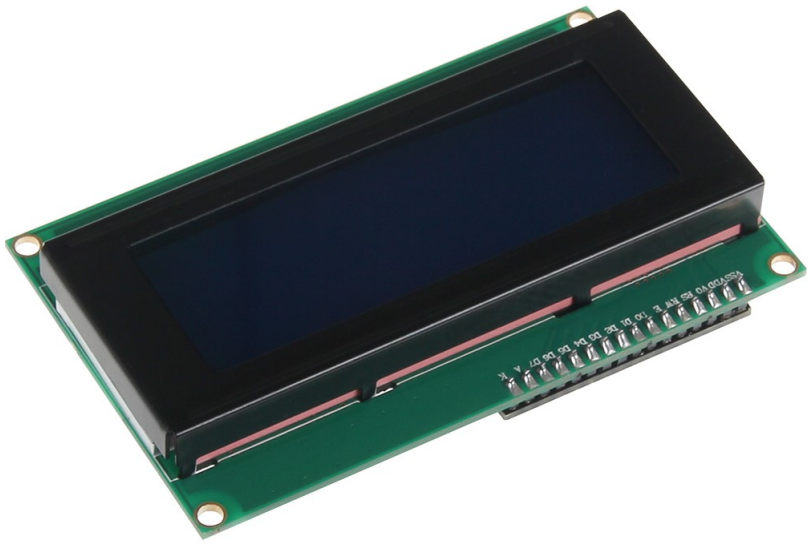
I2C Serial 20x4 2004 LCD Module



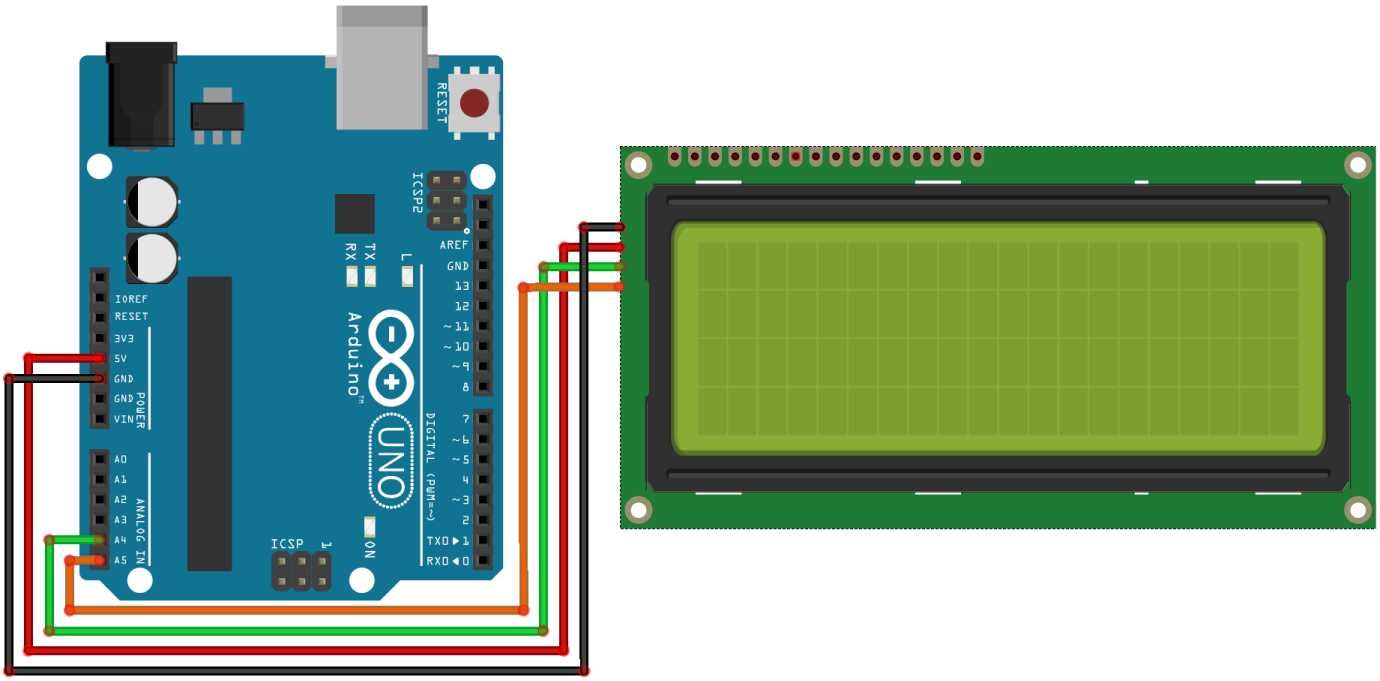
Dear customer,

Thank you for purchasing our product.
Please find our instructions below.

