How To Build ILI9325 TFT LCD Component for Proteus VSM

Vahid Heidari

June 10, 2024

1 Introduction

`Proteus VSM` is an electronic circuit simulator best known for its reach digital component library. It comes with a CAD program, named `ARES`, that allows you to design PCB boards based on schematics.

Unfortunately, `Proteus` does not have any `ILI9325` TFT LCD component. The only way to simulate this component is to provide a custom-made library. In this project, thanks to `Proteus VSM` C++ SDK, I wrote an `ILI9325` library.

In this article, I explain the process of designing a 2D graphical representation, building a `.DLL` module for handling active digital model, and installing this user build component to `Proteus VSM` for simulation.

2 Designing 2D Graphical Representation

Select `2D Graphics Box Mode` tool from the left side tool bar and draw a rectangle as a base of the LCD panel.
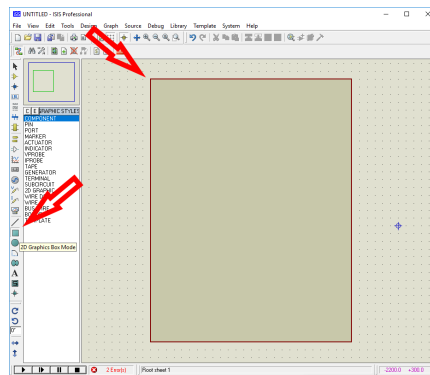
Then select `Device Pin Mode` and put a pin (orient the pin if needed). Right click on the pin and select `Edit Properties` from the drop down menu to open
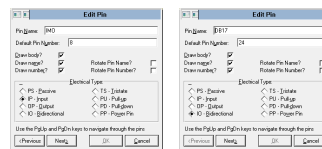
the `Edit Pin` dialog. Type the name of the and number of the pin, and check
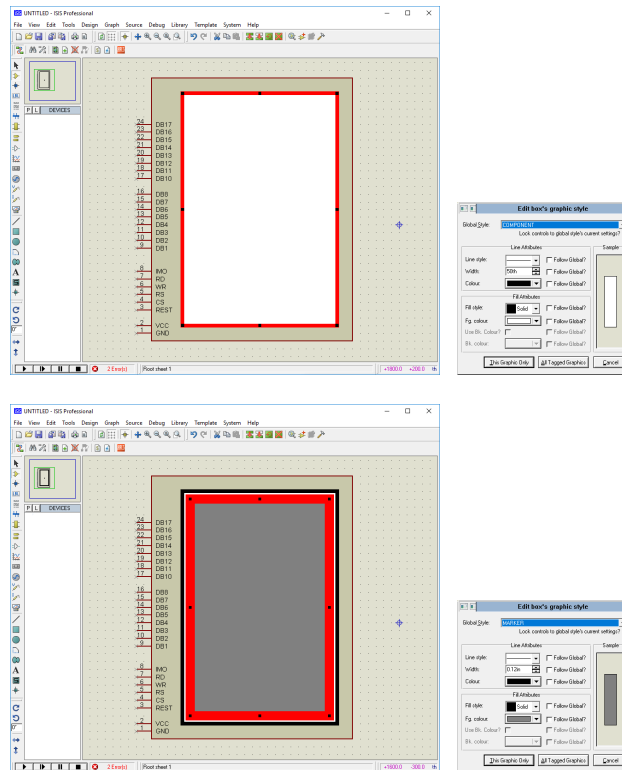the `IP - Input` option.
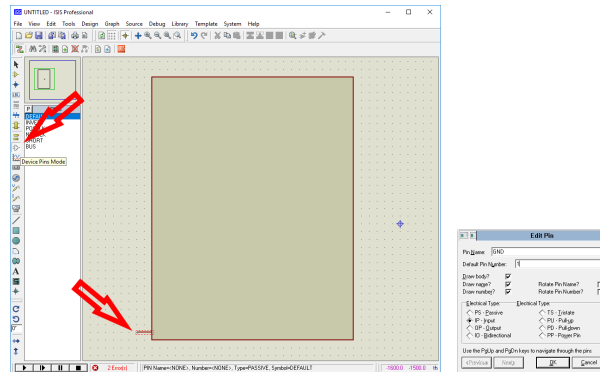

Repeat the previous steps for all other pins.
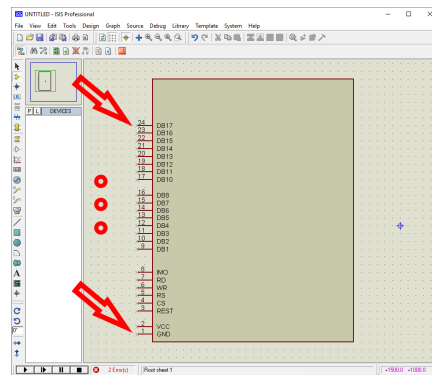
Check `IO - Bidirectional` option for DB1 to DB17 as shown in the picture.

