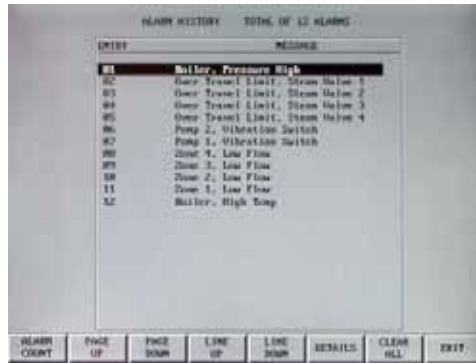


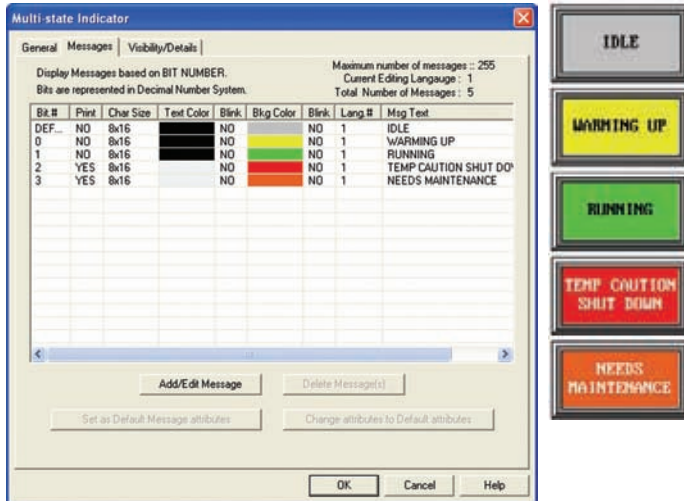
## Alarm History

Use our pre-built Alarm History button to show Alarm Count and Alarm History with one touch of a button. Alarm History displays all the alarms triggered sequentially with the most recent one right on the top, whereas Alarm Count displays the exact number of times a certain alarm has been triggered. Use this great preventive maintenance tool to replace any components that need to be changed. Any time an alarm is highlighted and selected, it will show you all the details that you'll ever need to know including the time it was triggered, the time it was cleared, date stamp, upper and lower limits along with the limit that tripped an alarm.



## Multi-state Indicator

Display preprogrammed messages within a frame. Each object has its own message storage and does not need an external message database. Up to 256 messages can be stored, and the one message that is displayed is based upon the value of a bit or a tag. Messages can also embed data variables. In addition, this object can be used as an indicator light, displaying only colors without messages.



## Increment/Decrement Value

This button allows addition or subtraction from a value using two predefined tags and a preprogrammed value. Once pressed, this object will read the value from the first tag, add/subtract the value defined, and write the new value to the second tag.



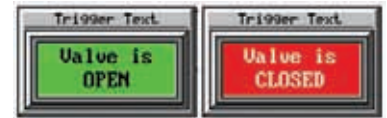
## Static Text

Place text anywhere on the screen to provide information, screen description, etc. As with any other object, you can fully customize the colors and size, choose whether to display a frame or not, and whether or not you want its background to be transparent.



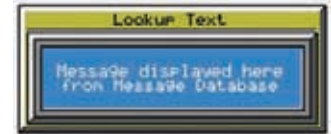
## Trigger Text

This object monitors a bit to display different text strings for "ON" and "OFF" conditions. This would be used in applications where you want to provide a message or a description of the process or condition.



## Lookup Text

Similar to the Multi-state indicator object, create a Lookup Text object to display pre-programmed messages within a frame on the screen. The difference is these messages are stored in the "Message Database" which acts as a global database for any lookup text object to reference throughout the project. A value corresponding to the tag name is the message number that will be displayed inside the frame. Messages are numbered from 1 to 999, so if the value corresponding to the tag name is 10 for example, the message number 10 will be displayed.



## Dynamic Text

The Dynamic Text object will allow you to display the characters from ASCII values stored in a Tag. The tag will read a block of registers in the PLC. Each 16-bit register in the PLC can contain 2 ASCII characters. The maximum number of PLC registers in the block is 20 (a maximum of 40 ASCII characters). This object is typically used for displaying part numbers, VIN numbers, or production numbers. Dynamic Text is triggered by a bit Tag in the PLC. You choose whether the Text is triggered by the bit when it is in the ON state or the OFF state. The Dynamic Text object will then display a text string that is programmed in the PLC.

