This site uses cookies from Google to deliver its services, to personalize ads and to analyze traffic. Information about your use of this site is shared with Google. By using this site, you agree to its use of cookies. LEARN MORE GOT IT

Geek 101 : Introduction to Geekiness

Learn about electronics, microcontrollers, sensors and much more!

Saturday, July 5, 2014

Intermediate: QC12864B (128x64 LCD using u8glib.h)

Hi, everyone. Today, I have some time to kill. So, I decided to play around with my 128x64 pixels LCD. This is also a product distributed by GJ Electronic which can be found here.

There are some sample code written in C attached on the product page, but for me, it's not very helpful as I'm more familiar with Arduino. So, this tutorial will be about interfacing Arduino and the LCD.

Introduction

This LCD comes with build-in ST7920 controller chip which will handle the data sent to it. The attached user manual somewhat tells you how it should work and what command you should send, but it is still blurry, isn't it? It's just look too complicated. But, there's an easy solution for that. An external library makes our life so much easier. The library is called **u8glib** and it supports wide range of LCD. However, this tutorial will focus on this LCD only. You can see the list of supported LCD here.

Usually, you need to send 8 bits of data to make the LCD display what you want it to display. So, basically, there'll be 8 wires going from Arduino to the LCD. Imagine how complicated that would look like. However, there's the sweetest part of all. The library allows you to program the LCD using 3 methods:

1. Using 8 bits of data as mentioned above. (Parallel Bus Mode)
2. Using hardware/built-in SPI port on Arduino. (Serial Bus Mode)
   3. Using software SPI port.

Personally, I'd choose the second method in case where the SPI ports are available. It uses less port, making the connection far simpler and also leave the other ports available for other applications. For your information, SPI ports are on pin 10 (Slave Select), 11 (MOSI), 12 (MISO), 13 (SCK) on Arduino Nano and Arduino Uno. Using the software SPI port may slow down the refresh rate.

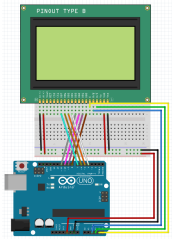
Connection

This product comes with 20x1 header, which really helpful to shove the LCD into a breadboard. I'll show you how to connect using the first and second methods as mentioned above. The third one is almost the same as the second one, but it is actually depends on what port you want to use.

Note that the pin in the following photo is equivalent as follows:

D/I = RS
CS1 = PSB
CS2 = NC

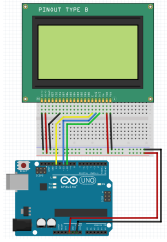
First method:



Second method: (Special thanks to this website for showing this method)

IMPORTANT! For this LCD to work using this method, you need to de-solder R9 and R11 at the back panel of the LCD.

The reason is, VDD is actually connected to PSB through the SD resistors (R9 and R11), making PSB always HIGH, putting the LCD always in parallel bus mode. Connecting GND to this could end badly to your LCD and Arduino as you are basically connecting 5V to GND directly. So, just de-solder it. Don't worry, you can still connect it in parallel bus mode by connecting PSB to 5V.



Testing the LCD

To test the LCD, simply copy the **U8glib** library/folder into Documents > Arduino > libraries.
Start Arduino IDE. Goto File > Examples > U8glib > GraphicsTest.

The next step depends on the method that you choose.

If you choose the first method, un-comment the line:

```
U8GLIB_ST7920_128064 6X u8g(8, 9, 10, 11, 4, 5, 6, 7, 18, 17, 16);  // KB4A Conn: D0..D7
```

If you choose the second method, add the following line:

```
U8GLIB_ST7920_128064_6X u8g(10);
```

If you choose the third method, modify the following line as you needed to:

```
U8GLIB_ST7920_128064_6X u8g(18, 16, 17); // SPI Conn: SCK = en = 18, MOSI = rw = 16, CS =
```

If you choose the first method but using different pin numbers, make sure you also change the pin numbers in the declaration above.

