How-To Guide

LCD 4x3 Code Creation Assistant

Table of Contents

Overview

The LCD 4x3 Code Assistant (**www.topazsystems.com/software/lcd4x3codeasst.exe**) assists in the creation of "Hotspots" and in the placement of bitmap images on the LCD 4x3 screen by providing a graphical user interface for the placement of these elements. The assistant provides code for these SigPlusLCD methods:

1. LCDWriteBitmap   2. KeyPadAddHotSpot

After visually placing these elements within a rectangular display representing the tablet's LCD, the code is generated to place these elements in the same locations programmatically. When starting the "Code Assistant", the "Position Window" will open.
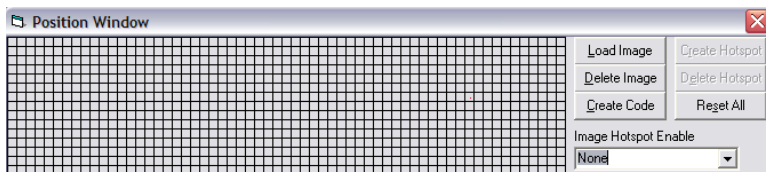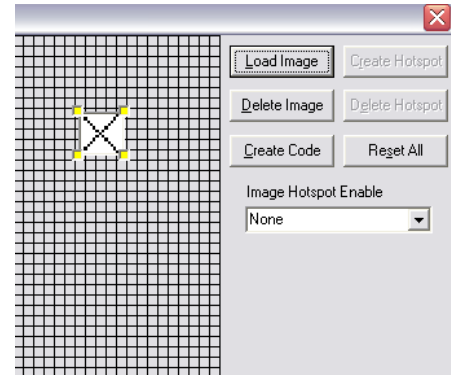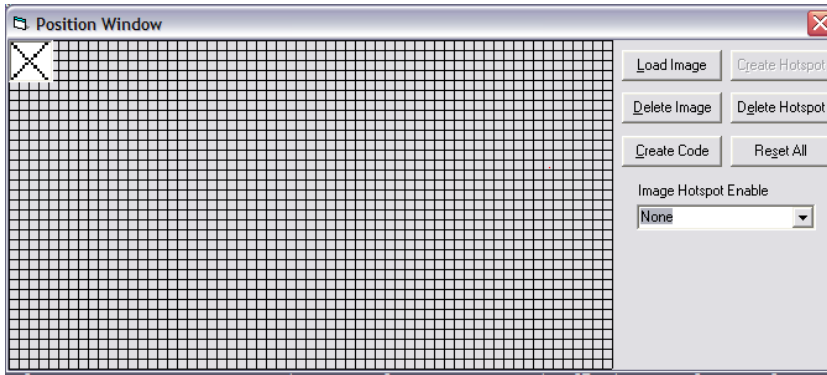

Images

From this window, the user can use the "Load Image" function by clicking the "Load Image" button. The "Load Image" function allows the user to load a black and white bitmap (.BMP) into the "Position Window". After clicking the "Load Image" button, an "Open" window will appear.
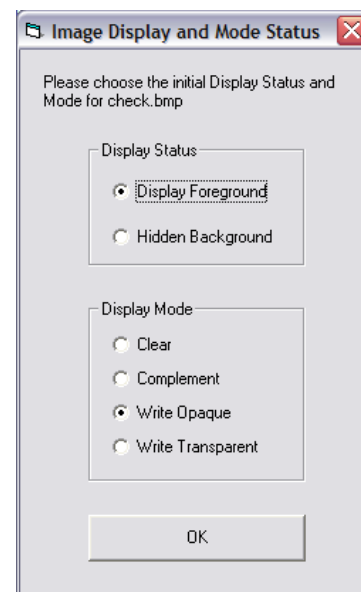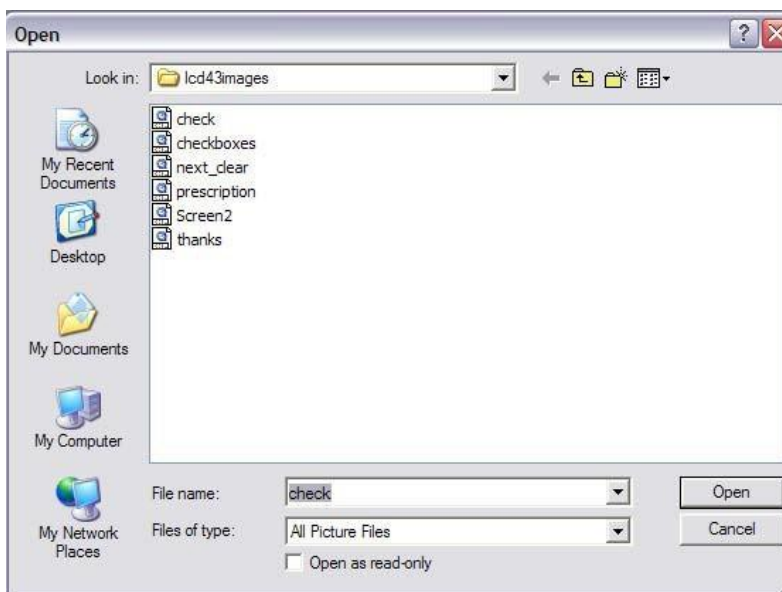
Each image placed will entail a choice of background or foreground placement, as well as the display mode (these will later correspond to the first and second arguments of each use of the LCDWriteBitmap method). These choices will appear in the "Image Display and Mode Status" window. Once the desired choices have been made, simply click "OK" to place the image.

