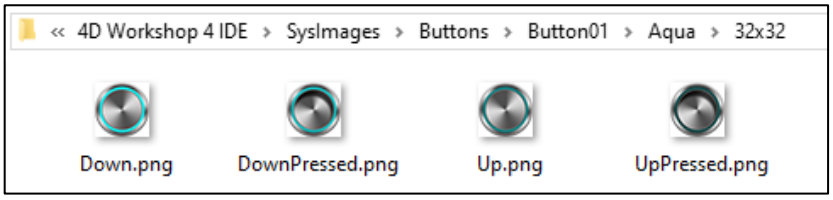
There could be one or more **'style'** folder for each button type. Each valid **'style'** folder will have a corresponding entry in the **Style** property of the object inspector.



A '**style**' folder can have one or more **'size'** folders representing the size of the custom buttons for that particular style. Each folder in this directory should have the name format **[Width]x[Height]** (i.e. 32x32 for button having a size of 32 by 32 pixels). This will create a corresponding entry in the **Size** property of the object inspector. An optional folder called **'Scaled'** can be created in order to add a resizable button, the images have a maximum dimension of 128 by 128 pixels.



Inside each **'size'** folder are the images that will be used as the button state frames. There should be four .png image files named **'Up.png'**, **'UpPresssed.png'**, **'Down.png'** and **'downpressed.png'** in this folder. These images will be used by Workshop4 to create the state frames of a button.



2.2. GCI Digits Widget

These are widgets commonly used for displaying values or status information.

The following widgets can be selected from the Digits Pane by selecting their respective icon.

LED Digits

LED

Custom Digits

User LED

2.2.1. GCI LED Digits

The LED Digits widget is shown in the diagram below.

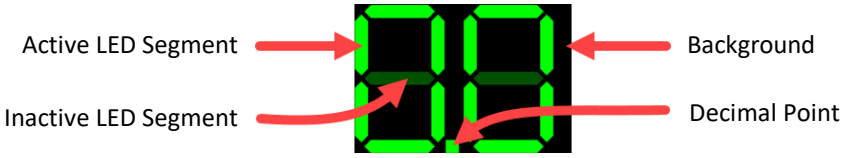


Figure 5: LED Digits Diagram

Property	Description
Name	Widget name
Alias	Widget alias
Color	Active LED segment colour
Decimals	Decimal Place relative to separator
Digits	Total number of LED digit
Height	Widget height
LeadingZero	Enable or disable leading zeroes
Left	Top-left X-position
OutlineColor	LED segment outline colour
Palette	LED segment colour
High	Active LED Segment colour
Low	Inactive LED Segment colour
Top	Top-left Y-position
Width	Widget width

2.2.2. GCI Custom Digits

The Custom Digits widget properties are listed below.

Figure 6: Custom LED digits example

Shown below are the sample bitmap images required for creating a custom digit widget. In creating the bitmap image each digit must have equal width and height, as this will be divided into the individual digit.

Figure 7: Bitmap Image for unsigned custom digit.

Figure 8: Bitmap Image for signed custom digit.

Property	Description
Name	Widget name
Alias	Widget alias
Bitmap	Custom LED digit bitmap image
Digits	Total number of LED digit
LeadingZero	Enable or disable leading zeroes
Left	Top-left X-position
Top	Top-left Y-position

2.2.3. GCI LED

The LED widget is shown in the diagram below.

Figure 9: LED Diagram

Property	Description
Name	Widget name
Alias	Widget alias
Active	Set LED initial state to active
Caption	Caption text string
Color	Background colour
Font	Button text font options
Glyphs	Custom LED type image (Conjoined image containing two states)
Height	Widget height
Layout	LED position
LedType	LED type (Custom, LargeRect, Rectangle, Rounded)
Left	Top-left X-position
Palette	LED colour
High	Active LED colour
Low	Inactive LED colour
Spacing	Digit size
Top	Top-left Y-position
Width	Widget width

2.2.4. GCI User LED

The User LED widget is shown in the diagram below.

Figure 10: User LED diagram

Property	Description
Name	Widget name
Alias	Widget alias
Active	Set LED initial state to active
Bevel	Bevel options
BorderColor	Border colour
BorderWidth	Border width
InnerColor	Inner Space colour
InnerHighlight	Inner Bevel highlight colour
InnerOutline	Inner Bevel outline colour
InnerShadow	Inner Bevel shadow colour
InnerSpace	Inner Space width
InnerStyle	Inner Bevel style
InnerWidth	Inner Bevel width
OuterColor	Outer Space colour
OuterHighlight	Outer Bevel highlight colour
OuterOutline	Outer Bevel outline colour
OuterShadow	Outer Bevel shadow colour
OuterSpace	Outer Space width
OuterStyle	Outer Bevel style
Outerwidth	Outer Bevel width
Visible	Enable or disable bevel
Height	Widget height
Left	Top-left X-position
OutlineColor	Outline colour
OutlineWidth	Outline width
PaletteEx	LED colour
High1	Active LED lamp gradient start colour
High2	Active LED lamp gradient end colour
Low1	Inactive LED lamp gradient start colour
Low2	Inactive LED lamp gradient end colour
Top	Top-left Y-position
Width	Widget width

2.3. GCI Gauge Widgets

These are widgets commonly used for displaying values.

The following widgets can be selected from the Gauges Pane by selecting their respective icon.

Meter

Tank

Gauge

Spectrum

Angular Meter

Scope

Cool Gauge

Smart Gauge

Thermometer

Note: Smart Widgets are only enabled when a Workshop4 Pro License is purchased.

2.3.2. GCI Meter

The Meter widget is shown in the diagram below.

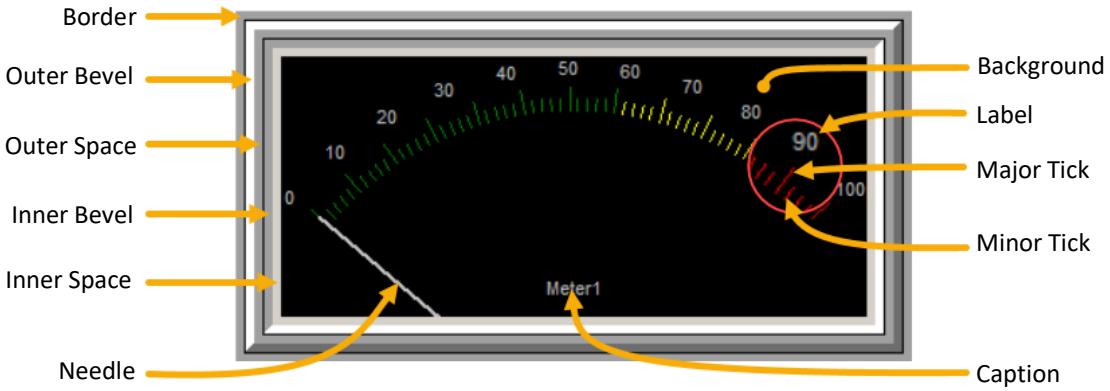


Figure 11: Meter Diagram

Property	Description
Name	Widget name
Alias	Widget alias
Angle	Start and End angle offset from horizontal
BackImage	Background image
BGcolor	Background colour (Not applicable if background image is present)
Bevel	Bevel options
BorderColor	Border colour
BorderWidth	Border width
InnerColor	Inner Space colour
InnerHighlight	Inner Bevel highlight colour
InnerOutline	Inner Bevel outline colour
InnerShadow	Inner Bevel shadow colour
InnerSpace	Inner Space width
InnerStyle	Inner Bevel style
InnerWidth	Inner Bevel width
OuterColor	Outer Space colour
OuterHighlight	Outer Bevel highlight colour
OuterOutline	Outer Bevel outline colour
OuterShadow	Outer Bevel shadow colour
OuterSpace	Outer Space width
OuterStyle	Outer Bevel style
Outerwidth	Outer Bevel width
Visible	Enable or disable bevel
Caption	Caption text string
Color	Background colour
Font	Caption font options
Height	Widget height
LabelOffsetX	Label offset distance from centre point
LabelOffsetY	Label offset distance from centre point

Property	Description
Labels	Total number of labels
Left	Top-left X-position
Maxvalue	Maximum value
Minvalue	Minimum value
NeedleColor	Needle colour
NeedleWidth	Needle width
Scale	Scale options
Color1	Colour of scale partition 1
Color2	Colour of scale partition 2
Color3	Colour of scale partition 3
Enlarge	Number of partitions between each major tick
Heightmax	Length of the major ticks
Heightmin	Length of the minor ticks
Percent1	Scale percentage of partition 1
Percent2	Scale percentage of partition 2
Ticks	Total number of ticks
Visible	Enable or disable scale
Spacing	Spacing distance from bevel
Top	Top-left Y-position
TotalTicks	Total number of partitions between each ticks
Width	Widget width

2.3.3. GCI Gauge

The Gauge widget is shown in the diagram below.

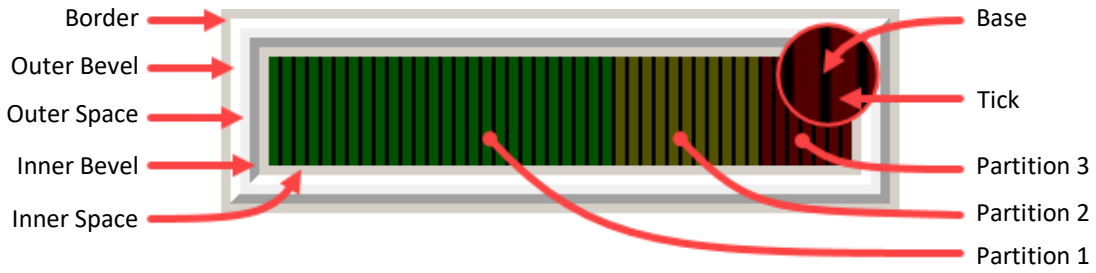


Figure 12: Gauge Diagram

Property	Description
Name	Widget name
Alias	Widget alias
Bevel	Bevel options
BorderColor	Border colour
BorderWidth	Border width
InnerColor	Inner Space colour
InnerHighlight	Inner Bevel highlight colour
InnerOutline	Inner Bevel outline colour
InnerShadow	Inner Bevel shadow colour
InnerSpace	Inner Space width
InnerStyle	Inner Bevel style
InnerWidth	Inner Bevel width
OuterColor	Outer Space colour
OuterHighlight	Outer Bevel highlight colour
OuterOutline	Outer Bevel outline colour
OuterShadow	Outer Bevel shadow colour
OuterSpace	Outer Space width
OuterStyle	Outer Bevel style
Outerwidth	Outer Bevel width
Visible	Enable or disable bevel
Color	Base colour
Height	Widget height
Left	Top-left X-position
Maxvalue	Maximum value
Minvalue	Minimum value
Orientation	Gauge orientation
Palette1	Partition 1 ticks
High	Active tick colour
Low	Inactive tick colour
Palette2	Partition 2 ticks
High	Active tick colour
Low	Inactive tick colour
Palette3	Partition 3 ticks

Property	Description
High	Active tick colour
Low	Inactive tick colour
Percent1	Percentage of partition 1
Percent2	Percentage of partition 2
Spacing	Tick spacing
Step	Step value
Style	Gauge direction
Tickheight	Tick width
Top	Top-left Y-position
Width	Widget width

2.3.4. GCI Angular Meter

The Angular Gauge widget is shown in the diagram below.

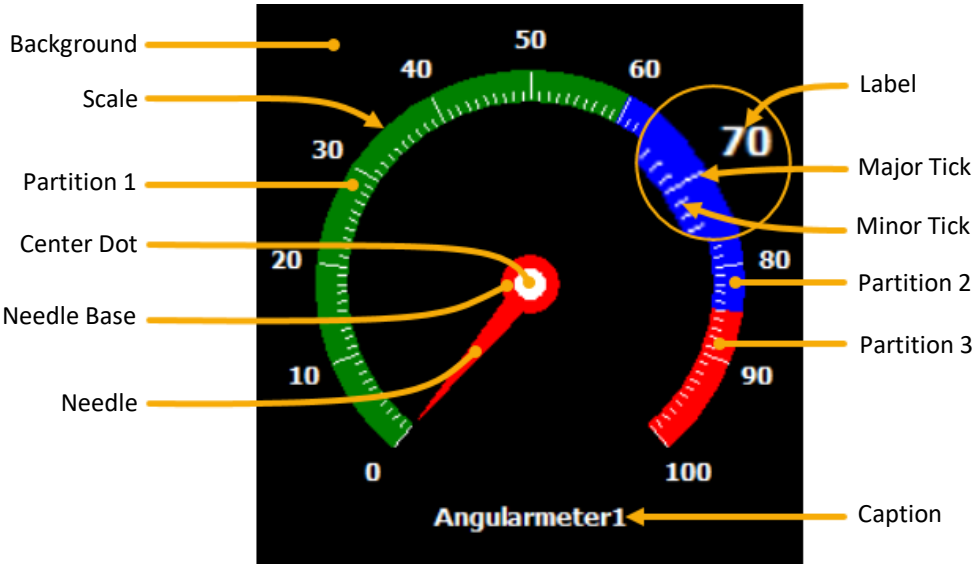


Figure 13: Angular Meter Diagram

Property	Description
Name	Widget name
Alias	Widget alias
Angle	Gauge starting angle
AngleOffset	Gauge end angle offset (in reference to the starting angle)
BaseColor	Gauge base colour
Caption	Caption text string
CaptionXOffset	Caption horizontal offset
CaptionYOffset	Caption vertical offset
CenterDotColor	Denter dot colour
CenterDotWidth	Center dot width
Color	Background colour
ColorZone1	Colour of scale partition 1
ColorZone2	Colour of scale partition 2
ColorZone3	Colour of scale partition 3
Decimals	Number of decimals in the labels
Font	Caption font options
Height	Widget height
Labels	Total number of labels
LabelsFont	Label font options
LabelsOffset	Label offset distance from the scale
Left	Top-left X-position
Maxvalue	Gauge maximum value
Minvalue	Gauge minimum value
NeedleBasewidth	Needle base width
NeedleColor	Needle base and Needle colour

NeedleLength	Needle length (Use -1 for automatic scaling)
Percent1	Percentage of partition 1
Percent2	Percentage of partition 2
Radius	Scale inner radius (Use -1 for automatic scaling)
Spacing	Spacing between scale inner circumference and widget boundary
Ticks	Total number of ticks
TicksColor	Tick colour
TicksEnlarge	Number of partitions between each major tick
TicksMax	Length of the major ticks
TicksMin	Length of the minor ticks
Top	Top-left Y-position
Width	Widget width

2.3.5. GCI Cool Gauge

The Cool Gauge widget is shown in the diagram below.