Go back to 2D Graphics Box Mode and draw another rectangle for the bezel frame and change its color and border line as shown in the picture.

Draw the TFT glass on top of bezel frame. This rectangle is the actual drawing area that is filled with pixel colors in code.
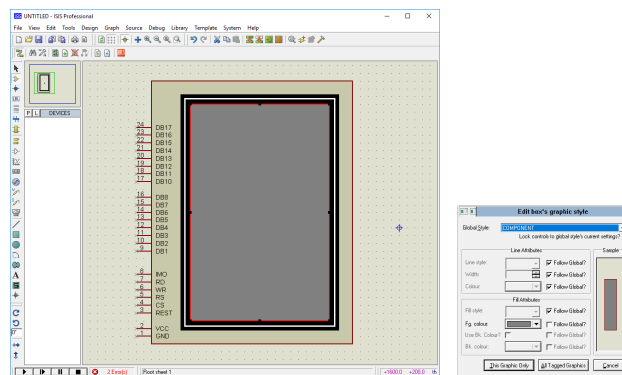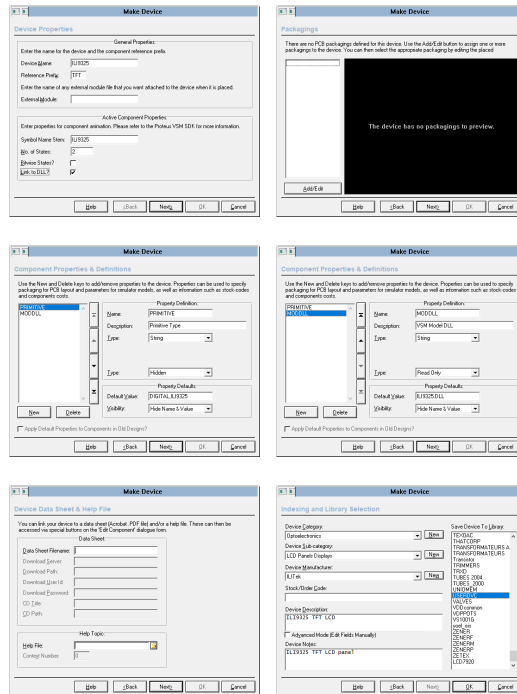

Select 2D Graphics Markers Mode and place a ORGIN marker in the center of the TFT glass.

Now our graphical representation of the LCD is completed.

Select the whole elements and right click. From the drop down menu select Make Device.

Fill the text-boxes as shown in the picture and click `Next` button.

In the last dialog, click OK to add the `ILI9325` device to the `Proteus` library. You can select the device by searching from the `Pick Devices` menu and place it in design sheet.

3   Making Symbols

`Proteus` uses symbols to animate active components. In this project we make two symbols for `ILI9325` LCD panel. Select `ILI9325` component form `Pick Devices` menu.
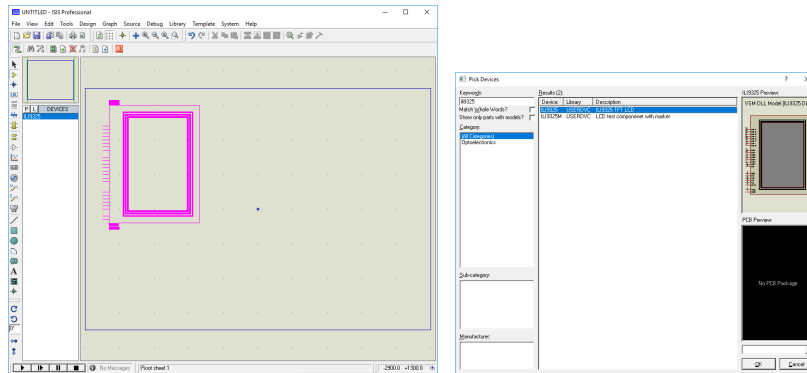