Here is how check.bmp has been loaded into the "Position Window". Once the image is loaded, simply drag and drop to move it into the desired location. To delete an image from the "Position Window", right click on it. This will bring up four yellow boxes on the corners of the image signifying that the image has been selected. When an image displays these yellow markers, it cannot be dragged and dropped. Simply click again on the image to get rid of the yellow delete markers.

Once the image is selected, click on the "Delete Image" button to remove the image from the "Position Window". To reset the "Image Window" to its initial state, click on the "Reset All" button.
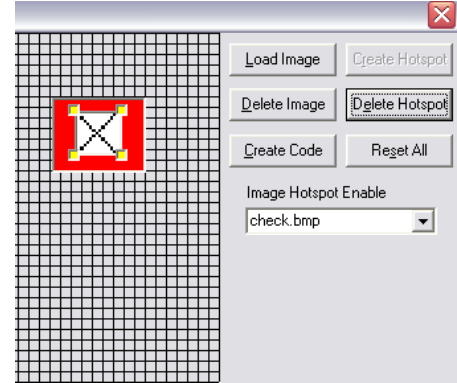
HotSpots

To outline a hotspot, click and hold in a chosen spot on the form, and drag the red-outlined box to the desired size. Then, let go of the mouse button.

To create a hotspot in this outlined area, click the "Create Hotspot" button. A red box indicates the hotspot. To delete the hotspot, click on the red box, and it will turn blue. Pressing the "Delete Hotspot" button will then delete that hotspot.

To draw a hotspot directly over an image, you must first choose that image's name from the "Image Hotspot Enable" combo box.
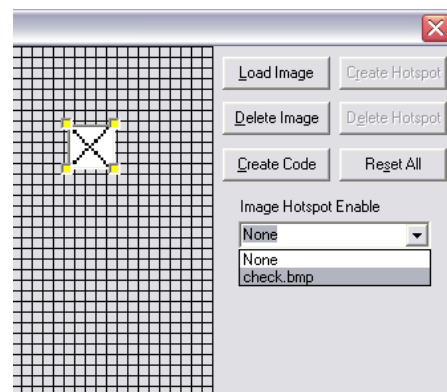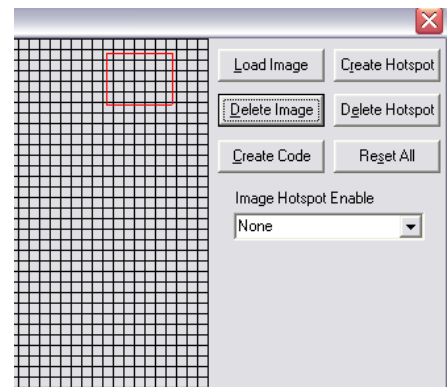
When an image is chosen for "hotspotting", you can drag the hotspot outline directly on the image, and press the "Create Hotspot" button to create a hotspot there. The hotspot will be created behind the image, so if you want to delete it later you will have to move the image out of the way.