Figure 14: Cool Gauge Diagram

Property	Description
Name	Widget name
Alias	Widget alias
Arc	Arc options
Color	Arc colour
Opacity	Arc opacity value (0 to 255)
StartAngle	Arc start angle
StopAngle	Arc stop angle
Width	Arc width
Threshold	Arc threshold options
Center	Arc threshold center position relative to value
Color	Arc threshold colour
EndValue	Arc end value
Opacity	Arc threshold opacity value (0 to 255)
Span	Arc threshold value span
StartAngle	Arc threshold start angle
StartValue	Arc start value
SweepAngle	Arc threshold sweep angle
ThresholdKind	Arc starting and ending point reference type (Value or Angle)
CircleEndValue	Gauge circular form end angle
CircleStartValue	Gauge circular form start angle

DialText	Gauge dial text string
Digit	Gauge Digit
Alignment	Digit alignment
BackGroundColor	Digit background colour
BackGroundOpacity	Digit background opacity value (0 to 255)
Color	Digit colour
Visible	Enable or disable digit
DivisionColor	Major tick colour
DivisionCount	Total number of major ticks
DivisionWidth	Major tick width
EqualDimensions	Enable or disable equal width and height auto resize
Font	Dial text font options
Height	Widget height
Innercircle	Gauge face options
Color	Inner circle colour
Opacity	Inner circle opacity value (0 to 255)
Gloss	Inner circle glossy effect
Color	Inner circle gloss colour
Opacity	Inner circle gloss opacity value (0 to 255)
Left	Top-left X-position
Logarithmic	Enable or disable logarithmic scaling of labels
LogarithmicBase	Base value for logarithmic scaling
MaximumValue	Gauge maximum value
MinimumValue	Gauge minimum value
Needle	Needle options
Color	Needle color
InnerCenterColor	Inner center gradient colour 1
InnerCenterColorTo	Inner center gradient colour 2
InnerCenterOpacity	Inner center opacity value (0 to 255)
OuterCenterColor	Outer center gradient colour 1
OuterCenterColorTo	Outer center gradient colour 2
OuterCenterOpacity	Outer center opacity value (0 to 255)
ShineColor	Needle shine gradient colour 1
ShineColorTo	Needle shine gradient colour 2
OuterCircle	Outer circle options
Color	Outer circle colour
Opacity	Outer circle opacity value (0 to 255)
Width	Outer circle width
OuterRim	Outer rim options
Color	Outer rim colour
Opacity	Outer rim opacity value (0 to 255)
Width	Outer rim width
ShowValues	Enable or disable scale labels
SubDivisionColor	Minor tick colour
SubDivisionCount	Number of minor ticks between each major tick
SubDivisionWidth	Minor tick width
TextRendering	Text rendering option for labels
Top	Top-left Y-position

ValueFont	Value font options
ValueFormat	Value formatting option
Width	Widget width

Note: The Cool Gauge widget, usually, is composed with 18 different shades of one colour. The user can change that colour by right-clicking on the widget and selecting a colour from the Colour Picker. That colour will then have various shades applied to each of the 18 colours.

2.3.6. GCI Thermometer

The Thermometer widget is shown in the diagram below.

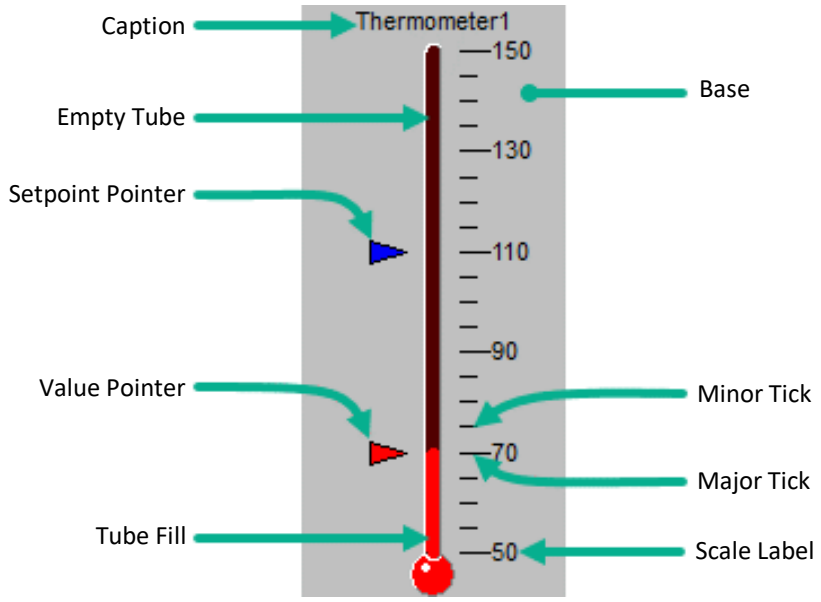


Figure 15: Thermometer Diagram

Property	Description
Name	Widget name
Alias	Widget alias
Caption	Caption text string
Color	Base colour
ColorAfter	Empty tube colour
ColorBefore	Tube fill colour
Direction	Widget orientation
Font	Label and caption font options
Height	Widget height
Left	Top-left X-position
Scale	Scale options
Color	Scale tick colour
Decimals	Decimal precision of scale labels
Heightmax	Major tick length
Heightmin	Minor tick length
Max	Maximum scale value
Min	Minimum scale value
Offset	Scale offset from tube
Position	Scale position relative to tube
ScaleType	Use logarithmic or linear scale values
SmallStep	Minor tick division
Step	Major tick steps
Visible	Enable or disable scale

Setpoint	Setpoint pointer options
Bitmap	Image for custom setpoint pointer
Color	Setpoint pointer colour
Shape	Setpoint pointer type
Value	Setpoint pointer value
Visible	Enable or disable setpoint pointer
ShowCaption	Enable or disable caption text
Top	Top-left Y-position
Value	Value pointer options
Bitmap	Image for custom value pointer
Color	Value pointer colour
Shape	Value pointer type
Value	Value pointer value
Visible	Enable or disable value pointer
Width	Widget width

2.3.7. GCI Tank

The Tank widget is shown in the diagram below.

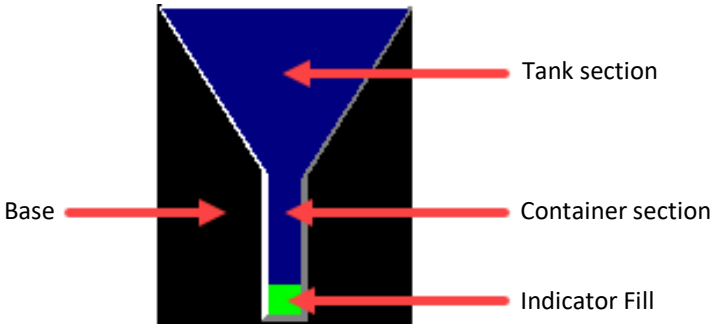


Figure 16: Tank Diagram

Property	Description
Name	Widget name
Alias	Widget alias
Color	Base colour
Container	Container section options
Height	Container section height
Visible	Enable or disable container section
Width	Container section width
EmptyColor	Empty section colour
FillColor	Indicator fill colour
Height	Widget height
Left	Top-left X-position
Max	Widget maximum value
Min	Widget minimum value
Position	Initial indicator fill value
Shape	Tank section shape
Top	Top-left Y-position
Width	Widget width

2.3.8. GCI Spectrum

The Spectrum widget is shown in the diagram below.

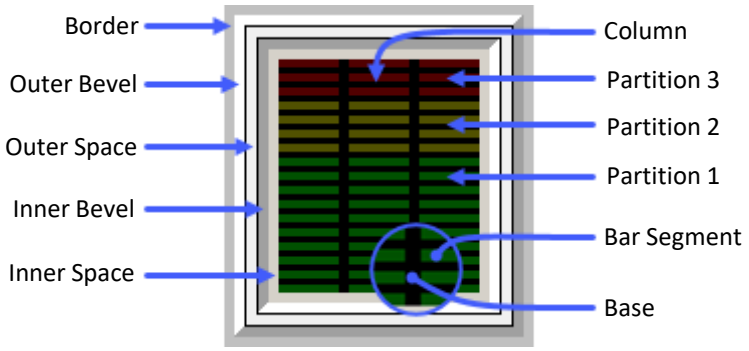


Figure 17: Spectrum Diagram

Property	Description
Name	Widget name
Alias	Widget alias
BarSpacing	Bar segment spacing
BarWidth	Bar segment width
Bevel	Bevel options
BorderColor	Border colour
BorderWidth	Border width
InnerColor	Inner Space colour
InnerHighlight	Inner Bevel highlight colour
InnerOutline	Inner Bevel outline colour
InnerShadow	Inner Bevel shadow colour
InnerSpace	Inner Space width
InnerStyle	Inner Bevel style
InnerWidth	Inner Bevel width
OuterColor	Outer Space colour
OuterHighlight	Outer Bevel highlight colour
OuterOutline	Outer Bevel outline colour
OuterShadow	Outer Bevel shadow colour
OuterSpace	Outer Space width
OuterStyle	Outer Bevel style
Outerwidth	Outer Bevel width
Visible	Enable or disable bevel
Color	Base colour
Columns	Total number of columns
Height	Widget height
Left	Top-left X-position
Maxvalue	Maximum value
Minvalue	Minimum value
Palette1	Partition 1 ticks
High	Active tick colour

Low	Inactive tick colour
Palette2	Partition 2 ticks
High	Active tick colour
Low	Inactive tick colour
Palette3	Partition 3 ticks
High	Active tick colour
Low	Inactive tick colour
Percent1	Percentage of partition 1
Percent2	Percentage of partition 2
Spacing	Column spacing
TickHeight	Bar segment thickness
Top	Top-left Y-position
Width	Widget width

2.3.9. GCI Scope

The Scope widget is shown in the diagram below.

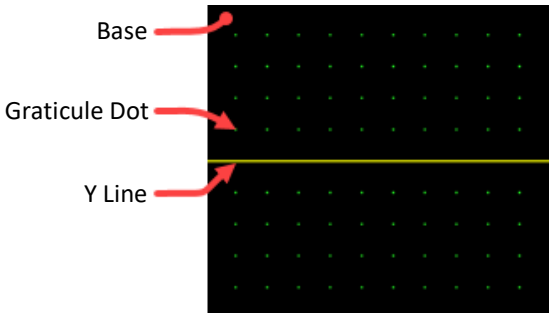


Figure 18: Scope diagram

Property	Description
Name	Widget name
Alias	Widget alias
Color	Base colour
GraticuleColor	Graticule dot colour
GraticuleVisible	Enable or disable graticule dots
GraticuleXinc	Graticule dot horizontal spacing
GraticuleYinc	Graticule dot vertical spacing
Height	Widget height
Left	Top-left X-position
RefreshIncrement	Update or redraw rate (ViSi-Genie only)
Top	Top-left Y-position
Trace1Color	Trace1 line colour
Trace2Color	Trace2 line colour
Trace3Color	Trace3 line colour
Trace4Color	Trace4 line colour
Traces	Number of traces
Width	Widget width
Xmag	Trace horizontal scaling
Yamp	Vertical trace scaling
YLine	X-axis offset distance from top
YLineColor	X-axis colour
YLineVisible	Enable or disable X-axis line
Yoffset	X-axis offset value (ViSi-Genie only)

2.3.10. GCI Smart Gauge

The Smart Gauge widget properties are listed below.

Figure 19: Smart Gauge

Property	Description
Name	Widget name
Alias	Widget alias
Config	Configure Smart Gauge (Opens the Smart Editor)
Left	Top-left X-position
Top	Top-left Y-position

Note: More details about creating and manipulating Smart Widgets can be found in the Smart Widgets Editor User Guide.

2.4. GCI Input Widgets

These are widgets commonly used for receiving touch inputs from the user.

The following widgets can be selected from the Inputs Pane by selecting their respective icon.

Knob

Dipswitch

Rotary Switch

Rocker

Slider

Color Picker

Trackbar

Smart Knob

Keyboard

Smart Slider

Note: Smart Widgets are only enabled when a Workshop4 Pro License is purchased.

2.4.2. GCI Knob

The Knob widget is shown in the diagram below.

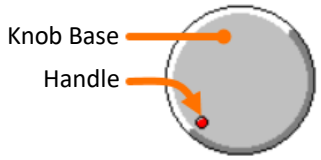


Figure 20: Knob Diagram

Property	Description
Name	Widget name
Alias	Widget alias
BackImage	Knob base image
BaseAngle	Rotation endpoint offset angle (Counterclockwise rotation from the south)
HandleImage	Handle image
Left	Top-left X-position
Maxvalue	Knob maximum value
Minvalue	Knob minimum value
Radius	Knob radius
Top	Top-left Y-position
Width	Widget width

2.4.3. GCI Rotary Switch

The Rotary Switch widget is shown in the diagram below.

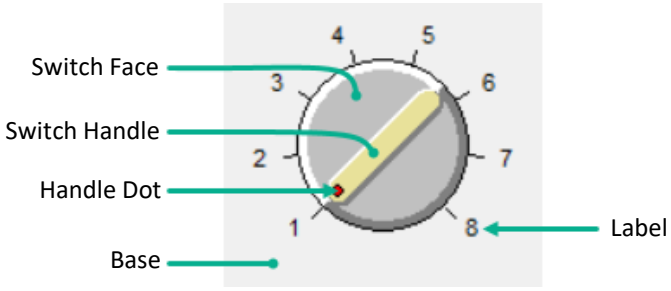


Figure 21: Rotary Switch Diagram

Property	Description
Name	Widget name
Alias	Widget alias
ButtonColor	Switch handle dot colour
Color	Base colour
Font	Font options
Height	Widget height
LabelsOffset	Label offset from the switch
Left	Top-left X-position
Radius	Knob radius
ShowLabel	Enable or disable labels
SwitchAngleEnd	Switch rotation ending angle
SwitchAngleStart	Switch rotation starting angle
SwitchColor	Switch face colour
SwitchPositions	String list entry for each switch position
Top	Top-left Y-position
Width	Widget width
WinchColor	Switch handle colour
WinchOffset	Switch handle thickness

2.4.4. GCI Slider

The Slider widget is shown in the diagram below.

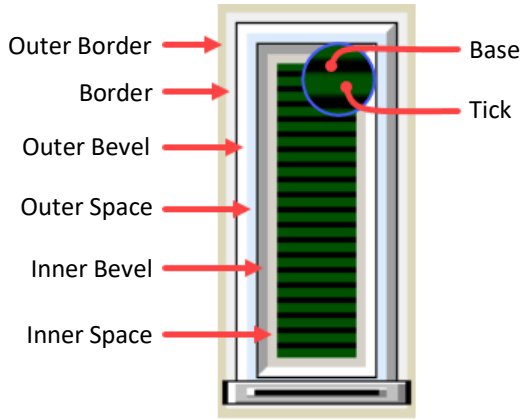


Figure 22: Slider Diagram

Property	Description
Name	Widget name
Alias	Widget alias
Bevel	Bevel options
BorderColor	Border colour
BorderWidth	Border width
InnerColor	Inner Space colour
InnerHighlight	Inner Bevel highlight colour
InnerOutline	Inner Bevel outline colour
InnerShadow	Inner Bevel shadow colour
InnerSpace	Inner Space width
InnerStyle	Inner Bevel style
InnerWidth	Inner Bevel width
OuterColor	Outer Space colour
OuterHighlight	Outer Bevel highlight colour
OuterOutline	Outer Bevel outline colour
OuterShadow	Outer Bevel shadow colour
OuterSpace	Outer Space width
OuterStyle	Outer Bevel style
Outerwidth	Outer Bevel width
Visible	Enable or disable bevel
BorderColor	Outer Border colour
BorderWidth	Outer Border width
Color	Base colour
Height	Widget height
Left	Top-left X-position
Maxvalue	Slider maximum value
Minvalue	Slider minimum value
Orientation	Slider orientation
Palette1	Tick colour
High	Active tick colour

Low	Inactive tick colour
Spacing	Tick spacing
SolidFill	Select tick fill type
TickWidth	Tick width
Top	Top-left Y-position
Width	Widget width

2.4.5. GCI Trackbar

The Trackbar widget is shown in the diagram below.