1. Using with an Arduino

1.1 Connecting the display

Connect the display, as seen in the following image and table, to the pins of your Arduino.

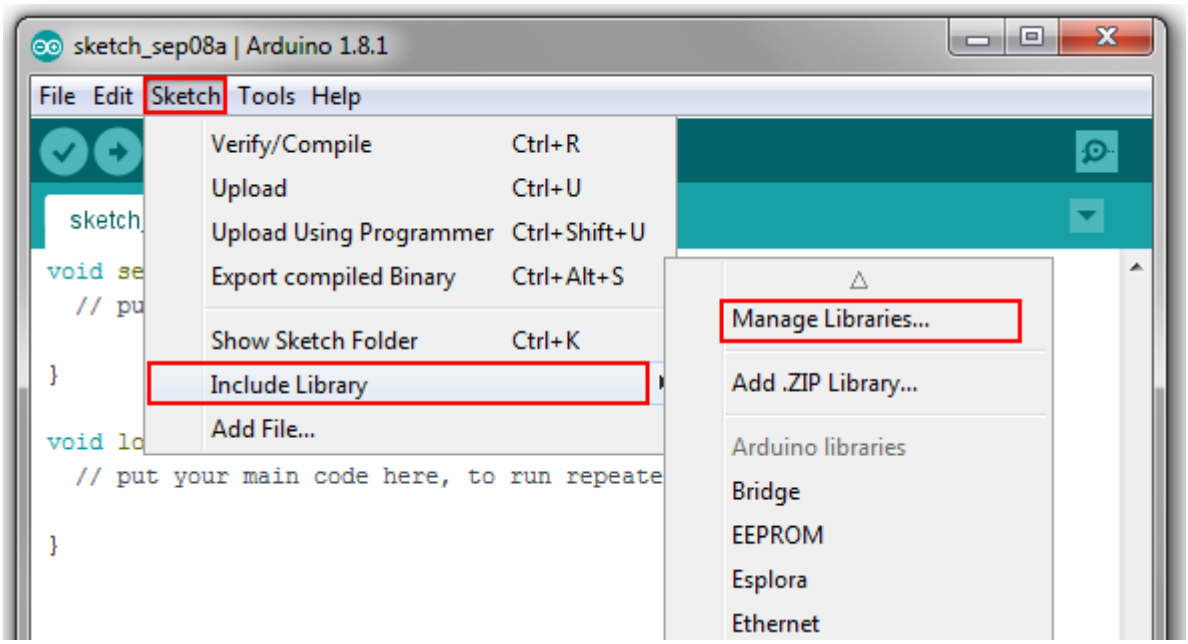


Arduino Uno	I2C 20x4 LCD
GND	GND
+5V	VCC
SDA	SDA
SCL	SCL

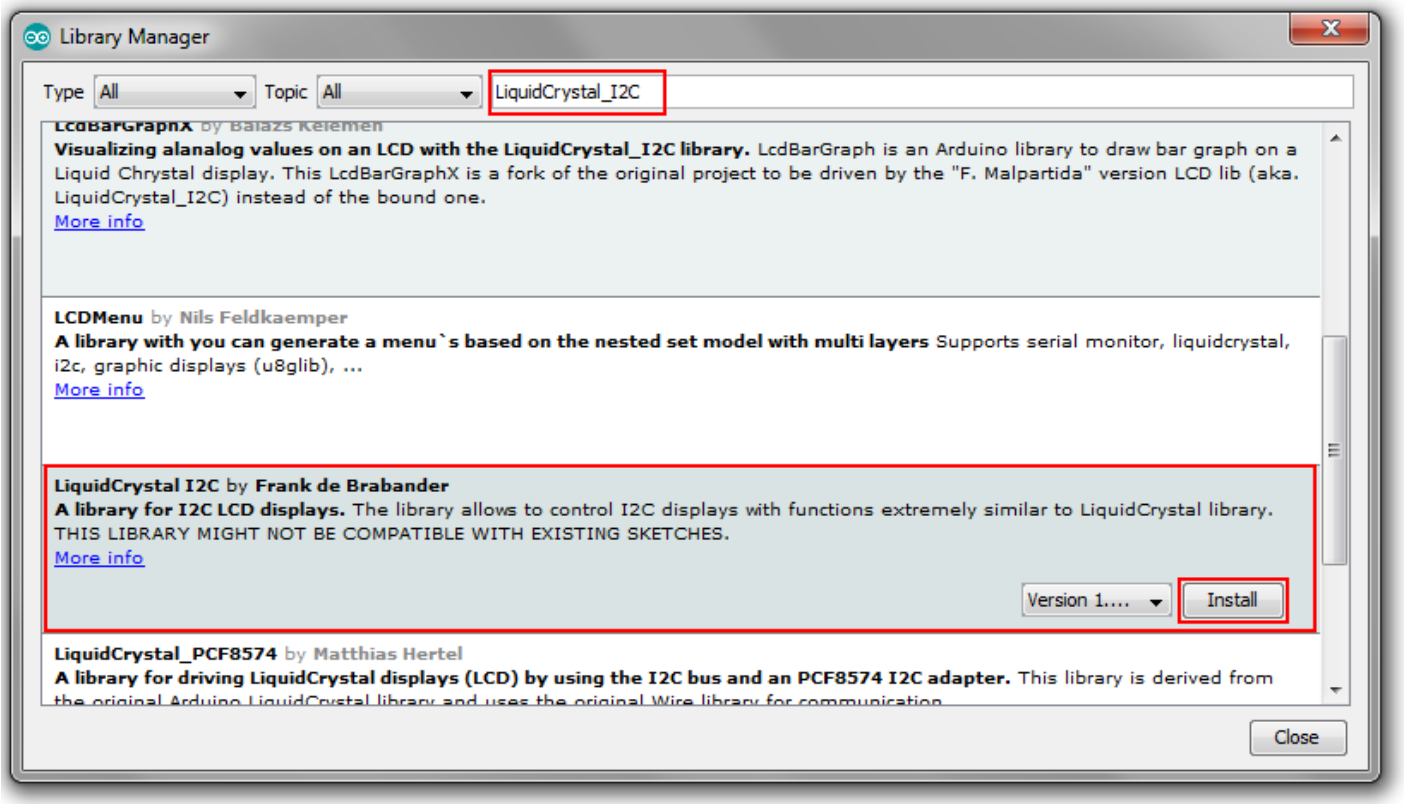
1.2 Installing the library

Before you can transfer the code-example to your Arduino, you need to install the additional **LiquidCrystal_I2C** library.
This library allows you to use your display as quick and easy as possible.

Therefore, open up your Arduinos Library-Manager.