Place a LCD on the design sheet. Right click on the component and select the `Decompose` option from the menu.
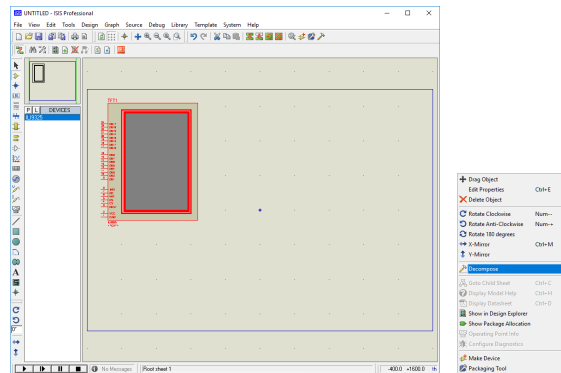
Delete all part of the LCD except TFT glass and `ORIGIN` marker, then block copy another instance.
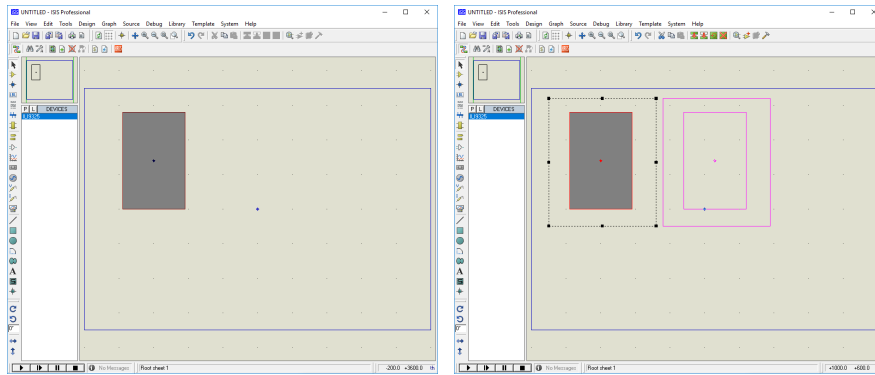

Select the first rectangle and change its color by right clicking on it and selecting `Edit Properties`. Uncheck the `follow Global?` check box and change the `Fg.   colour` option from the color picker menu, then click on `This Graphic Only` button.
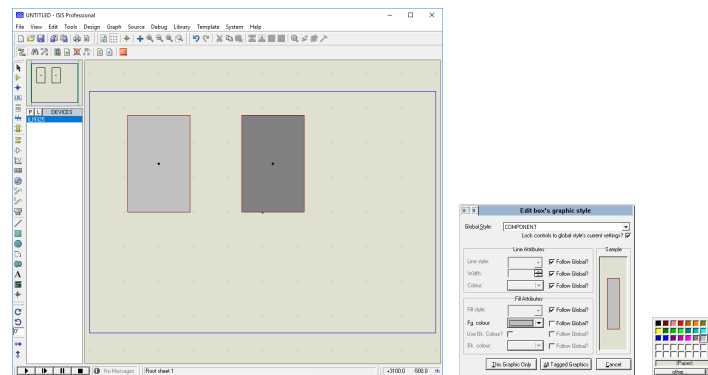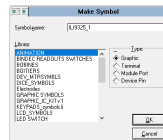
Do the same steps for the next rectangle but change its color to white.
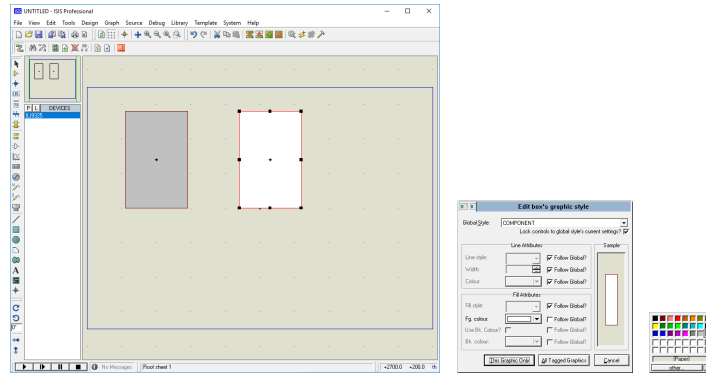
Select the first rectangle and `ORIGIN` marker and right click. From the menu, select `Make Symbol` option. Take care that the symbol name MUST follow the `Symbol Name Stem` pattern from the `Make Device` dialog.
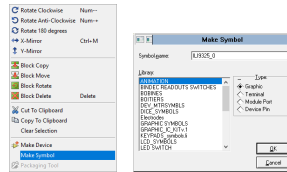
For the second symbol do the same steps and type `Symbol name` as follows.

You can check the addition of symbols by searching in `Pick Symbols` form the `2D Graphics Symbols Mode` on side panel.

4   Building `.DLL` Module

Making `.DLL` module is strait forward. You can follow instructions from `Proteus VSM` SDK documentation [2], section VSM MODELLING TUTORIAL. I also found a useful Github repository [1]. The only note you should take care of is you MUST make a 32-bit `.DLL` because `Proteus` does not recognize 64-bit .DLLs.

References

[1] cdhigh. Github — LCD12864 with controller ST7920 Proteus vsm model. https://github.com/cdhigh/lcd12864_st7920_proteus, Oct 3, 2021. [Online; accessed 04-June-2024].

[2] Labcenter Electronics. Proteus VSM SDK — fractal gecif. http://fractale.gecif.net/si/logiciels/proteus/ProteusVSMSDK.pdf, Oct 30, 2000. [Online; accessed 04-June-2024].