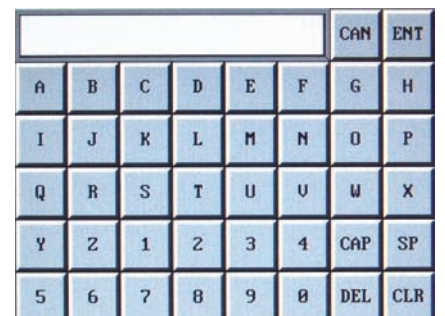


## Text Entry

The Text Entry object, when pressed on the panel, brings up a character entry (alphanumeric) keypad. This allows the operator to enter text up to 40 characters to send to a tag assigned to an address in a PLC.



It has many uses, some of which may be: to send part numbers or production numbers to a PLC, or to send a message to a PLC that will, in turn, route it to one or more plant floor message display(s), such as ToughMarquee.



## Adjust Contrast

Use the Adjust Contrast object to place a button on the UTTouch screen that gives you access to the panel's adjust contrast feature. Use the UP and DOWN arrows that appear on the bottom to adjust the screen contrast.



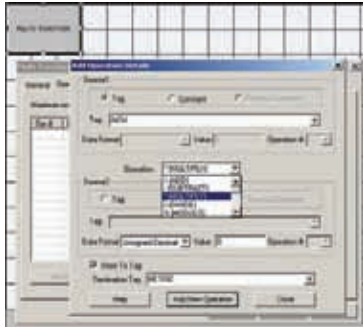
## Increment/Decrement Hour

Place an object on the screen that allows you to adjust the hour (up or down) of the internal Real-time clock.



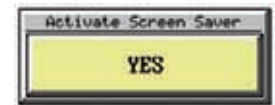
## Multi-Function Math/Logical Operation

The Multi-function Object allows you to configure a button, that when pressed, will perform a Boolean or Arithmetic operation using two tags and will store the result in a third tag. The operations supported are + (ADD), - (SUBTRACT), \* (MULTIPLY), / (DIVIDE), % (MODULO), ~ (NEGATE), II (ABSOLUTE), (ROUND), & (AND), I (OR), ~I (XOR), !(NOT), << (LEFT SHIFT), >> (RIGHT SHIFT), and (MOVE).



## Activate Screen Saver

Place a button on the screen that enables you to activate the panel screen saver, to lengthen the longevity of the panel.



## Select Language

This object allows the operator to change the language by pressing the button on the panel. Text that has been programmed for that language will convert to the language that the operator selects.

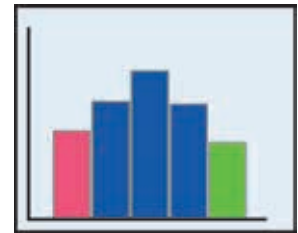


## Statistical Process Control (SPC)

Toughpanel can perform a variety of Statistical computations on the data stored in the FIFO, which is useful for SPC (Statistical Process Control).

The Toughpanel performs and displays the following computations:

- Mean or X-bar
- Median
- Range
- Min-Max
- Mode
- Cpk



## uWin<sup>08</sup> Programming Editor allows you to create Dynamic Bitmaps, Multi-state Bitmaps, Button Bitmaps, and Static Bitmap Objects in a breeze.

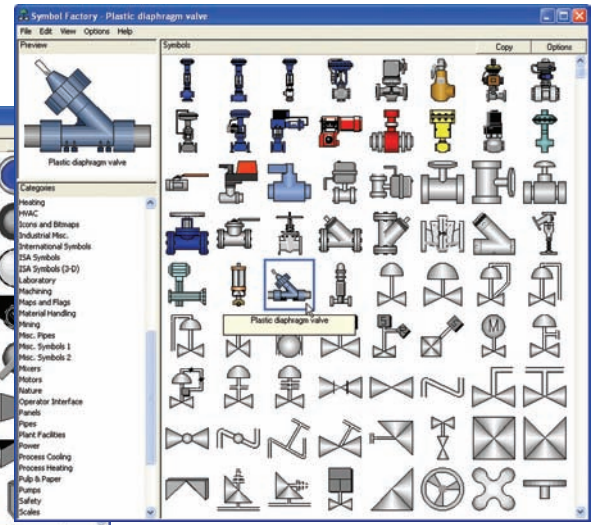
There is a built in library of 4,000 objects available for you to copy and paste directly to a project screen and use them in any arrangement or position them just the way you like.

