GTT Designer
Screen Tracking Example

A GTT Designer example for keypad input

Application Note

Revision 1.0

Introduction

With the simple and intuitive GTT Designer, multiple screens can be made, and implemented into your project quickly and easily. When transitioning between multiple screens, it is important to keep track of which screen you are currently on so that the display is properly updated when a user interacts with it. This example demonstrates a simple method of keeping track of which screen you are currently on, and which region has been pressed on that screen.

Connections

To design this example, a GTT43A-TPR-BLS-B0-H1-C4-V5 was connected to a 5V power source via its power/communication header, and to a PC through its Mass Storage USB header with an external USB cable (EXTMUSB3FT). To provide control, an Arduino Uno was connected to the I2C header of the GTT43A with a BreadBoard Cable (BBC), and to a PC via a USB cable.
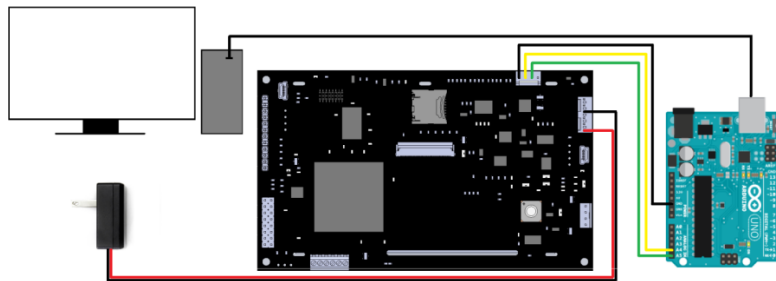

Figure 1: Screen Tracking Example PC, Arduino, and Power Connections

Design

Due to the nature of this example, multiple screen designs are used. Each screen design follows one of two templates; a menu template, and a product information screen.

The menu template is comprised of two arrow buttons to navigate between products, a reset button to restart the GTT, a button to report what screen the display is currently showing, and a central image button displaying one of our Matrix Orbital GTT products. A user can navigate between products by pressing the left or right arrows. Furthermore, pressing the central image button will bring the user to the specific product's information page.
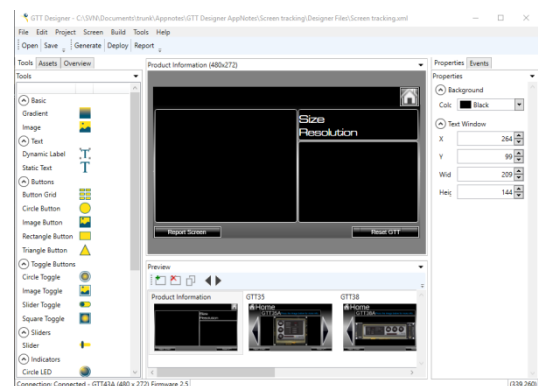

Figure 2: GTT Designer Product Information Design

Upon startup, users will start on the GTT43A product screen. Navigation between different GTT series products is done by pressing the arrow buttons. When a user selects a product, they will be directed to the product's information screen. When a product information screen is brought up, the name of the product selected, as well as the size, and the resolution of said product will be updated. A larger image of the product will be drawn, and a description of the product will be updated as well. Located in the top

right hand corner is a home button that can be used to return the user to the main menu. A restart button is also offered if needed.

Figure 3:Screen Tracking Main Menu design

GTT scripts have been implemented in the project to remove some of the processing load from the microcontroller, and speed up the project overall. Any commands listed in the GTT protocol manual can be included in a script, allowing any of the GTT's settings, or any of the dynamic assets on the screen to be changed. Scripts can also be linked to assets on screen, and will run when a user interacts with the asset. Furthermore, scripts can be linked to screens, and can run before or after said screen is loaded.

In this project, scripts are used to transition from screen to screen, and are also used to identify what screen the GTT is currently displaying. In addition, scripts are used to adjust screen brightness, indicating the GTT's loading status. Prior to every screen transition, a script will run, letting the microcontroller know what screen is currently being displayed, and what screen the GTT will be transitioning to. You can view the GTT Designer events tab to determine which scripts are linked to an asset, or to a screen.

Code

The code for this example was written in Arduino C. To start, a number of global variables are defined for the program. Note that the I2C address in code must match that set in the design.

Upon startup, the Arduino will establish communication to the PC, and update the Arduino Serial Monitor, notifying the user that the program has initiated. In addition, I2C communication between the Arduino and GTT begins. Upon startup, the GTT will display the GTT43A home menu screen, and wait for input.

Figure 4: Screen Tracking Code

Once the initial conditions of the program are set, the code runs in a loop that continuously reads input from the GTT and handles specific touch regions that are defined in the design. When a recognized region is received from the GTT it is processed.

Whenever a region is pressed, the GTT will echo an ASCII character representing the current screen that is being displayed, as well as return the ID of the region that was pressed. Furthermore, if the GTT transitions to a new screen, it will return the ASCII character of the new screen. Using this information,

the Arduino Serial Monitor will specify which screen it is currently on, which region has been pressed, and which screen it has transitioned to.

The main function of the program is to update the GTT appropriately, based on which screen it is currently on. Whenever the user navigates between screens, the GTT will alert the user of which screen it is currently on through the Arduino Serial Monitor, and the program will properly update the GTT with new product information, based on said screen.
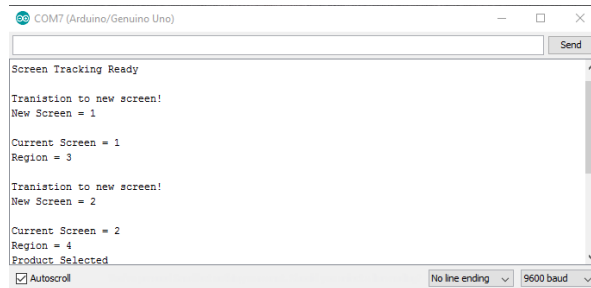

Figure 5: Arduino Serial Monitor responses

Conclusion

This simple code demonstrates the elegance that the GTT display and designer software can bring to even the most basic controllers.

For this example, GTT Designer was used to generate and deploy each screen designs to the GTT43A. Using the information available in the Designer report, a simple piece of code was written for an Arduino platform to control the interface.

Figure 6: GTT43A running the Screen Tracking Demo

This Screen Tracking example demonstrates the capabilities of both the Matrix Orbital GTT display series and the GTT Designer software.

For more demos and tutorials on the GTT and the GTT Designer, check out our forums at lcdforums.com

Contact

Sales
Phone: 403.229.2737
Email: sales@matrixorbital.ca

Support
Phone: 403.204.3750
Email: support@matrixorbital.ca

Online
Purchasing: www.matrixorbital.com
Support: www.matrixorbital.ca