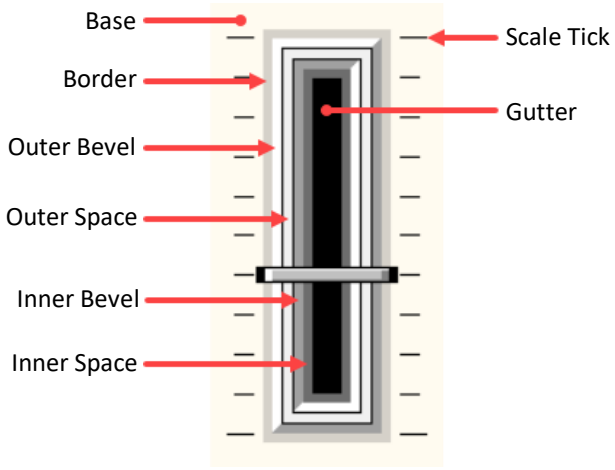


Figure 23: Trackbar Diagram

Property	Description
Name	Widget name
Alias	Widget alias
BorderWidth	Spacing between track and base edge
Color	Base colour
Frequency	Frequency of scale tick partitions based on maximum value
GutterBevel	Gutter Bevel options
BorderColor	Border colour
BorderWidth	Border width
InnerColor	Inner Space colour
InnerHighlight	Inner Bevel highlight colour
InnerOutline	Inner Bevel outline colour
InnerShadow	Inner Bevel shadow colour
InnerSpace	Inner Space width
InnerStyle	Inner Bevel style
InnerWidth	Inner Bevel width
OuterColor	Outer Space colour
OuterHighlight	Outer Bevel highlight colour
OuterOutline	Outer Bevel outline colour
OuterShadow	Outer Bevel shadow colour
OuterSpace	Outer Space width
OuterStyle	Outer Bevel style
Outerwidth	Outer Bevel width
Visible	Enable or disable bevel
GutterColor	Gutter colour
GutterWidth	Gutter width
Height	Widget height
Maxvalue	Slider maximum value
Minvalue	Slider minimum value

Orientation	Slider orientation
ScaleOffset	Scale tick offset distance from track
TickColor	Tick colour
TickMarks	Tick position
Top	Top-left Y-position
Width	Widget width

2.4.6. GCI Keyboard

The Keyboard widget is shown in the diagram below.

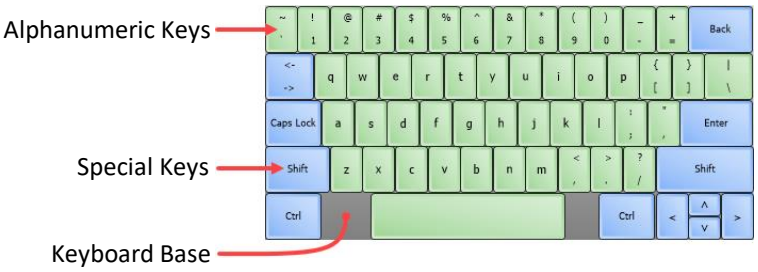


Figure 24: Qwerty Keyboard Diagram

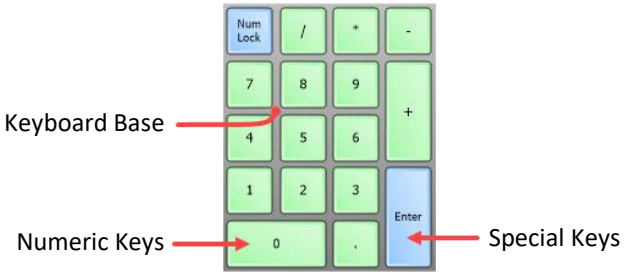


Figure 25: Numeric Keyboard Diagram

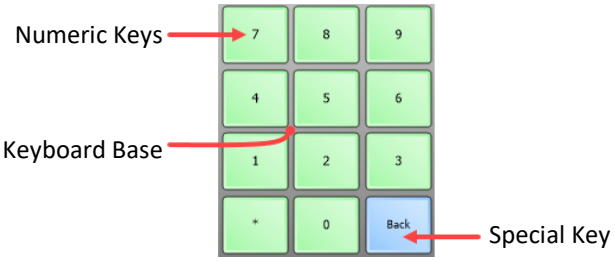


Figure 26: Cellphone Keyboard Diagram

Property	Description
Name	Widget name
Alias	Widget alias
AutoCapsDisplay	Enables or disables auto capitalization of letters for the Caps Lock key
Fill	Keyboard base fill options (Opens Fill Editor)
Font	Font options for the main keys
Height	Widget height
HighlightCaps	Highlight colour for keys toggled by the caps lock or shift keys
KeyboardType	Key layout type (Opens Keyboard Editor)
KeyDistance	Distance between each key
Left	Top-left X-position
SmallFont	Font options for the special keys
Top	Top-left Y-position
Width	Widget width

2.4.7. GCI Dipswitch

The Dipswitch widget is shown in the diagram below.

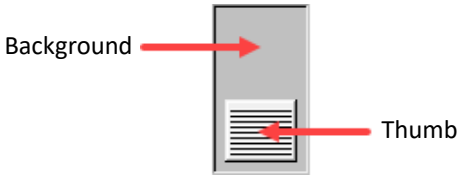


Figure 27: Dipswitch Diagram

Property	Description
Name	Widget name
Alias	Widget alias
Background	Background options
BorderStyle	Bevel style
BrightHighlightColor	Bevel highlight colour 1
DarkShadowColor	Bevel shadow colour 1
FaceColor	Background track colour
HighlightColor	Bevel highlight colour 2
ShadowColor	Bevel shadow colour 2
Height	Widget height
Left	Top-left X-position
Margin	Thumb distance from background edges
NumPositions	Number of switch positions
Orientation	Slider orientation
Thumb	Thumb options
BorderStyle	Bevel style
BrightHighlightColor	Bevel highlight colour 1
DarkShadowColor	Bevel shadow colour 1
FaceColor	Thumb face colour
HighlightColor	Bevel highlight colour 2
Length	Thumb length
Ripples	Enable or disable ripple line design
ShadowColor	Bevel shadow colour 2
Top	Top-left Y-position
Width	Widget width

2.4.8. GCI Rocker

The Rocker widget is shown in the diagram below.

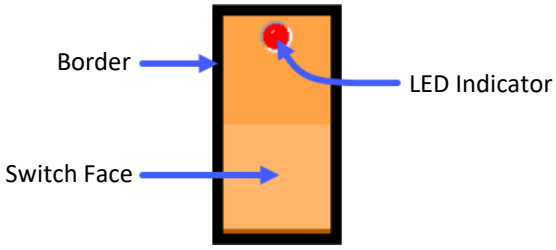


Figure 28: Rocker Diagram

Property	Description
Name	Widget name
Alias	Widget alias
BorderColor	Border colour
BorderWidth	Border width
ClickRect	Set touch detection area (Whole or Half)
Height	Widget height
LED	LED indicator options
UseDefaultColours	Enable or disable default LED colour
AutoInactiveColor	Enable auto selection of inactive LED colour
Centered	Centre LED position relative to rocker orientation axis
ColorActive	Active state LED colour
ColorDarkShadow	LED bevel dark shadow colour (Not applicable for Ellipse shape)
ColorHighlight	LED bevel highlight colour
ColorInactive	Inactive state LED colour
ColorShadow	LED bevel shadow colour
Height	LED height
Left	LED horizontal offset distance
Shape	LED shape selection (Diamond, Ellipse or Rectangle)
ShowReflection	Enable or disable LED reflection effect
Top	LED vertical offset distance
Visible	Enable or disable LED indicator
Width	LED width
Left	Top-left X-position
Orientation	Slider orientation
SwitchColors	Switch face colour options
UseDefaultColors	Use monochromatic colour
COffAngled	Face colour for the Off half (Rocker toggled to On state)
COffDown	Face colour for the Off half (Rocker toggled to Off state)
COffEdge	Edge colour for the Off half (Rocker toggled to On state)
COnAngled	Face colour for the On half (Rocker toggled to Off state)
COnDown	Face colour for the On half (Rocker toggled to On state)
COnEdge	Edge colour for the On half (Rocker toggled to Off state)
Top	Top-left Y-position
Width	Widget width

2.4.9. GCI Color Picker

The Color Picker widget is shown in the diagram below.

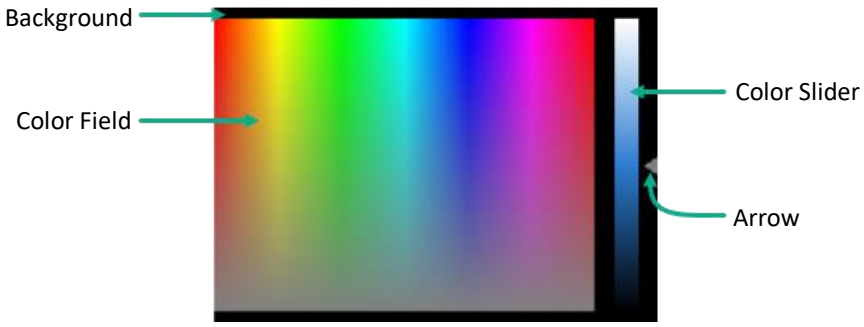


Figure 29: Color Picker diagram

Property	Description
Name	Widget name
Alias	Widget alias
Color	Background colour
Height	Widget height
Left	Top-left X-position
Top	Top-left Y-position
Width	Widget width

2.4.10. GCI Smart Knob

The Smart Knob widget properties are listed below.

Figure 30: Smart Knob

Property	Description
Name	Widget name
Alias	Widget alias
Config	Configure Smart Knob (Opens the Smart Editor)
Left	Top-left X-position
Top	Top-left Y-position

Note: More details about creating and manipulating Smart Widgets can be found in the Smart Widgets Editor User Guide.

2.4.11. GCI Smart Slider

The Smart Slider widget properties are listed below.

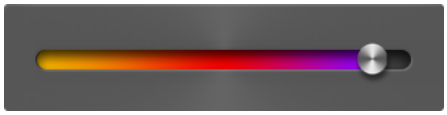


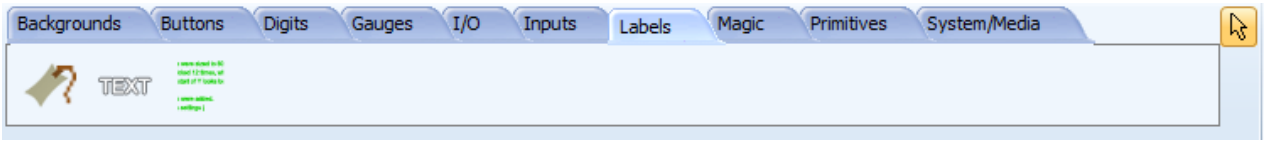
Figure 31: Smart Slider

Property	Description
Name	Widget name
Alias	Widget alias
Config	Configure Smart Slider (Opens the Smart Editor)
Left	Top-left X-position
Top	Top-left Y-position

Note: More details about creating and manipulating Smart Widgets can be found in the Smart Widgets Editor User Guide.

2.5. GCI Label Widgets

These are widgets commonly used for displaying text information.



The following widgets can be selected from the Labels Pane by selecting their respective icon.

Label

Strings

Static Text

2.5.1. GCI Label

The Label widget is shown in the diagram below.

Figure 32: Label Diagram

Property	Description
Name	Widget name
Alias	Widget alias
4DGLFont	Text alignment inside the textbox
Font	System font selection
Bold	Enable or disable bold font style
Italic	Enable or disable italicized font style
Inverse	Swap font and background colour
Underline	Enable or disable underline
MagWidth	Font width multiplier
MagHeight	Font height multiplier
Opaque	Enable or disable background
BGcolor	Label background colour
Caption	Label text string
Font	Text font style
FGcolor	Text colour
FontSize	Text font size
Left	Top-left X-position
Top	Top-left Y-position

2.5.2. GCI Static Text

The Static Text widget is shown in the diagram below.

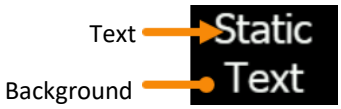


Figure 33: Static text diagram

The text inside the static text can be aligned as shown below.

Property	Description
Name	Widget name
Alias	Widget alias
Alignment	Text alignment inside the textbox
Autosize	Enable or disable textbox auto sizing relative to contents
Caption	Text string
Color	Background colour
Font	Text font options
Height	Widget height
Left	Top-left X-position
Top	Top-left Y-position
Transparent	Enable or disable textbox background
Width	Widget width
Wordwrap	Enable or disable word wrapping

2.5.3. GCI Strings

The Strings widget properties are listed below.

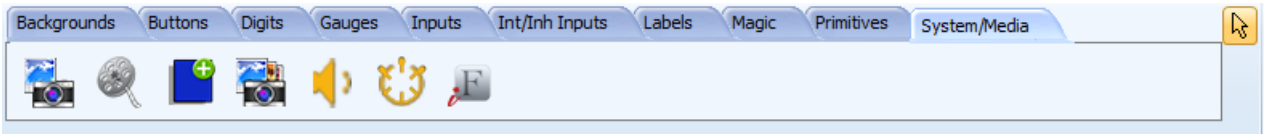
Figure 34: String diagram

The text inside the static text can be aligned as shown below.

Property	Description
Name	Widget name
Alias	Widget alias
Alignment	Text alignment inside the textbox
BGcolor	Text background colour
Height	Widget height
FGcolor	Text font colour
Font	Text font options
Left	Top-left X-position
Strings	String input (Open string editor)
StringStyle	String content style
Top	Top-left Y-position
Width	Widget width

2.6. GCI System/Media

These are widgets commonly used for multimedia and system.



The following widgets can be selected from the System/Media Pane by selecting their respective icon.

Image

Form

Video

User Image

2.6.1. GCI Image

The Image widget properties are listed below.

Figure 35: Image widget

Property	Description
Name	Widget name
Alias	Widget alias
Height	Widget height
Image	Image file selection (Opens Image + Video Converter)
Left	Top-left X-position
Source	Source file crop options
Height	Frame crop height
Left	Frame crop horizontal starting point
Top	Frame crop vertical starting point
Width	Frame crop width
Top	Top-left Y-position
Width	Widget width

Note: When this widget is added, it will initially prompt the user to select an image file.

2.6.2. GCI Video

The Video widget properties are listed below.

Figure 36: Video widget

Property	Description
Name	Widget name
Alias	Widget alias
Frames	Video frames options
First	Select initial video frame
Last	Select final video frame
FrameDelay	Video frame delay (ViSi-Genie only)
Height	Widget height
Left	Top-left X-position
Source	Source file clipping window options
Height	Frame clip window height
Left	Frame clip window horizontal starting point
Top	Frame clip window vertical starting point
Width	Frame clip window width
Top	Top-left Y-position
Video	Video file selection (Opens Image + Video Converter)
Width	Widget width

Note: When this widget is added, it will initially prompt the user to select a media file. The user can then replace or edit the video by pressing the ··· button of the **Video** property.

2.6.3. GCI Form

The Form properties are listed below.

Property	Description
Name	Widget name
Alias	Widget alias
Bgtype	Form background type (Colour or Image)
Color	Form background colour
Image	Form background image file selection (Opens Image + Video Converter)
Height	Widget height
Left	Top-left X-position
Source	Source file crop options
Height	Frame crop height
Left	Frame crop horizontal starting point
Top	Frame crop vertical starting point
Width	Frame crop width
Top	Top-left Y-position
Width	Widget width

2.6.4. GCI User Image

The User Image widget properties are listed below.

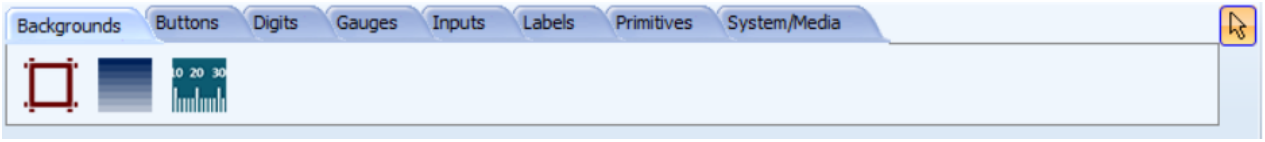
Figure 37: User Image widget

Property	Description
Name	Widget name
Alias	Widget alias
Height	Widget height
Images	Image file selection (Opens Image List Editor)
Left	Top-left X-position
Stretch	Enable or disable image resizing
Top	Top-left Y-position
Width	Widget width

Note: When this widget is added, it will be initially empty. The user can add the images by pressing the ··· button or double-clicking the **Images** property.

