

Advanced Direct Drive LCD

Description: Using the Renesas API to create real world LCD products

Objectives

1. Place Bitmaps on a TFT-LCD panel
2. Generate color tables and backgrounds
3. Add and manipulate icons
4. Implement a touch-screen based system and show how to create a menu based application.

Lab Materials:

Please verify you have the following materials at your lab station.

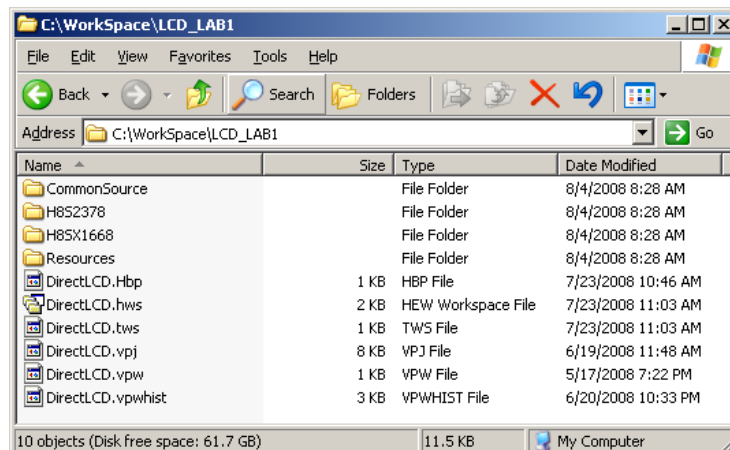
- Laptop
- E10a Debugger
- Renesas RLCD Kit and Kyocera LCD Panel
- HEW Version 4.04.01
- H8S Tools Version 6.2.0.0

Skill Level : Intermediate to Advanced knowledge of C programming. Intermediate knowledge of Renesas MCU's and Basic understanding of LCD Displays

Time to Complete Lab: 1hr 20 minutes

Lab Sections

- | | | |
|----------|--|-----------------------------------|
| 1 | Putting Custom Graphics on the LCD | Time to complete task: 15 minutes |
| 2 | Presenting Multiple Bitmaps and Animation | Time to complete task: 15 minutes |
| 3 | Icon Manipulation and the Touch-screen | Time to complete task: 20 minutes |
| 4 | Making a Complete Application | Time to complete task: 30 minutes |




1 Putting Custom Graphics on the LCD

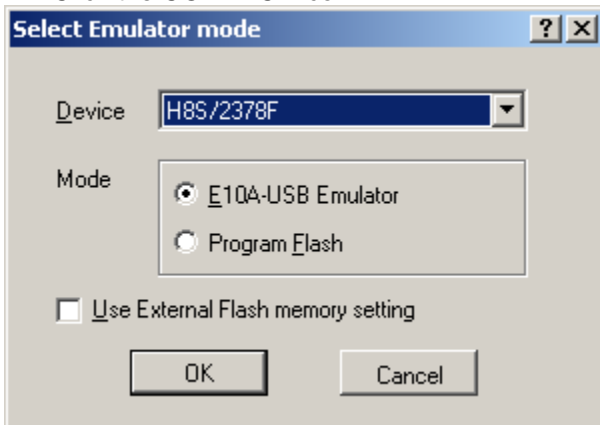
Time to complete task: 15 minutes


Overview:

In this section we will build our first working project and add a background and picture to the LCD panel

Procedural Steps


1. Open HEW by double clicking the icon on the desktop
2. Open the following workspace at startup – you may need to use the “Browse to another workspace option”
C:\WorkSpace\LCD_LAB1\DirectLCD.hws
3. Check your Demo board has power, and that the E10a is connected to the board and the PC.
4. Click the CONNECT icon 



5. Make sure the Use External Flash memory box is Unchecked (as above), click ok
6. When prompted Reset the demo board using the switch next to the RS232 connector and “OK” your way through the following screens choosing 16MHz as the clock option - until the device is connected
7. Click the BUILD ALL icon 



Your code should build with No Errors – if your code has errors, please ask for assistance.