If the LCD is showing various shapes and text, that means your LCD is working correctly. Now, you may go on and program your own graphic or anything that you want on the LCD. You may refer to more functions and explanation about the U8glib library on their official page and user reference page. Check out this page as well. There are links to the explanation of the examples on that page.

I've shown you the basics, so you should be able to explore more by yourselves. Just comment down here if you do not understand any step. Until next time, good bye. Have fun!

Posted by Ahmad Shahril at 3:11 AM +1 Recommend this on Google

Labels: Arduino , Intermediate , LCD , SPI , Tutorial , u8glib

Search This Blog

Search

About Me

Ahmad Shahril

Follow

I'm a unique (e.g., unique enough not to be known by anyone ...)

View my complete profile

Popular Posts

- Intermediate: 4-Digit Mini 4A GPS Module
  U8GBATC : I doubt another tutorial on this module which add-on library that you might encounter after trying out the method below. It ...
- Intermediate: QC12864B (128x64 LCD using u8glib.h)
  Hi, everyone. Today, I have some time to kill. So, I decided to play around with my 128x64 pixels LCD. This is also a product distributed b...
- Intermediate: u-Blox Neo-6M GPS Module
  Previously, I've posted about this module which you can find here . This time around I'm going to share with you another library th...
- Basic: HMC5883L (Digital Compass)
  Hi, everyone. I haven't been watching the time for breakfast. I'm writing this post to take my mind off the news. So, this time, start ...
- Basic: HC-06 Bluetooth Module
  Hi. It has been a while since my last post. I've been busy with the Eid preparation, getting my driving license, babysitting, etc. Finally I a ...
- Intermediate: Getting Started with Firmata on PC
  Hi, everyone. It's been soooo long since I last posted a new blog. Mostly because I'm busy and lazy. :D Anyway, I'm in a d ...
- Upcoming Post: LCD Code
  Hi! It has been too long since I last posted a new blog. I'm currently working on multiple projects and really busy. This is ...
- Basic: Getting Started (Arduino)
  Well, actually, I don't have time for this kind of post. There's a lot of information in getting started with Arduino already...

Blog Archive

- 2015 (2)
- 2014 (6)
  - September (2)
  - July (2)
    - Basic: HMC5883L (Digital Compass)
    - Intermediate: QC12864B (128x64 LCD using u8glib.h)
  - June (2)

Labels

Arduino (7) Intermediate (4) Basic (3) GPS (2) I2C (2) LCD (2) HMC5883L (1) Firmata (1) Keypad (1) Bluetooth (1) HC-06 (1) HC05 (1) SD (1) Green Energy (1) TinyGPS (1) Digital Compass (1) u8glib (1)

open in browser PRO version Are you a developer? Try out the HTML to PDF API pdfcrowd.com

3 comments Google+

Add a comment

Top comments

Alex Rezzik 2 months ago · Shared publicly

Thank you for good tutorial Checked - All 3 methods works for me

+1 · Reply

Ahmad Shahril (Shah) 1 year ago · Shared publicly

Interwebzler: 32128x96 (128x64 LCD using ssd1306 ic)

Hi, everyone. Today, I have some time to kill. So, I decided to play around with my 128x64 pixels LCD.
This is also a product distributed by GI Electronic which can be found here . There are some sample code written in C attached on the product page, but ..

+1 · Reply

Adriana Duarte 9 months ago · Shared publicly

Thank you very much! It was very helpful, I was having trouble connecting with SPI.

+1 · Reply

Newer Post Home Older Post

Subscribe to: Post Comments (Atom)

Google+ Followers

@Ahmad Shahril (Shah)

Add to circles

71 have me in circles View all

© Geekstuff 2014. Template images by neon-polonium. Powered by Blogger.

open in browser PRO version Are you a developer? Try out the HTML to PDF API pdfcrowd.com