2.7. GCI Background Widgets

These are widgets commonly used for adding visual cues or design.



The following widgets can be selected from the Background Pane by selecting their respective icon.

Border

Scale

Gradient

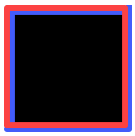
2.7.1. GCI Border

The Border widget is shown below.

Figure 38: Border widget

The border can have the following styles:

Box



Frame

TopLine

LeftLine

BottomLine

RightLine

Spacer

Property	Description
Name	Widget name
Alias	Widget alias
Height	Widget height
HighlightColor	Border bevel highlight colour
Left	Top-left X-position
ShadowColor	Border bevel shadow colour
Shape	Border shape
Style	Border bevel direction (Lowered or Raised)
Top	Top-left Y-position
Width	Widget width

2.7.2. GCI Gradient

The Gradient widget is shown below.

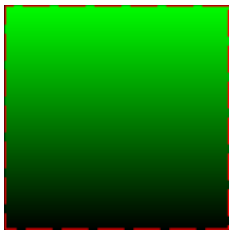


Figure 39: Gradient widget

Property	Description
Name	Widget name
Alias	Widget alias
ColorWidth	Gradient colour step size
Direction	Gradient direction
EndColor	Gradient end colour
Height	Widget height
Left	Top-left X-position
StartColor	Gradient start colour
SwapColors	Swap start and end colours
Top	Top-left Y-position
Width	Widget width

2.7.3. GCI Scale

The Scale widget is shown in the diagram below.

Figure 40: Scale Diagram

Property	Description
Name	Widget name
Alias	Widget alias
Alignment	Label alignment relative to scale ticks
Color	Background colour
Digits	Label format
Font	Scale font colour
Height	Widget height
Layout	Label alignment relative to orientation axis
LeadingZero	Enable or disable zero
Left	Top-left X-position
Length	Widget length
Maxvalue	Maximum value of the numerical mark
Minvalue	Minimum value of the numerical mark
Orientation	Scale orientation (Horizontal or Vertical)
PeakColor	Peak value colour
PeakLevel	Label peak value
ScaleColor	Scale tick colour
ScaleOffset	Scale tick offset
ShowSign	Enable or disable label sign
TickMarks	Tick mark layout
Ticks	Total number of partitions
TickHeight	Tick length
TickWidth	Tick width
Top	Top-left Y-position
Transparent	Enable or disable widget transparency
Width	Widget width

3. Internal (PmmC) Widgets

The **Internal Functions (PmmC) Widgets** provides a way to deliver high quality graphical widgets without using any external memory storage device. This type of widget is generated during runtime by executing widget functions included in the PmmC. This widget type is ideal for standalone applications that requires light graphics.

These widgets are available for Pixxi and Diablo16 processors only. **These are not available for Goldelox or Picaso.**

Internal Widgets requires a small amount of program space to store widget parameters. This reduces that amount of useable code space for the project. If the project will require a huge amount of code space and multiple widgets, it is highly advisable to use GCI widgets instead.

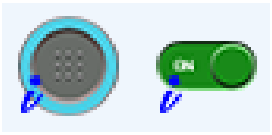
In Workshop4, this type of widget can be identified by checking for a blue / located at the bottom left of the widget icon.

Here is a sample application screen utilizing Internal widgets:

3.1. Internal Button Widgets

These are widgets commonly used for receiving touch inputs from the user.

All buttons can be found under the **Buttons** tab of the widget selection pane as shown above.



Internal (PmmC) buttons can be identified with the blue *i* located at the bottom left of the button icon.

3.1.1. Internal Button D

The function required for rendering this widget is discussed in the **gfx_Button4** section of Pixxi or Diablo Internal Functions Reference Manual. The Button widget is shown in the diagram below.

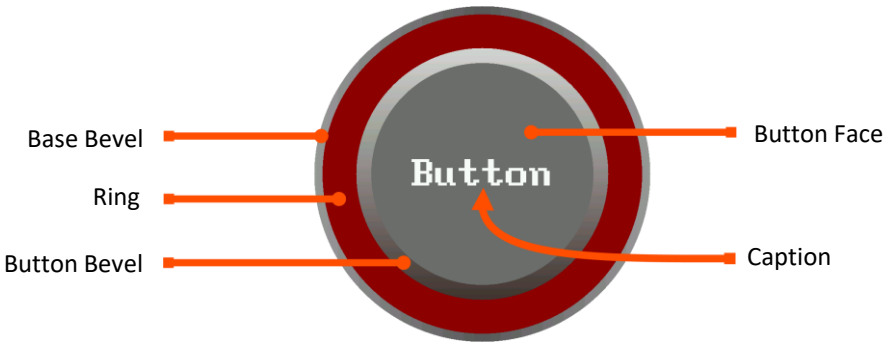


Figure 41: Button with Caption

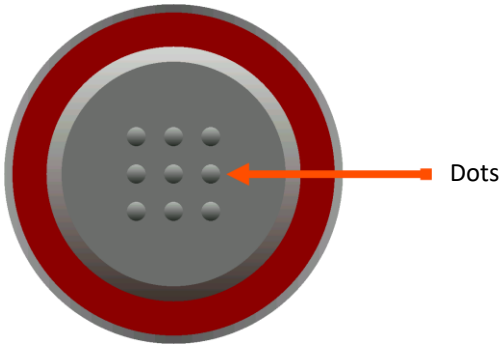


Figure 42: Button with Dots

Property	Description
Name	Widget name
Alias	Widget alias
BaseColorA	Base bevel gradient start colour
BaseColorB	Base bevel gradient end colour
BaseGradient	Base bevel gradient direction/style
ButtonColorA	Button bevel gradient start colour
ButtonColorB	Button bevel gradient end colour
ButtonGradientOff	Off position button bevel gradient direction/style
ButtonGradientOn	On position button bevel gradient direction/style
Caption	Caption text (Applicable if using text caption on button face)
CaptionColorOff	Off position text label colour (Applicable if using Caption)
CaptionColorOn	On position text label colour (Applicable if using Caption)
CaptionStyle	Setting for caption or dots on button face (Set to Yes for Caption, No for Dots)
DotColorA	Dot gradient start colour (Applicable if using Dots on button face)
DotColorB	Dot gradient end colour (Applicable if using Dots on button face)
DotGradient	Dot gradient direction/style (Applicable if using Dots on button face)
FaceColor	Button face colour

Property	Description
Font	Caption text font (Applicable if using Caption), refer to the section **External Font** when using custom fonts
FontSize	Caption text font size (Applicable if using Caption)
Matrix	Button grouping (ViSi-Genie only)
Momentary	Button type
Left	Top-left X-position
Radius	Button size
RingColorOff	Off position ring colour
RingColorOn	On position ring colour
Top	Top-left Y-position

3.1.2. Internal Switch

The function required for rendering this widget is discussed in the **gfx_Switch** section of Pixxi or Diablo Internal Functions Reference Manual. The Switch widget is shown in the diagram below.

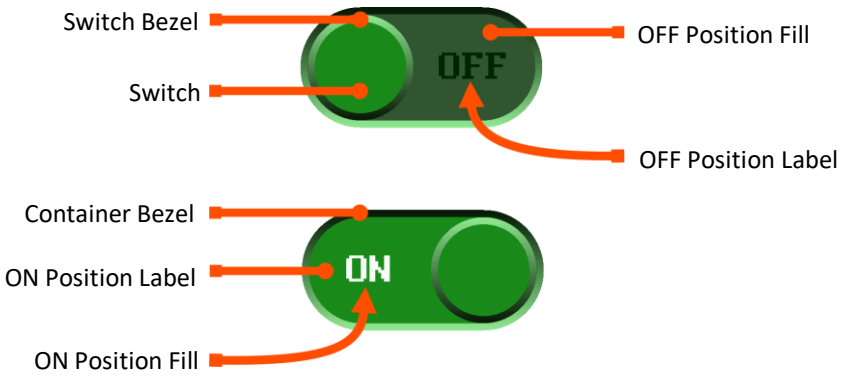
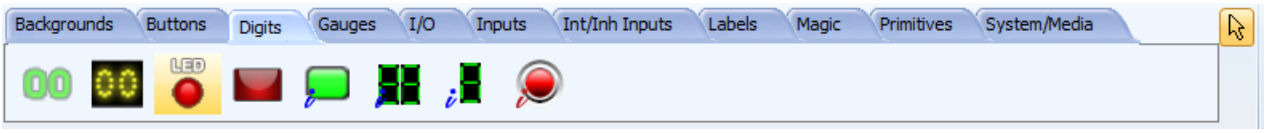


Figure 43: Switch Diagram

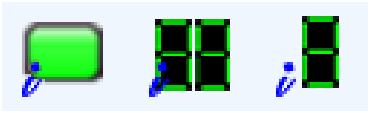
Property	Description
Name	Widget name
Alias	Widget alias
BevelMainColor	Container and switch bevel main colour
BevelShadowColor	Container and switch bevel shadow colour
ContainerBevel	Container bevel thickness
FontSize	Label font size
Font	Label font, refer to the section **External Font** when using custom fonts
Height	Widget height
Left	Top-left X-position
OFFColor	OFF position fill colour
OFFLabel	OFF position text label
OFFLabelColor	OFF position text label colour
ONColor	ON position fill colour
ONLabel	ON position text label
ONLabelColor	ON position text label colour
Orientation	Switch orientation
SwitchBevel	Switch bezel thickness
Top	Top-left Y-position
Width	Widget width

3.2. Internal Digits Widget

These are widgets commonly used for displaying values or status information.



All Digits widgets can be found under the **Digits** tab of the widget selection pane as shown above. This pane is mostly composed of Leds and Digits.



Internal (PmmC) Led and Digits can be identified with the blue *i* located at the bottom left of the widget icon.

3.2.1. Internal LED

The function required for rendering this widget is discussed in the **gfx_Led** section of Pixxi or Diablo Internal Functions Reference Manual. The LED widget is shown in the diagram below.

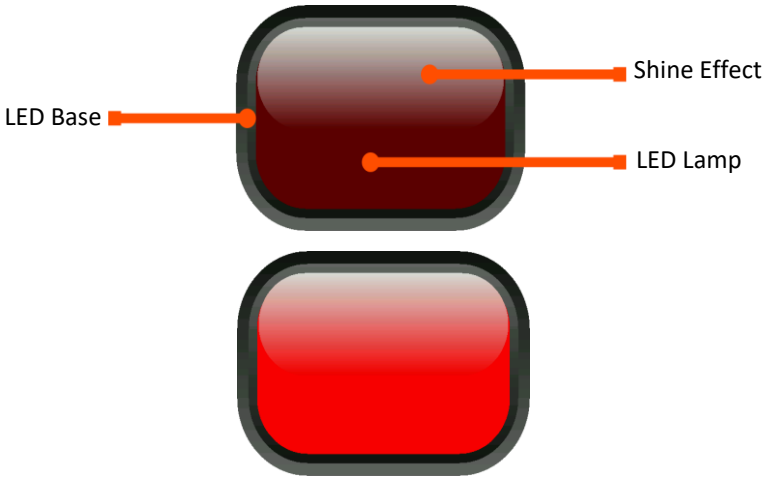


Figure 44: LED Diagram

Property	Description
Name	Widget name
Alias	Widget alias
BaseColorA	Base bevel gradient start colour
BaseColorB	Base bevel gradient end colour
BaseInnerCorner	Inner corner radius (Ideally less than BaseOuterCorner)
BaseOuterCorner	Outer corner (Ideally less than half of the widget height)
Height	Widget height
LEDcolorOff	LED Off colour
LEDcolorOn	LED On colour
LEDLampRadius	LED lamp radius
Left	Top-left X-position
ShineColor	LED shine effect colour
ShineEffect	Enable or disable LED shine effect
ShineRadius	LED shine effect radius (Ideally less than or equal to BaseInnerCorner)
Top	Top-left Y-position
Width	Widget width

3.2.2. Internal LED Digits

The function required for rendering this widget is discussed in the **gfx_LedDigits** section of Pixxi or Diablo Internal Functions Reference Manual. The LED Digits widget is shown in the diagram below.

Figure 45: LED Digits Diagram

Property	Description
Name	Widget name
Alias	Widget alias
ColorActive	Active LED segment colour
ColorInactive	Inactive LED segment colour
DecimalPlace	Decimal Place relative to separator
Digits	Total number of LED digit
DPisComma	Separator style for decimal point (Set to Yes for Comma, No for Dot)
Format	Input format
LeadingZero	Enable or disable leading zeroes
Left	Top-left X-position
ShowDP	Enable or disable separator
Signed	Enable or disable signed value output
Size	Digit size
Spacing	Spacing between each digit
Top	Top-left Y-position

3.2.3. Internal LED Digit

The function required for rendering this widget is discussed in the **gfx_LedDigit** section of Pixxi or Diablo Internal Functions Reference Manual. The LED Digit widget is shown in the diagram below.

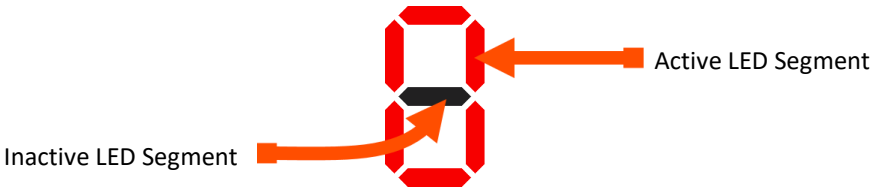
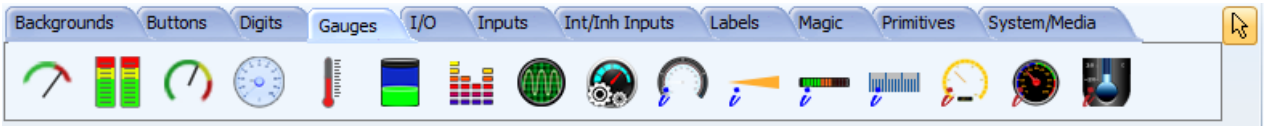


Figure 46: LED Digit Diagram

Property	Description
Name	Widget name
Alias	Widget alias
ColorActive	Active LED segment colour
ColorInactive	Inactive LED segment colour
Left	Top-left X-position
Size	Digit size
Top	Top-left Y-position

3.3. Internal Gauge Widgets

These are widgets commonly used for displaying values.



All gauges can be found under the **Gauges** tab of the widget selection pane as shown above.

Internal (PmmC) gauges can be identified with the blue *i* located at the bottom left of the gauge icon.

3.3.1. Internal Angular Meter

The function required for rendering this widget is discussed in the **gfx_AngularMeter** section of Pixxi or Diablo Internal Functions Reference Manual. The Angular gauge is shown in the diagram below.