While an image is highlighted in the combo box, it cannot be dragged and dropped. A choice of "None" in the combo box will mean all images can be dragged and dropped, and none are set for "hotspotting". Once again, the "Reset All" button will reset the entire LCD grid, allowing the user to start over.
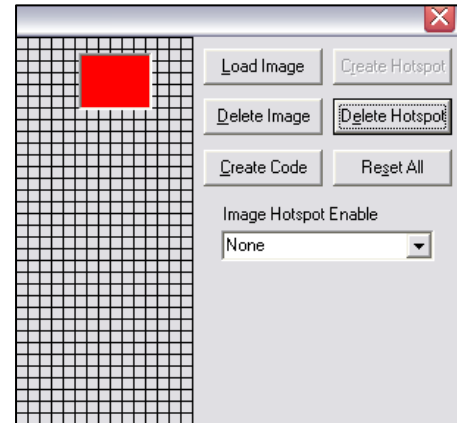
Creating Code

When you're satisfied with the layout, press the "Create Code" button, and the LCDWriteBitmap and KeyPadAddHotspot code (for each image and hotspot created using the program) will be generated, including all the necessary arguments. The "Return to Form" button will take you back to the LCD grid.
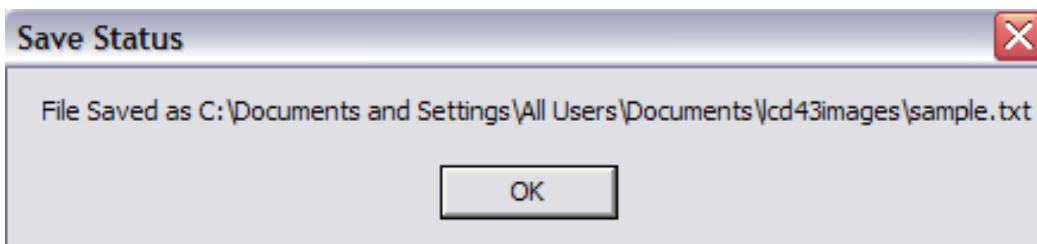
You may write this code to a text file with the "Write To Text File" button. Then, you can copy and add to a later programming endeavor if you wish.
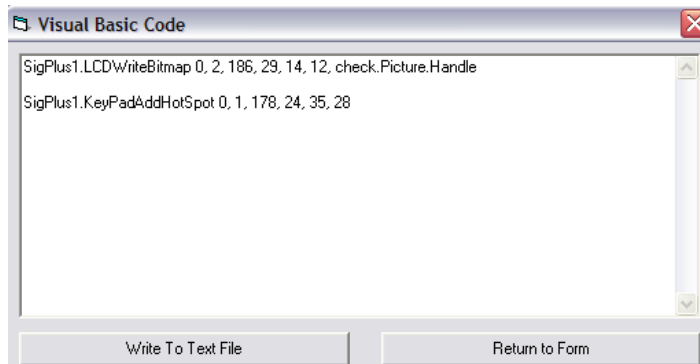
Clicking "Write To Text File" will open a "Save As" window, where the user names the .txt file where the code will be saved.
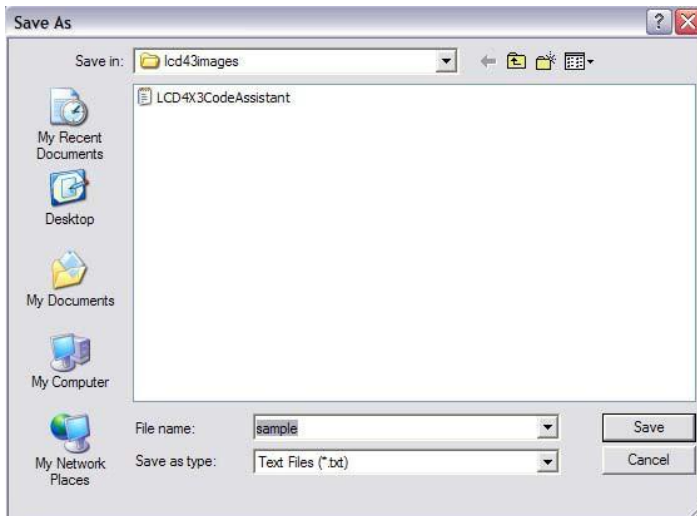
When the code has successfully been saved, a confirmation message will be generated.

You may add up to 10 images and 10 hotspots per session. If you reach this limit, you'll need to create the code, save it, and start again to add more.

Finally, the LCDWriteBitmap method requires the use of each bitmap's handle. The handle created with this program is based upon the assumption that the image or picture object that will eventually hold the image will be named the same as the image itself. If you wish to name the object differently, you will need to adjust that part of the LCDWriteImage call accordingly.