8. You can now  RESET and RUN the code – Note: There will be a delay of a few seconds as picture images are written to the internal flash.
9. Nothing should happen – but it should happen smoothly! Press STOP
10. Open the file: DemoTasks.c and find the comments to INITIALIZE the LCD
11. Remove the comments so that the initialization routine is called.
12. Find the comments to turn on the LCDBacklight – look in the API Handout for a value to pass.
13. Rebuild your project and Run your code. What Happens?
14. Open the file Demolnit.c and using the Handout Guide and the LCDBMPFill command; add the background where the comments indicate.
15. Rebuild and Run to check your code works. What Happens?

BONUS POINTS(1): Use HEW to view your embedded Bitmap as a graphic!

VIEW>GRAPHIC>IMAGE will allow you access to graphics residing in memory.

They are stored in 8bit color RGB format

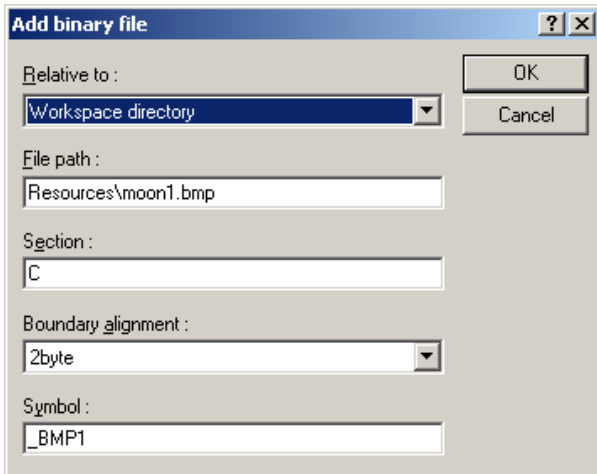
Indexes to the raster image and palette can be found by looking in memory

(VIEW>CPU>MEMORY) and referencing the BIG ENDIAN words to the Bitmap Secret Decoder

Ring

e.g. _BMP2+123

Close the Workspace once you are done....



7. Test
8. Open the file DemoTasks.c and using the Handout Guide add a LCDBMPCopy command which will put the currently indexed Bitmap in the Frame Buffer. This line of code finishes a loop which will scroll through the images one by one, creating our animation.

BONUS POINTS(2a): Move the bitmaps to a different location on the screen.

BONUS POINTS(2b): Change the "Tick_Rate" to speed up and slow down the animation

Close the Workspace once you are done....

3
Icon Manipulation and the Touch-screen

Time to complete task: 20 minutes

Overview:

In this section we will add icons to the screen and explore the touch-screen

Procedural Steps

1. Open the following workspace:
 C:\WorkSpace\LCD_LAB3\DirectLCD.hws
2. Connect to your target via the E10a and Build the Project
3. Add the Initialize and Repaint Icons calls to DemoTasks.c
 Look in icons.c to find the function names.
4. Build and Run to test the icons are on screen.
5. Stop the demo.
6. Open Touchscreen.c and find the x and y coordinate variables used to store touch position.
7. 'Watch' these variables (right click on a variable and choose Add Watch) and use the **R** button to make these a realtime watch.



Due to these variables being locals – you will need to run to a breakpoint somewhere in the function in order to select the real-time watch feature.

8. Run the code – and watch the variables change – what do you notice if you keep your finger/pointer in the same spot?

BONUS POINTS(3): Alter the brightness of the icons

Close the Workspace once you are done....

4
Making a Complete Application

Time to complete task: 30 minutes

Overview:

In this section you will build a final application:

The requirements are that it should:

- ✓ Initialize the first bitmap and icons with a central PLAY button.
- ✓ Scrolls through the bitmaps when the arrows are pressed.
- ✓ Animate the bitmaps and replace the PLAY with a PAUSE icon when PLAY is pressed
- ✓ Returns to the initial state when the PAUSE button is pressed.

Procedural Steps

1. Open the following workspace:
 C:\WorkSpace\LCD_LAB4\DirectLCD.hws
2. Connect to your target via the E10a and Build the Project
3. Add/modify the code to create the application above, or use the time to modify it to be more suited to your next application.
4. Help is on hand as needed.