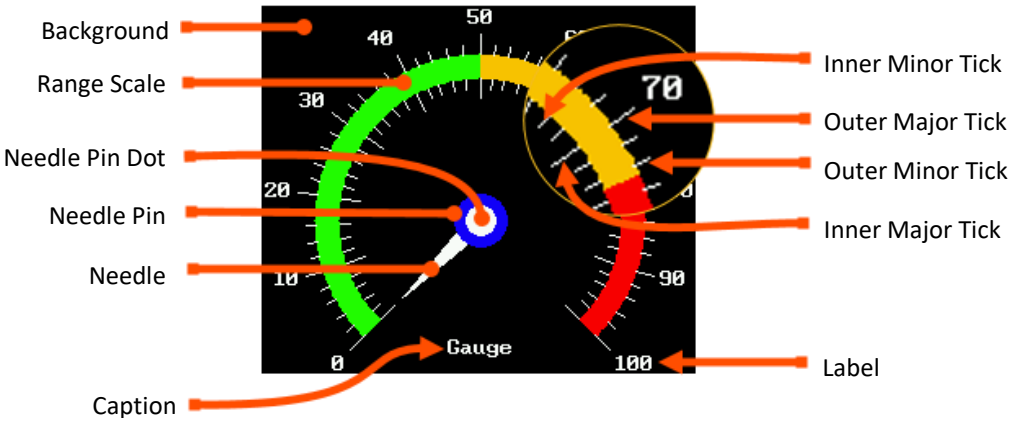


Figure 47: Angular Gauge Diagram

Property	Description
Name	Widget name
Alias	Widget alias
BGcolor	Gauge background colour (Required for erasing Needle path)
BGtransparent	Enable/Disable background
Caption	Gauge caption text
CaptionColor	Gauge caption colour
CaptionFont	Gauge caption font family, refer to the section **External Font** when using custom fonts
CaptionX0ffset	Caption horizontal offset from centre of rotation
CaptionYOffset	Caption vertical offset from centre of rotation
GaugeEndAngle	Set the ending angle for the scale and needle element
GaugeStartAngle	Starting angle for the scale and needle element
Height	Widget height
LabelColor	Label text colour
LabelColorFromScale	Use scale colour at current point as label text colour (Overrides Label text colour)
LabelCount	Total number of labels (Actual labels = LabelCount + 1; set to -1 to disable labels)
LabelDiscrete	Label type (Set to yes for individual text labels)
LabelFont	Label text font, refer to the section **External Font** when using custom fonts
LabelOffset	Label offset distance from centre point
Labels	String input for individual text labels
Left	Top-left X-position
Maxvalue	Gauge maximum value
MinorTicks	Number of minor ticks between each major ticks (Set to 0 to disable)
Minvalue	Gauge minimum value
NeedleColor	Gauge Needle colour
NeedleLength	Needle pointer length
NeedleOffset	Needle offset from centre of rotation

Property	Description
NeedlePinColor	Needle pin colour
NeedlePinDotColor	Needle pin dot colour
NeedlePinDotRadius	Needle pin dot radius
NeedlePinRadius	Needle pin radius
NeedleStyle	Needle pointer style
NeedleTail	Needle tail length (Applicable only for double triangle style)
NeedleThickness	Needle width (Half value of overall needle thickness)
Sect1Color	Colour of the first ring section of the range scale
Sect2Color	Colour of the second ring section of the range scale
Sect2StartingAngle	Starting angle of the second ring section of the range scale
Sect3Color	Colour of the third ring section of the range scale
Sect3StartingAngle	Starting angle of the third ring section of the range scale
SectionStepSize	Range scale section incremental step size
ScaleInnerRadius	Range scale inner radius
ScaleOuterRadius	Range scale outer radius
TickColor	Major and Minor tick colour
TickColorFromScaleColor	Use scale colour at current point as tick colour (Overrides tick colour)
TickInnerMajorlength	Length of the inner major ticks
TickInnerMinorLength	Length of the inner minor ticks
TickOuterMajorLength	Length of the outer major ticks
TickOuterMinorLength	Length of the outer minor ticks
TickWidth	Tick line thickness
Top	Top-left Y-position
TotalTicks	Total number of partitions between each ticks
Width	Widget width

Note: When designing the gauge, note that the needle will use the background colour parameter for erasing its path, whether the background is set as transparent or opaque.

3.3.2. Internal Needle

The function required for rendering this widget is discussed in the **gfx_Needle** section of Pixxi or Diablo Internal Functions Reference Manual. The Needle widget is shown in the diagram below.

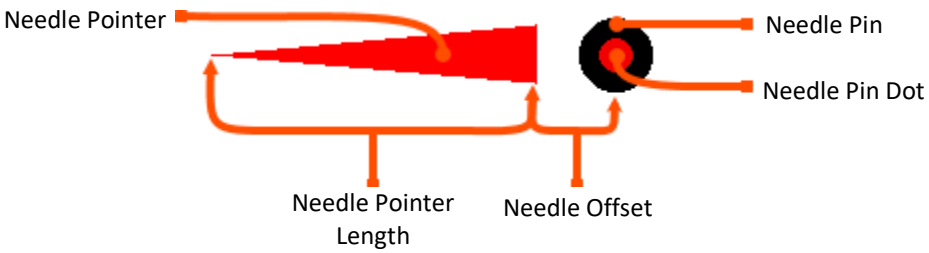
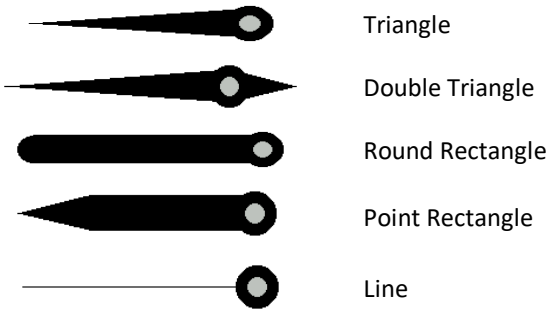


Figure 48: Needle Diagram

There are five different styles available for the needle as shown below.



Property	Description
Name	Widget name
Alias	Widget alias
BGcolor	Background colour (Required for erasing Needle path)
EndAngle	Needle rotation end angle
Height	Widget height
Left	Top-left X-position
Maxvalue	Needle maximum value
Minvalue	Needle minimum value
NeedleColor	Needle pointer colour
NeedleLength	Needle pointer length
NeedleOffset	Needle offset from centre of rotation
NeedlePinColor	Needle pin colour
NeedlePinDotColor	Needle pin dot colour
NeedlePinDotRadius	Needle pin dot radius
NeedlePinRadius	Needle pin radius
Needlestyle	Needle pointer style
NeedleTail	Needle tail length (Applicable only for double triangle style)
NeedleThickess	Needle width (Half value of overall needle thickness)
StartAngle	Needle rotation starting angle
Top	Top-left Y-position
Width	Widget width

3.3.3. Internal Gauge

The function required for rendering this widget is discussed in the **gfx_Gauge** section of Pixxi or Diablo Internal Functions Reference Manual. The Gauge widget is shown in the diagram below.

Figure 49: Gauge Diagram

Property	Description
Name	Widget name
Alias	Widget alias
BaseColor	Gauge base colour
Height	Widget height
Left	Top-left X-position
Maxvalue	Gauge maximum value
Minvalue	Gauge minimum value
Orientation	Gauge orientation
P1ActiveColor	Partition 1 bar active colour
P1InactiveColor	Partition 1 bar inactive colour
P2ActiveColor	Partition 2 bar active colour
P2InactiveColor	Partition 2 bar inactive colour
P2start	Partition 2 percentage
P3ActiveColor	Partition 3 bar active colour
P3InactiveColor	Partition 3 bar inactive colour
P3start	Partition 3 percentage
Spacing	Bar spacing
Thickness	Gauge thickness
Top	Top-left Y-position
TopRight	Change gauge fill direction
Width	Widget width

3.3.4. Internal Ruler Gauge

The function required for rendering this widget is discussed in the **gfx_RulerGauge** section of Pixxi or Diablo Internal Functions Reference Manual. The Internal Ruler Gauge widget is shown in the diagram below.

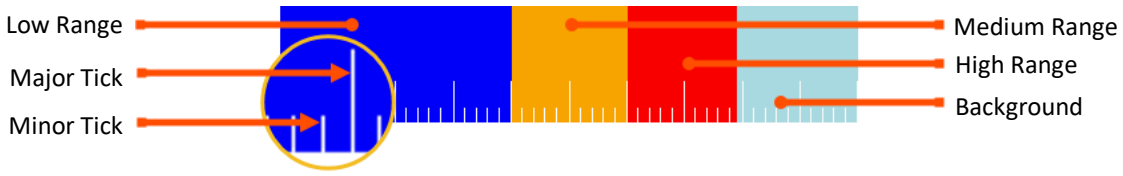
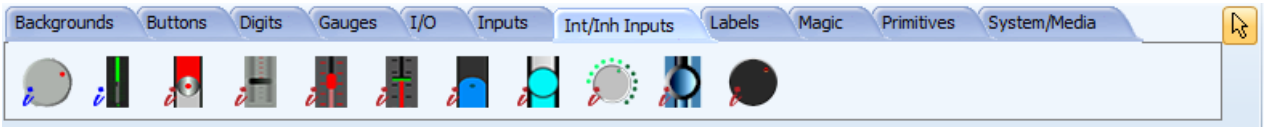


Figure 50: Ruler Gauge Diagram

Property	Description
Name	Widget name
Alias	Widget alias
BackGroundColor	Gauge background colour
Height	Widget height
Left	Top-left X-position
MajorTickLength	Major tick length
MajorTicks	Total number of major ticks
Maxvalue	Gauge maximum value
MinorTickLength	Minor tick length
MinorTicks	Total number of minor ticks
Orientation	Gauge orientation
P1color	Low range colour
P2color	Medium range colour
P2start	Starting value of the medium range
P3color	High range colour
P3start	Starting value of the high range
TickColor	Tick colour
TicksBR	Set tick position (Set to Yes for bottom/right side, otherwise top/left side)
Top	Top-left Y-position
Width	Widget width

3.4. Internal Input Widgets

These are widgets commonly used for receiving touch inputs from the user.



All Internal input widgets can be found under the **Int/Inh Inputs** tab of the widget selection pane as shown above.

Internal (PmmC) sliders and knobs can be identified with the blue *i* located at the bottom left of the widget icon.

3.4.1. Internal Knob

The function required for rendering this widget is discussed in the **gfx_Dial** section of Pixxi or Diablo Internal Functions Reference Manual. The Knob widget is shown in the diagram below.

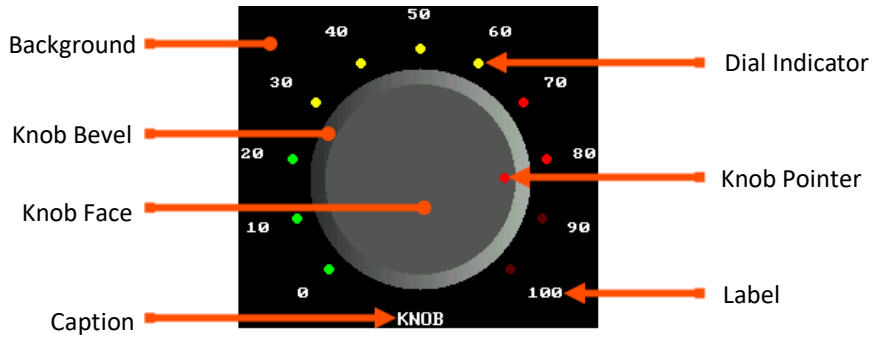


Figure 51: Knob Diagram

Property	Description
Name	Widget name
Alias	Widget alias
BackGroundColor	Knob background colour
BevelColorA	Knob bevel background left side colour
BevelColorB	Knob bevel background right side colour
BevelSize	Knob bevel thickness
BGtransparent	Enable or disable background
Caption	Knob caption text
CaptionColor	Knob caption text colour
CaptionFont	Knob caption text font, refer to the section **External Font** when using custom fonts
CaptionX0ffset	Knob caption horizontal offset from centre of rotation
CaptionYOffset	Knob caption vertical offset from centre of rotation
DialIndicatorStyle	Knob dial indicator style
DialLength	Knob dial length (Not applicable for Circle indicator style)
DialOffset	Dial indicator offset relative to knob
DialSize	Knob dial size (Not applicable for Line indicator style)
Height	Widget height
KnobColor	Knob face colour
KnobRadius	Knob radius
LabelColor	Label text colour
LabelCount	Total number of labels (Actual labels = LabelCount + 1; set to -1 to disable labels)
LabelDiscrete	Label type (Set to yes for individual text labels)
LabelFont	Label text font, refer to the section **External Font** when using custom fonts
LabelOffset	Label text offset relative to knob
Labels	String input for individual text labels
Left	Top-left X-position
MaxAngle	Knob maximum angle
Maxvalue	Knob maximum value
MinAngle	Knob minimum angle
Minvalue	Knob minimum value

Property	Description
P1ActiveColor	Partition 1 indicator active colour
P1InactiveColor	Partition 1 indicator inactive colour
P2ActiveColor	Partition 2 indicator active colour
P2InactiveColor	Partition 2 indicator inactive colour
P2StartingAngle	Partition 2 starting angle
P3ActiveColor	Partition 3 indicator active colour
P3InactiveColor	Partition 3 indicator inactive colour
P3StartingAngle	Partition 3 starting angle
PointerColor	Knob pointer colour
PointerLength	Knob pointer length (Not applicable for circle pointer style)
PointerSize	Knob pointer size (Not applicable for line pointer style)
PointerStyle	Knob pointer style
Top	Top-left Y-position
Width	Widget width

3.4.2. Internal Slider E

The function required for rendering this widget is discussed in the **gfx_Slider5** section of Pixxi or Diablo Internal Functions Reference Manual. The Slider widget is shown in the diagram below.

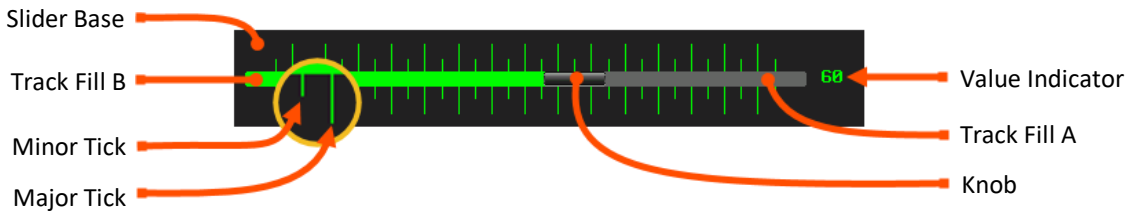
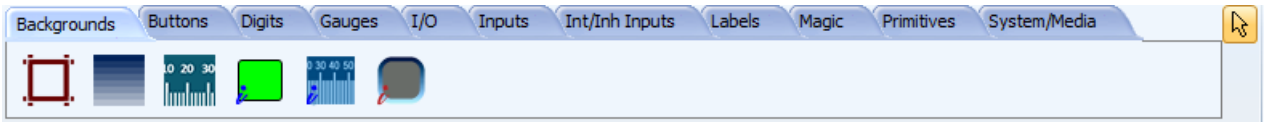


Figure 52: Slider E Diagram

Property	Description
Name	Widget name
Alias	Widget alias
BRminorTicks	Minor tick partitions between each major tick in the bottom/right side marker
BRtickCount	Total number of ticks for the bottom/right side marker
Color	Slider base colour
Font	Value indicator font, refer to the section **External Font** when using custom fonts
FontColor	Value indicator font colour
Height	Widget height
Indicator	Enable or disable value indicator
KnobColorA	Slider knob gradient start colour
KnobColorB	Slider knob gradient end colour
KnobGradient	Slider knob gradient direction/style
KnobOutlineColorA	Slider knob outline gradient start colour
KnobOutlineColorB	Slider knob outline gradient end colour
KnobOutlineGradient	Slider knob outline gradient direction/style
Left	Top-left X-position
MajorTickColor	Major tick colour
MajorTickLength	Major tick length
Maxvalue	Slider maximum value
MinorTickColor	Minor tick colour
MinorTickLength	Minor tick length
Minvalue	Slider minimum value
Orientation	Slider orientation
ProgressBar	Enable or disable progress bar mode (Set to Yes to turn the slider into a gauge)
Ticks	Enable or disable marker ticks
TLminorTicks	Minor tick partitions between each major ticks in the top/left side marker
TLtickCount	Total number of ticks for the top/left side marker
Top	Top-left Y-position
TrackFillColorA	Track fill colour from top/right side to current knob position
TrackFillColorB	Track fill colour from bottom/left side to current knob position
Width	Widget width

3.5. Internal Background Widgets

These are widgets commonly used for adding visual cues or design.