## Static Bitmap

A static bitmap lets you simply display a bitmap which can be resized within the editor and stays static on a screen, e.g. a company logo.

All bitmaps can be imported, copied from the clipboard, or pulled in from the symbol factory (as shown to the right) and the Toughpanel supports the following formats: .bmp, .wmf, .emf, .gif, .jpeg, .jpg, and .ico.

The Symbol Factory's 4,000+ symbols are available to all bitmap objects in the uWin<sup>08</sup> Programming software.



## Bitmap Button

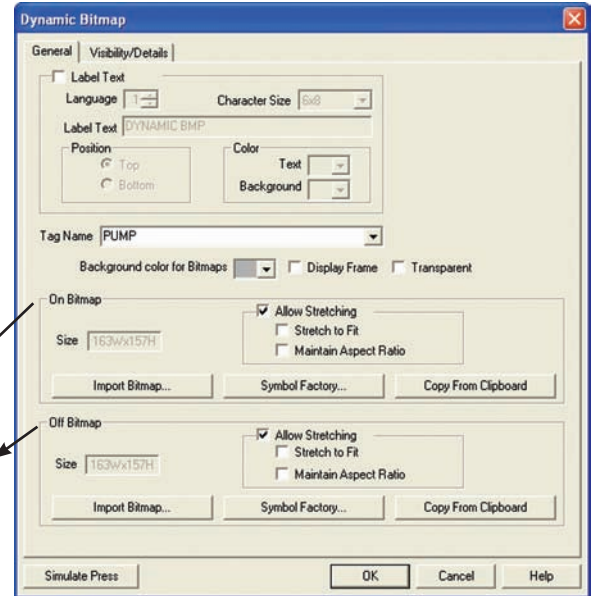
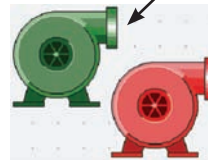
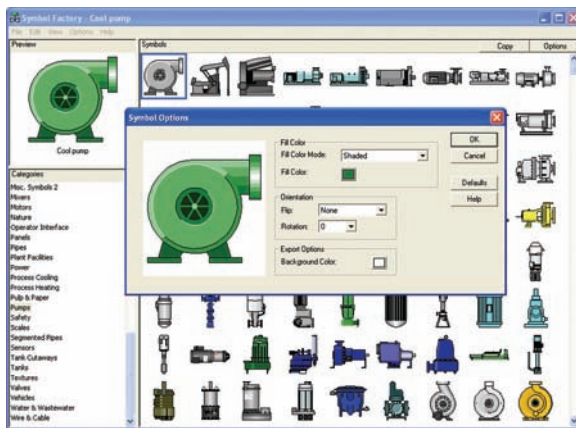
The Bitmap Button object allows the use of bitmaps for the ON/OFF states, instead of text or colors. For example, you could place a throw switch and static text labeling for on/off states. When the operator presses the switch, the bitmap is replaced with the OFF state bitmap, showing the switch down. This is a simple toggle between ON/OFF states. You can also have buttons stay ON or turn OFF only when the operator is pressing the button, as shown farther right. There is even one more option in which a button, once pressed, can only be turned ON or OFF by a non-HMI source, like a PLC.



## Dynamic Bitmap

A Dynamic Bitmap object, while not an interactive button, can show a dynamic visual representation of a process. Simply select a bitmap to represent the process running and another to represent the process not running, as the ON/OFF states of the bitmap.

For example, if your process uses a pump, and you want a visual representation of when the pump is on or off, then select the dynamic bitmap object. Click the symbol factory button from the On Bitmap section of the dialog box to access the symbol factory. After selecting the pumps category from the list on the bottom left and then the desired pump, click Options. This gives you the ability to change the object's shading, color, orientation, etc. When finished and out of the Symbol Options box, click Copy, which copies the bitmap to the clipboard. Then click Copy From Clipboard. This places the symbol into the object's On Bitmap state. Repeat these steps for the Off Bitmap, and you are done!



## Multi-state Bitmap

Similar to the Multi-state Indicator, this object displays predefined bitmaps instead of messages. Choose up to 16 different bitmaps to represent various conditions of a process. The bitmap shown is based on the value of a bit in a word or a word address. The example below shows how an operator would know the condition of the machine based on the colored bitmap which is displayed.