All background widgets can be found under the **Background** tab of the widget selection pane as shown above.

Internal (PmmC) background widgets can be identified with the blue *i* located at the bottom left of the widget icon.

3.5.1. Internal Panel B

The function required for rendering this widget is discussed in the **gfx_Panel2** section of Pixxi or Diablo Internal Functions Reference Manual. The Panel widget is shown in the diagram below.

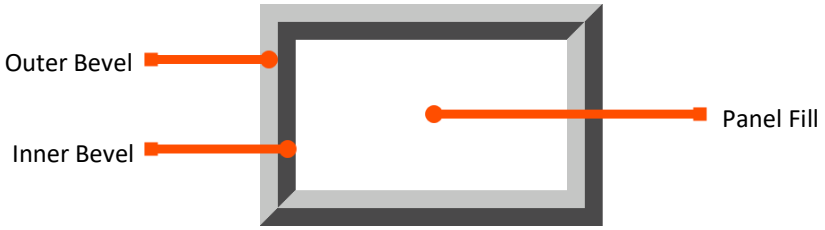


Figure 53: Panel B Diagram

Property	Description
Name	Widget name
Alias	Widget alias
ColorFill	Panel fill colour
Fill	Enable or disable panel fill
Height	Widget height
Left	Top-left X-position
LineColor1	First bevel colour
LineColor2	Second bevel colour
LineWidth1	Panel outer bevel thickness
LineWidth2	Panel inner bevel thickness
State	Panel bevel direction (Lowered or Raised)
Top	Top-left Y-position
Width	Widget width

3.5.2. Internal Scale

The function required for rendering this widget is discussed in the **gfx_Scale** section of Pixxi or Diablo Internal Functions Reference Manual. The Scale widget is shown in the diagram below.

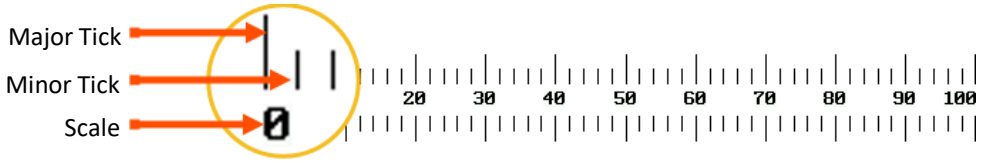


Figure 54: Scale Diagram

Property	Description
Name	Widget name
Alias	Widget alias
BackGroundColor	Numerical mark background colour
CentreGap	Gap size when scale is centred
Font	Scale font, refer to the section **External Font** when using custom fonts
FontColor	Scale font colour
Left	Top-left X-position
Length	Widget length
Maximum	Maximum value of the numerical mark
Minimum	Minimum value of the numerical mark
Orientation	Scale orientation (Horizontal or Vertical)
ScaleEndAlign	Enable or disable alignment of the scale end marks to the corners
ScaleLoc	Scale position relative to the central axis
ShowZero	Enable or disable the zero mark on the scale
TickColor	Tick colour
TickLength	Major tick length
TickLengthMinor	Minor tick length
TickMarks	Marker tick position
Ticks	Total number of major tick (Actual number of ticks = ticks – 1)
TicksMinor	Number of minor ticks between each major ticks
Top	Top-left Y-position

4. Inherent Widgets

The **Inherent Widgets** are the latest breakthrough in the 4D display graphics. It provides a way to deliver high quality graphical widgets without the use of microSD card.

These widgets are available for Pixxi and Diablo16 processors only. **These are not available for Goldelox or Picaso.**

In Pixxi processors, this type of widget requires external flash to store the resources for real time rendering. While in Diablo16, the same resources are stored in Flash Bank.

Like the Internal Widgets, this widget type is ideal for applications that requires complex graphics without using micro-SD cards.

Inherent Widgets requires a small amount of program space to store widget parameters. This reduces that amount of useable code space for the project. If the project will require a huge amount of code space and multiple widgets, it is highly advisable to use GCI widgets instead.

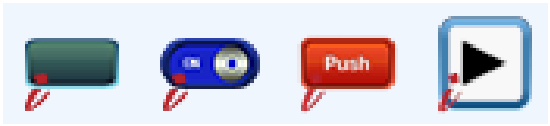
Pixxi projects utilizing inherent widgets needs to be redesigned to use GCI widgets instead. In Diablo, inherent widgets can be used together with GCI widgets.

Note: Since there is limited available space in Pixxi External Flash (or Diablo Flash Bank), only a few different types of inherent widget can be used. This should be considered when adding a new type of widget to the project.

4.2. Inherent Button Widgets

These are widgets commonly used for receiving touch inputs from the user.

All buttons can be found under the **Buttons** tab of the widget selection pane as shown above.



Inherent buttons can be identified with the red ***i*** located at the bottom left of the button icon.

4.2.1. Inherent Button E

The Button E widget is shown in the diagram below.

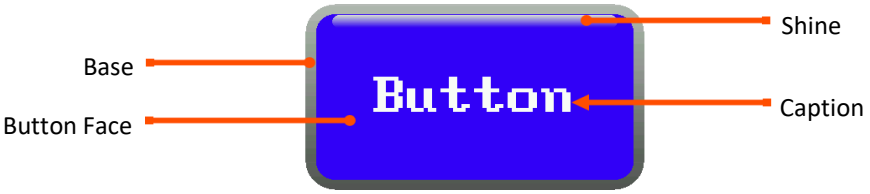


Figure 55: Button E Diagram

Figure 56: Button E OFF Position

Property	Description
Name	Widget name
Alias	Widget alias
BaseColorA	Base bevel gradient start colour
BaseColorB	Base bevel gradient end colour
BaseCornerRadius	Base bevel corner radius (Ideally less than half of widget height)
BaseOffset	Base bevel offset from the button face
FaceCornerRadius	Button face corner radius (Ideally equal or less than BaseCornerRadius)
Font	Caption text font, refer to the section **External Font** when using custom fonts
FontSize	Caption text font size
Height	Widget height
Left	Top-left X-position
Matrix	Button grouping (ViSi-Genie only)
Momentary	Button type
OFFColor	OFF position text label colour
OFFfaceColorA	OFF position button face gradient start colour
OFFfaceColorB	OFF position button face gradient start colour
OFFlabel	OFF position button text
ONColor	ON position text label colour
ONfaceColor	ON position button face colour
ONLabel	ON position button text
ShineColor	LED shine effect colour
ShineEffect	Enable or disable LED shine effect
ShineRadius	LED shine effect radius (Ideally less than (Height/2 – basewidth – 4))
Top	Top-left Y-position
Width	Widget width

4.2.2. Inherent Switch B

The Switch B widget is shown in the diagram below.

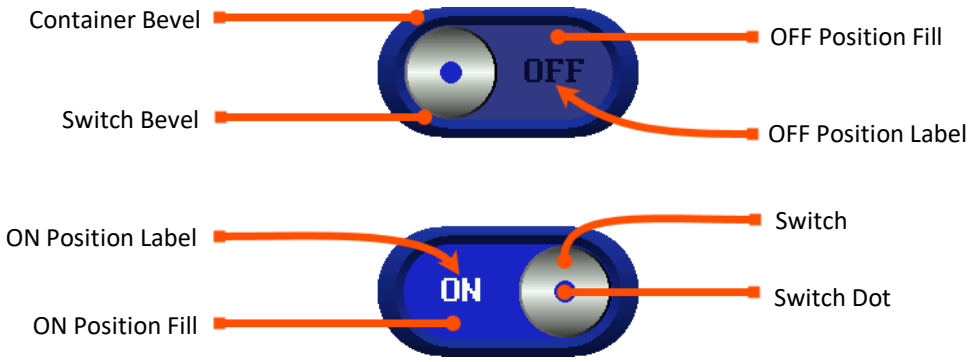


Figure 57: Switch B Diagram

Property	Description
Name	Widget name
Alias	Widget alias
BevelMainColor	Container and switch bevel main colour
BevelShadowColor	Container and switch bevel shadow colour
ContainerBevelThickness	Container bevel thickness
Font	Label font, refer to the section **External Font** when using custom fonts
FontSize	Label font size
Height	Widget height
Left	Top-left X-position
OFFColor	OFF position text label colour
OFFLabel	OFF position text label
OFFPositionFill	OFF position track fill colour
ONColor	ON position text label colour
ONLabel	ON position text label
ONPositionFill	ON position track fill colour
Orientation	Switch orientation
SwitchBevelThickness	Switch bevel thickness
Top	Top-left Y-position
Width	Widget width

4.2.3. Inherent Toggle Input

The Inherent Toggle Input Widget is a **Workshop4 PRO** widget that allows users to design their own buttons using 4DGL code. When added to a project, this widget is shown as a generic rectangular image as shown below.

This is merely a representation of the rectangular space occupied by the button and the colours selected for the widget properties. Note that the actual appearance will depend on the user code.

Property	Description
Name	Widget name
Alias	Widget alias
Code	Inherent Toggle Input User INC file
Color1	First colour available for user code
Color2	Second colour available for user code
Color3	Third colour available for user code
Color4	Fourth colour available for user code
Color5	Fifth colour available for user code
Color6	Sixth colour available for user code
Height	Widget height
Left	Top-left X-position
Matrix	Button grouping (ViSi-Genie only)
Momentary	Button type
Text1	First text available for user code
Text2	Second text available for user code
Text3	Third text available for user code
Top	Top-left Y-position
Value1	First value available for user code
Value2	Second value available for user code
Value3	Third value available for user code
Value4	Fourth value available for user code
Value5	Fifth value available for user code
Value6	Sixth value available for user code
Width	Widget width

4.2.4. Inherent Media Button

The Media Button widget is shown in the diagram below.

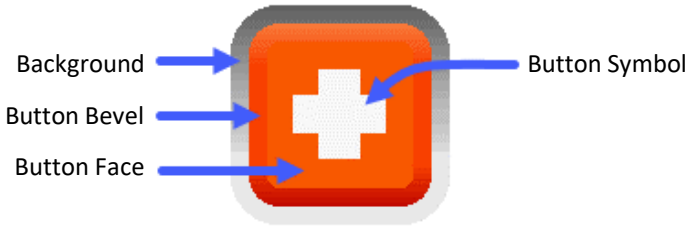


Figure 58. Media Button Diagram

The media button has 47 available Button Styles as shown below.


ROUND


ROUND FLAT


ROUND MEDIUM FLAT


SQUARE


SQUARE FLAT


ROUND RAISED SUNKEN


ROUND SUNKEN RAISED


ROUND SUNKEN FLAT


HALFMOON RIGHT FLAT


KEYPAD ROUND


KEYPAD SQUARE


UP


DOWN


LEFT


RIGHT


UP FLAT

ROUND SUNKEN SUNKEN

SQUARE RAISED SUNKEN

SQUARE SUNKEN RAISED

SQUARE SUNKEN FLAT

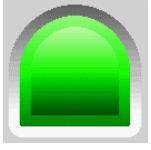
SQUARE SUNKEN SUNKEN

HALFMOON UP SUNKEN

HALFMOON DOWN SUNKEN

HALFMOON LEFT SUNKEN

HALFMOON RIGHT SUNKEN



HALFMOON UP RAISED

HALFMOON DOWN RAISED

DOWN FLAT

LEFT FLAT

RIGHT FLAT

LED1 ROUND

LED1 SQUARE

LED1 ROUND FLAT

LED1 SQUARE FLAT

LED1 ROUND SUNK

LED1 SQUARE SUNK

LED2 ROUND

LED2 SQUARE

HALFMOON LEFT RAISED

HALFMOON RIGHT RAISED

HALFMOON UP FLAT

HALFMOON DOWN FLAT

HALFMOON LEFT FLAT

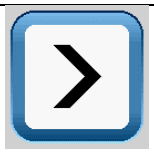
LED2 ROUND FLAT

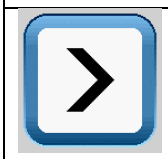
LED2 SQUARE FLAT

LED2 ROUND SUNK

LED2 SQUARE SUNK

The Media Button has 21 different types of Button Symbols as shown below.

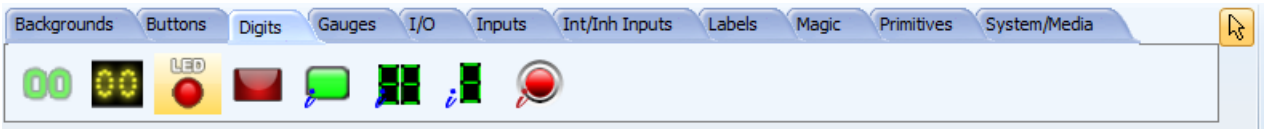
	Nothing		Pause
	TEXT SYMBOL		Next
	INC VERTICAL		Prev
	DEC VERTICAL		FF
	INC HORIZONTAL		Rewind

	**DEC HORIZONTAL**		**STOP**
	**INC HORIZONTAL THIN**		**PLUS**
	**DEC HORIZONTAL THIN**		**MINUS**
	**INC VERTICAL THIN**		**PLUS THIN**
	**DEC VERTICAL THIN**		**MINUS THIN**
	**PLAY**		

Property	Description
Name	Widget name
Alias	Widget alias
BackgroundColor	Background colour for the sunken ring
Font	Text font style, refer to the section **External Font** when using custom fonts
FontMagnification	Text font style
Height	Widget height
InnerGradColor	Button face colour
LedOffColor	LED Off colour
LedOnColor	LED On colour
Left	Top-left X-position
Matrix	Button grouping (ViSi-Genie only)
Momentary	Button type
OuterGradColor	Button face bevel colour
OuterGradWidth	Button face bevel width
RingWidth	Button recess width (0 to disable)
Style	Button style
Symbol	Symbol type
SymbolColor	Symbol colour
Text	Text symbol string input
Top	Top-left Y-position
Width	Widget width

4.3. Inherent Digits Widgets

These are widgets commonly used for displaying values or status information.



All Digits widgets can be found under the **Digits** tab of the widget selection pane as shown above. This pane is mostly composed of Leds and Digits.

Inherent widgets can be identified with the red ***i*** located at the bottom left of the widget icon.

4.3.1. Inherent Media LED

The Media LED widget is shown in the diagram below.

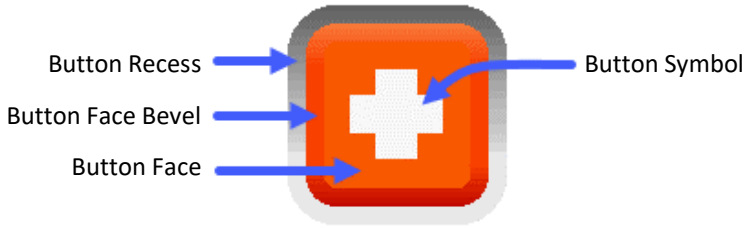


Figure 59. Media LED Diagram

The media button has 12 available LED Styles as shown below.

LED1 ROUND

LED2 ROUND

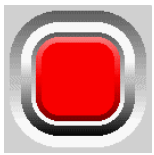
LED1 SQUARE

LED2 SQUARE

LED1 ROUND FLAT

LED2 ROUND FLAT

LED1 SQUARE FLAT



LED2 SQUARE FLAT

LED1 ROUND SUNK

LED2 ROUND SUNK

LED1 SQUARE SUNK

LED2 SQUARE SUNK

The Media Button has 21 different types of Button Symbols as shown below.

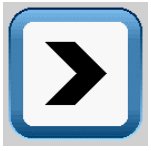
MEDIA_NO_SYMBOL

MEDIA_PAUSE

MEDIA_TEXT_SYMBOL

MEDIA_INC_VERTICAL

MEDIA_DEC_VERTICAL



MEDIA_INC_HORIZONTAL

MEDIA_DEC_HORIZONTAL

MEDIA_INC_HORIZONTAL_THIN

MEDIA_DEC_HORIZONTAL_THIN

MEDIA_INC_VERTICAL_THIN

MEDIA_DEC_VERTICAL_THIN

MEDIA_PLAY

MEDIA_NEXT

MEDIA_PREV

MEDIA_FASTFORWARD

MEDIA_REWIND

MEDIA_STOP

MEDIA_PLUS

MEDIA_MINUS

MEDIA_PLUS_THIN

MEDIA_MINUS_THIN

Property	Description
Name	Widget name
Alias	Widget alias
BackgroundColor	Background colour for sunken ring
Font	Label font style, refer to the section **External Font** when using custom fonts
FontMagnification	Font size multiplier for optional text
Height	Widget height
InnerGradColor	Button face colour
LedOffColor	LED Off colour
LedOnColor	LED On colour
Left	Top-left X-position
OuterGradColor	LED face bevel colour
OuterGradWidth	LED face bevel width
RingWidth	LED recess width (0 to disable)
Style	LED style
Symbol	Symbol type
SymbolColor	Symbol colour
Text	Text symbol string input
Top	Top-left Y-position
Width	Widget width

4.4. Inherent Gauge Widgets

These are widgets commonly used for displaying values.

All gauges can be found under the **Gauges** tab of the widget selection pane as shown above.

Inherent gauges can be identified with the red ***i*** located at the bottom left of the gauge icon.

4.4.1. Inherent User Gauge

The Inherent User Gauge widget is a **Workshop4 PRO** widget that allows users to design their own gauges using 4DGL code. When added to a project, this widget is shown as a generic rectangular image as shown below.

This is merely a representation of the rectangular space occupied by the gauge and the colours selected for the widget properties. Note that the actual appearance will depend on the user code.

Property	Description
Name	Widget name
Alias	Widget alias
Code	Inherent User Gauge INC file
Color1	First colour available for user code
Color2	Second colour available for user code
Color3	Third colour available for user code
Color4	Fourth colour available for user code
Color5	Fifth colour available for user code
Color6	Sixth colour available for user code
Height	Widget height
Left	Top-left X-position
Text1	First text available for user code
Text2	Second text available for user code
Text3	Third text available for user code
Top	Top-left Y-position
Value1	First value available for user code
Value2	Second value available for user code
Value3	Third value available for user code
Value4	Fourth value available for user code
Value5	Fifth value available for user code
Value6	Sixth value available for user code
Width	Widget width

4.4.2. Inherent Media Gauge

The Media Gauge widget is shown in the diagram below.

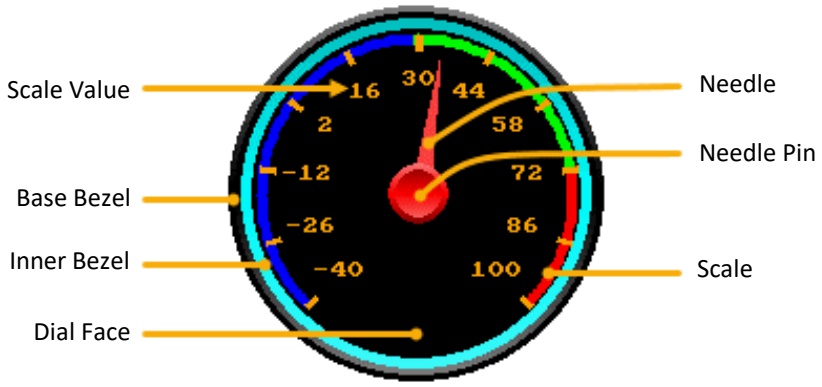


Figure 60: Media Gauge diagram.

The Media Gauge has sixteen (16) available styles as shown below.

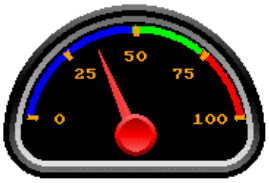
Circular

HalfmoonLeft

Square

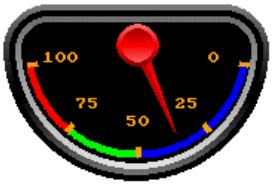
HalfmoonRight

Rectangular



HalfmoonTop

Meter



HalfmoonBottom

QuartermoonTopLeft

SquareTopLeft

QuartermoonTopRight

SquareTopRight

QuartermoonBottomLeft

SquareBottomLeft

QuartermoonBottomRight

SquareBottomRight

Property	Description
Name	Widget name
Alias	Widget alias
Bezel	Enable or disable bezel
BezelBackground	Bezel effect
BezelColor	Bezel colour
BezelGradient	Bezel gradient style
BezelInnerGradient	Bezel Inner gradient style
DialColor	Dial face colour
DialGradient	Dial inner gradient style
Font	Ticks values font, refer to the section **External Font** when using custom fonts
FontMagnification	Ticks value magnification
GaugeEndAngle	End angle of the scale
GaugeStartAngle	Start angle of the scale
Height	Height of the Widget
HighColor	High range scale colour
InnerBezelColor	Inner Bezel Colour
InnerWidth	Width of ring inside the bezel
LabelOffset	Offset of labels from edge excluding bezel. Can be negative
Left	Top-left X-position
LowColor	Low range scale colour
LowPercent	Coloured scale low range percent value

MedColor	Medium range scale colour
MedPercent	Coloured scale medium range value
MeterCaption	Text for meter caption
NeedleColor	Colour of needle and pin
OuterWidth	Width of bezel
RangeHigh	Maximum value of range
RangeLow	Minimum value of range
Scale	Enable or disable scale
ScaleDivisor	Divide range value to print smaller tick values
ShineLevel	Shine level (-30 to 30)
Style	Style of gauge
TextColor	Text colour
TextEnd	High range end text (will remove the tick values if filled with character or string)
TextStart	Low range end text (will remove the tick values if filled with character or string)
TickColor	Ticks colour
Ticks	Number of ticks
Top	Top-left Y-position
Width	Widget width

4.4.3. Inherent Media Thermometer

The Fancy Thermometer widget is shown in the diagram below.

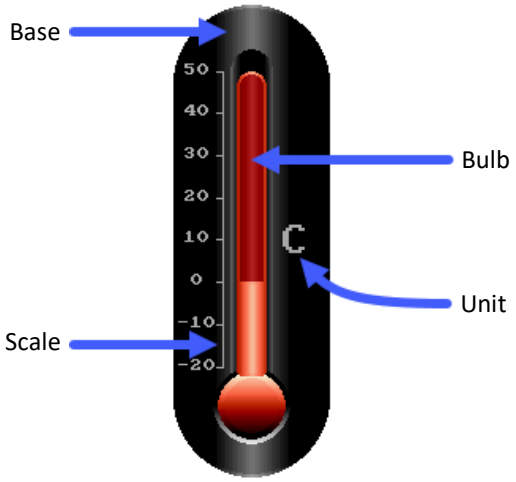
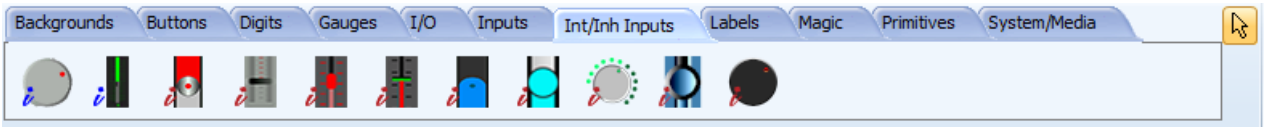


Figure 61: Fancy Thermometer Diagram.

Property	Description
Name	Widget name
Alias	Widget alias
BaseColor	Base colour
BulbColor	Color of the thermometer bulb
Font	Font for C or F Unit, refer to the section **External Font** when using custom fonts
FontMagnification	Font Maginification for C or F Unit
Height	Widget height
Left	Top-left X-position
Max	Maximum value
Metric	Select Unit Metric (C) or Imperial (F)
Min	Minimum value
ScaleColor	Scale tick colour
ScaleFont	Scale values font style, refer to the section **External Font** when using custom fonts
ScaleFontMagnification	Maginification of scale font
TextColor	Font colour for scale value and label
Ticks	Number of ticks
Top	Top-left Y-position
Width	Widget width

4.5. Inherent Input Widgets

These are widgets commonly used for receiving touch inputs from the user.



All Inherent input widgets can be found under the **Int/Inh Inputs** tab of the widget selection pane as shown above.

Inherent sliders and knobs can be identified with the red ***i*** located at the bottom left of the widget icon.

4.5.1. Inherent Slider C

The Slider C widget is shown in the diagram below.

Figure 62: Slider C Diagram

Property	Description
Name	Widget name
Alias	Widget alias
Frames	Slider maximum value
Height	Widget height
InnerGradient	Slider container inner lining gradient direction/style
InnerGradientColorA	Slider container inner lining gradient start colour
InnerGradientColorB	Slider container inner lining gradient end colour
KnobColorA	Slider knob gradient start colour
KnobColorB	Slider knob gradient end colour
KnobFaceDotColor	Slider knob dot colour
KnobGradient	Slider knob gradient direction/style
KnobOutlineColorA	Slider knob outline gradient start colour
KnobOutlineColorB	Slider knob outline gradient end colour
KnobOutlineGradient	Slider knob outline gradient direction/style
Left	Top-left X-position
Orientation	Slider orientation
OuterGradient	Slider container outer lining gradient direction/style
OuterGradientColorA	Slider container outer lining gradient start colour
OuterGradientColorB	Slider container outer lining gradient end colour
ProgressBar	Enable or disable progress bar mode (Set to Yes to turn the slider into a gauge)
Top	Top-left Y-position
TrackFillColorA	Track fill colour from top/right side to current knob position
TrackFillColorB	Track fill colour from bottom/left side to current knob position
Width	Widget width

4.5.2. Inherent Slider D

The Slider D widget is shown in the diagram below.

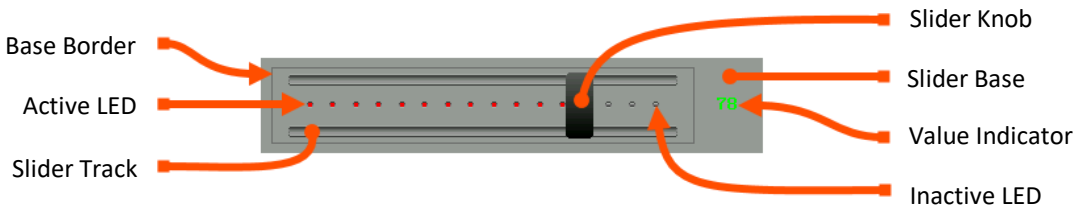


Figure 63: Inherent Slider D Diagram

Property	Description
Name	Widget name
Alias	Widget alias
BorderColor	Slider base border colour
Color	Slider base colour
Font	Value indicator font, refer to the section **External Font** when using custom fonts
FontColor	Value indicator font colour
Height	Widget height
Indicator	Enable or disable value indicator
KnobGradient	Slider knob gradient direction/style
KnobOutlineColorA	Slider knob gradient start colour
KnobOutlineColorB	Slider knob gradient end colour
LEDActiveColor	Active LED colour
LEDColorA	LED outline gradient start colour
LEDColorB	LED outline gradient end colour
LEDGradient	LED outline gradient direction/style
LEDInactiveColorA	Inactive LED gradient start colour
LEDInactiveColorB	Inactive LED gradient end colour
LEDInactiveGradient	Inactive LED gradient direction/style
LEDPartition	Number of partitions between each LED
LEDRadius	LED radius
Left	Top-left X-position
Maxvalue	Slider maximum value
Minvalue	Slider minimum value
Orientation	Slider orientation
SwitchMode	Enable or disable multiposition switch mode
Top	Top-left Y-position
TrackFillColorA	Track gradient start colour
TrackFillColorB	Track gradient end colour
TrackOutlineColorA	Track outline gradient start colour
TrackOutlineColorB	Track outline gradient end colour
Width	Widget width

4.5.3. Inherent Slider F

The Slider F widget is shown in the diagram below.

Figure 64: Slider F Diagram

Property	Description
Name	Widget name
Alias	Widget alias
Color	Slider base colour
Font	Value indicator font, refer to the section **External Font** when using custom fonts
FontColor	Value indicator font colour
Height	Widget height
Indicator	Enable or disable value indicator
KnobColor	Slider knob colour
Left	Top-left X-position
Maxvalue	Slider maximum value
Minvalue	Slider minimum value
Orientation	Slider orientation
ProgressBar	Enable or disable progress bar mode (Set to Yes to turn the slider into a gauge)
TickColor	Tick colour
TickCount	Total number of ticks
Ticks	Enable or disable marker ticks
TickWidth	Tick thickness
Top	Top-left Y-position
TrackFillColorA	Track fill colour from top/right side to current knob position
TrackFillColorB	Track fill colour from bottom/left side to current knob position
Width	Widget width

4.5.4. Inherent Slider G

The Slider G widget is shown in the diagram below.

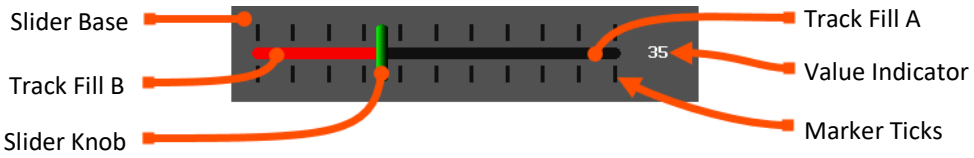


Figure 65: Inherent Slider G Diagram

Property	Description
Name	Widget name
Alias	Widget alias
Color	Slider base colour
Font	Value indicator font, refer to the section **External Font** when using custom fonts
FontColor	Value indicator font colour
Height	Widget height
Indicator	Enable or disable value indicator
KnobColor	Slider knob colour
Left	Top-left X-position
Maxvalue	Slider maximum value
Minvalue	Slider minimum value
Orientation	Slider orientation
ProgressBar	Enable or disable progress bar mode (Set to Yes to turn the slider into a gauge)
TickColor	Tick colour
TickCount	Total number of ticks
Ticks	Enable or disable marker ticks
TickWidth	Tick thickness
Top	Top-left Y-position
TrackFillColorA	Track fill colour from top/right side to current knob position
TrackFillColorB	Track fill colour from bottom/left side to current knob position
Width	Widget width

4.5.5. Inherent Slider H

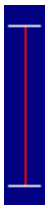
The Slider H widget is shown in the diagram below.

Figure 66: Slider H Diagram

Property	Description
Name	Widget name
Alias	Widget alias
Color	Slider base colour
Frames	Slider maximum value
Height	Widget height
KnobDotColor	Slider dot colour
Left	Top-left X-position
Orientation	Slider orientation
ProgressBar	Enable or disable progress bar mode (Set to Yes to turn the slider into a gauge)
Top	Top-left Y-position
TrackFillColorA	Track fill colour from top/right side to current knob position
TrackFillColorB	Track fill colour from bottom/left side to current knob position
Width	Widget width

4.5.6. Inherent Linear Input

The Inherent Linear Input widget is a **Workshop4 PRO** widget that allows users to design their own linear inputs such as sliders and trackbars using 4DGL code. When added to a project, this widget is shown as a generic rectangular image as shown below.



This is merely a representation of the rectangular space occupied by the linear input and the colours selected for the widget properties. Note that the actual appearance will depend on the user code.

Property	Description
Name	Widget name
Alias	Widget alias
Code	Inherent Linear Input INC file
Color1	First colour available for user code
Color2	Second colour available for user code
Color3	Third colour available for user code
Color4	Fourth colour available for user code
Color5	Fifth colour available for user code
Color6	Sixth colour available for user code
Height	Widget height
Left	Top-left X-position
Maxvalue	Value at maximum position
Minvalue	Value at minimum postion
OffsetBR	Touch offset at Bottom (vertical) or Right (horizontal)
OffsetTL	Touch offset at Top (vertical) or Left (horizontal)
Orientation	Linear input orientation
Text1	First text available for user code
Text2	Second text available for user code
Text3	Third text available for user code
Top	Top-left Y-position
Value1	First value available for user code
Value2	Second value available for user code
Value3	Third value available for user code
Value4	Fourth value available for user code
Value5	Fifth value available for user code
Value6	Sixth value available for user code
Width	Widget width

4.5.7. Inherent Rotary Input

The Inherent Rotary Input widget is a **Workshop4 PRO** widget that allows users to design their own circular inputs using 4DGL code. When added to a project, this widget is shown as a generic rectangular image as shown below.

This is merely a representation of the rectangular space occupied by the rotary input and the colours selected for the widget properties. Note that the actual appearance will depend on the user code.

Property	Description
Name	Widget name
Alias	Widget alias
Code	Inherent User Gauge INC file
Color1	First colour available for user code
Color2	Second colour available for user code
Color3	Third colour available for user code
Color4	Fourth colour available for user code
Color5	Fifth colour available for user code
Color6	Sixth colour available for user code
Height	Widget height
Left	Top-left X-position
MaxAngle	Angle position at maximum value (0 is East)
Maxvalue	Value at maximum position
MinAngle	Angle position at minimum value (0 is East)
Minvalue	Value at minimum position
Text1	First text available for user code
Text2	Second text available for user code
Text3	Third text available for user code
Top	Top-left Y-position
Value1	First value available for user code
Value2	Second value available for user code
Value3	Third value available for user code
Value4	Fourth value available for user code
Value5	Fifth value available for user code
Value6	Sixth value available for user code
Width	Widget width

4.5.8. Inherent Media Slider

The Media Slider widget is shown in the diagram below.

Figure 67: Media Slider diagram

Property	Description
Name	Widget name
Alias	Widget alias
BackGroundColor	Base colour
Frames	Slider maximum value starting from zero
Height	Widget height
KnobCentreCcolor	Slider knob centre face colour
KnobCentreSunken	Slider knob centre face direction (Yes for Sunken, No for Raised)
KnobOuterColor	Slider knob ring colour
KnobRingSunken	Slider knob ring direction (Yes for Sunken, No for Raised)
Left	Top-left X-position
NewOW	Slider knob ring thickness
Orientation	Widget orientation
ScaleBevelColor	Scale bevel colour
ScaleBevelSunken	Scale bevel direction (Yes for Sunken, No for Raised)
ScaleBevelWidth	Scale bevel width
ScaleCentreSunken	Scale centre fill direction (Yes for Sunken, No for Raised)
ScaleHeight	Scale centre fill thickness
ScaleLeftColor	Colour on top/left fill of the scale
ScaleRightColor	Colour on bottom/right fill of the scale
ShineLevel	Shine effect value
Top	Top-left Y-position
Width	Widget width

4.5.9. Inherent Media Rotary

The Media Rotary widget is shown in the diagram below.

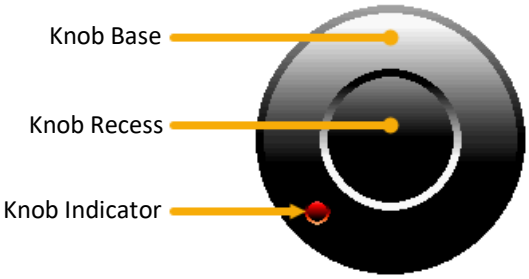
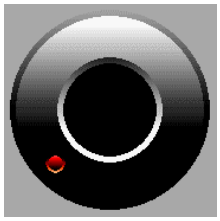


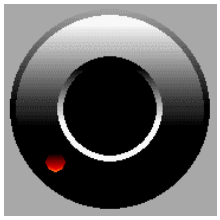
Figure 68: Media Rotary diagram.

The Media Rotary widget has fourteen (14) available input styles as shown below.

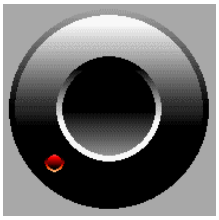


Style00

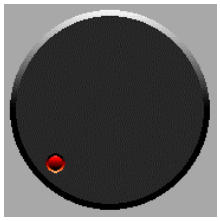
Style07



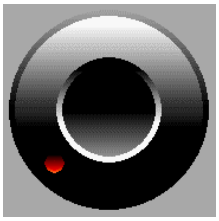
Style01



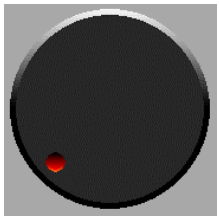
Style08



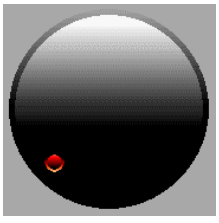
Style02



Style09

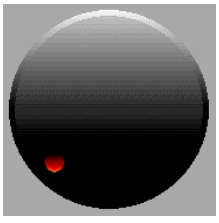


Style03



Style10

Style04

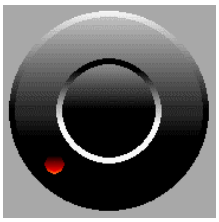


Style11

Style05

Style12

Style06

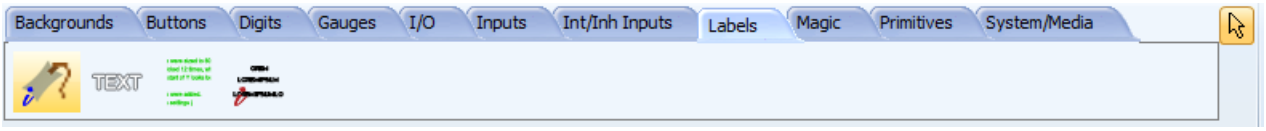


Style13

Property	Description
Name	Widget name
Alias	Widget alias
Frames	Maximum value starting from zero
Height	Widget height
IndicatorOffset	Indicator position from the centre of rotation
IndicatorRadius	Indicator radius
IndInnerColor	Indicator colour
IndOuterColor	Indicator bevel colour
InnerDiameter	Knob recess diameter
InnerEdgeColor	Knob recess bevel colour
InnerEdgeWidth	Knob recess bevel thickness
InnerRadiusColor	Knob recess face colour
Left	Top-left X-position
MaxAngle	Maximum angle
MinAngle	Minimum angle
OuterEdgeColor	Knob base bevel colour
OuterEdgeWidth	Knob base bevel thickness
OuterRadiusColor	Knob base colour
RotaryInputStyle	Rotary input style
ShineLevel	Shine Level (0-21)
Top	Top-left Y-position
Width	Widget width

4.6. Inherent Label Widgets

These are widgets commonly used for displaying text information.



All label widgets can be found under the **Labels** tab of the widget selection pane as shown above.

Inherent widgets can be identified with the red ***i*** located at the bottom left of the widget icon.

4.6.1. Inherent Label B

The Inherent Label B widget is shown in the diagram below.

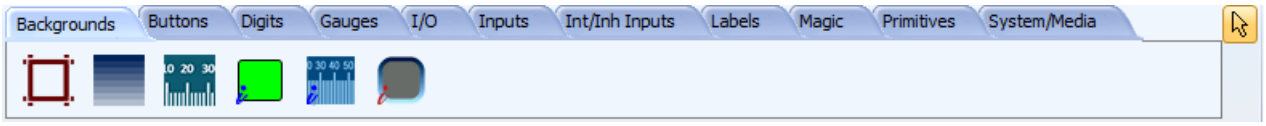
Figure 69: Inherent Label B Diagram

The text inside the label can be aligned as shown below.

Property	Description
Name	Widget name
Alias	Widget alias
Alignment	Text alignment inside the textbox
BGcolor	Label background colour
Buffer	Maximum textbox string buffer size (including trailing null)
Caption	Label text string input
Font	Text font style, refer to the section **External Font** when using custom fonts
FontColor	Text colour
FontSize	Text font size
Height	Textbox height
Left	Top-left X-position
Margin	Margin from top, left for left alignment, right for right alignment
Top	Top-left Y-position
Width	Widget width

4.7. Inherent Background Widgets

These are widgets commonly used for adding visual cues or design.



All background widgets can be found under the **Background** tab of the widget selection pane as shown above.

Inherent background widgets can be identified with the red ***i*** located at the bottom left of the widget icon.

4.7.1. Inherent Media Panel

The Media Panel widget is shown in the diagram below.

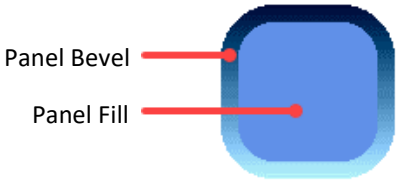


Figure 70: Media Panel Diagram

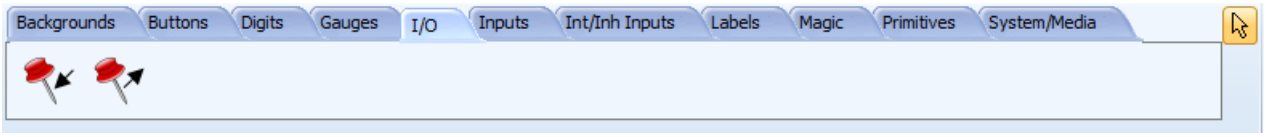
Property	Description
Name	Widget name
Alias	Widget alias
Left	Top-left X-position
Top	Top-left Y-position
Width	Widget width
Height	Widget height
Bevelcolour	Bevel colour
Panelcolour	Panel colour
Outerwidth	Width of bezel
RadTopLeft	Fillet radius of top-left corner
RadTopRight	Fillet radius of top-right corner
RadBottomLeft	Fillet radius of bottom-left corner
RadBottomRight	Fillet radius of bottom-right corner
BevelGradSunken	Raise or sunk bezel gradient
PanelGradSunken	Raise or sunk panel gradient
BevelGradLevel	Bezel Gradient Level. 0-63, -1 is top and bottom gradient
PanelGradLevel	Panel Gradient Level. 0-63, -1 is top and bottom gradient

5. Miscellaneous Widgets

There are several non-GUI widgets available to use depending on environment and target module. This includes hardware I/O, timer and sound related functionalities.

5.1. Pin I/O Widgets

Pin I/O Widgets are available to use when in ViSi Genie environment with a compatible processor.



5.1.1. Pin Input

The pin input widget allows ViSi Genie projects to add basic digital input functionality.

Property	Description
Name	Widget name
Alias	Widget alias
IdleState	State of the pin when idle
Pin	Pin number

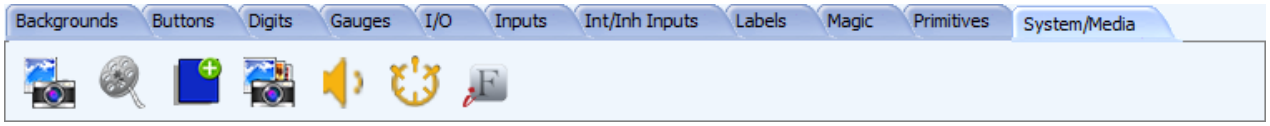
5.1.2. Pin Output

The pin output widget allows ViSi Genie projects to add basic digital output functionality.

Property	Description
Name	Widget name
Alias	Widget alias
IdleState	State of the pin when idle
PulseDuration	Duration of pulse when active
Pin	Pin number

5.2. System/Media

Several system and media widgets are available for the user.



 **Sound**

 **Timer**

 **External Font**

5.2.1. Sound

The sound widget allows **ViSi Genie** projects to add basic sound functionality.

Property	Description
Name	Widget name
Alias	Widget alias
WavFiles	List of wav files to be played

The Sound object is different to other objects in that there is only one of them (**Sounds0**) and that the values have predefined meanings. Performing a write object sets them. Reading Index 0 returns the number of blocks left to play.

Here are the object indexes used by the Sound Widget.

Object Index	Description
0	Play wav file at specified value
1	Set Volume to specified value
2	Pause
3	Continue
4	Stop

Note: The Sound object (like the Timer object) will always reside in Form0.

5.2.2. Timer

The timer widget allows **ViSi Genie** projects to add basic timer functionality.

Property	Description
Name	Widget name
Alias	Widget alias
Enabled	Initial state of the timer Set to yes to indicate the timer is to start when the program is loaded. Once enabled, the timer continues until the Video displays its last frame.
Interval	The number of milliseconds between timer events.

Note: The Sound object (like the Timer object) will always reside in Form0.

5.2.3. External Font

External Fonts provides Workshop4 projects the ability to render custom fonts from external storage devices such as a microSD card or a flash chip.

When using this feature in a ViSi project, the following are required:

- The first (global) variable in your program must be an array of 'handles'.
- If using an external flash:
  - the elements of the array should be the indexes of the font in the Flash File system
  - An Inherent/Internal widget should use -1-array_index when using the font in its widget definition
- If using a microSD card:
  - the elements of the array should be the font 'handle' that can be acquired using **file_LoadImageControl**
  - An Inherent/Internal widget should use -8193-array_index when using the font in its widget definition

These requirements are done automatically in ViSi Genie projects.

Property	Description
Name	Widget name
Alias	Widget alias
Bold	Set if the font is **bold**
CharSet	Set whether the character set covers ANSI or Unicode set
FontName	Font Family
Italic	Set if the font is *italic*
Size	Size of the font
Strikethrough	Set if the font is ~~strikethrough~~
Underline	Set if the font is underlined

6. Revision History

Revision	Revision Content	Revision Date
1.0	Initial release version	07/02/2020
1.1	Document title changes, GCI, Internal, Inherent for better readability Fixed a few minor typos with widget names Added miscellaneous widgets	23/07/2020

7. Legal Notice

Proprietary Information

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

4D Labs Semiconductors 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 Labs Semiconductors products and services is continuous and published information may not be up to date. It is important to check the current position with 4D Labs Semiconductors. 4D Labs Semiconductors reserves the right to modify, update or makes changes to Specifications or written material without prior notice at any time.

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

Disclaimer of Warranties & Limitation of Liability

4D Labs Semiconductors makes no warranty, either expressed 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.

Images and graphics used throughout this document are for illustrative purposes only. All images and graphics used are possible to be displayed on the 4D Labs Semiconductors range of products, however the quality may vary.

In no event shall 4D Labs Semiconductors 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 Labs Semiconductors, or the use or inability to use the same, even if 4D Labs Semiconductors has been advised of the possibility of such damages.

4D Labs Semiconductors 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 Labs Semiconductors and its suppliers specifically disclaim any expressed or implied warranty of fitness for High Risk Activities.

Use of 4D Labs Semiconductors’ 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 Labs Semiconductors from any and all damages, claims, suits, or expenses resulting from such use. No licenses are conveyed, implicitly or otherwise, under any 4D Labs Semiconductors intellectual property rights.

8. Contact Information

For Technical Support: www.4dlabs.com.au/support
For Sales Support: sales@4dlabs.com.au
Website: www.4dlabs.com.au