In the now opened library manager, search for the **LiquidCrystal_I2C** library and install it.



1.3 Example-Code

To test your display, please copy the following example to your Arduino.

The displays may vary in their hardware-address.
Because of that, the right address needs to be set in the code.
If your display might not work, change the specific address in the code to the second address.

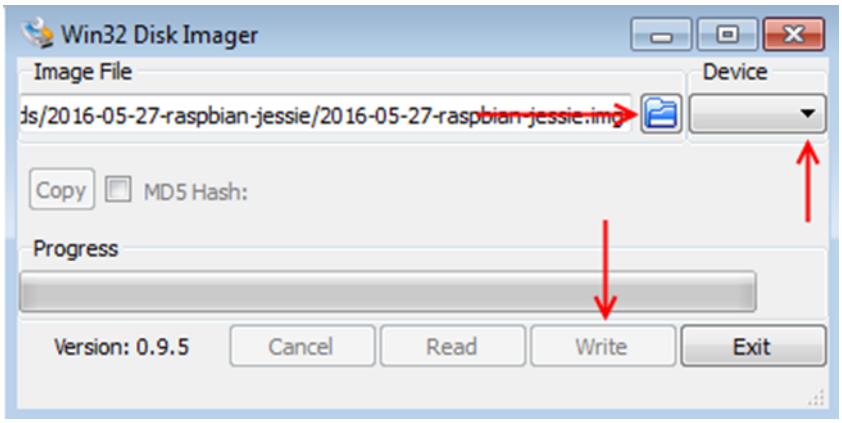
```
#include <Wire.h>
#include <LiquidCrystal_I2C.h>

//-----Hardware Addressing-----
//If your display does not work, comment the line below and uncomment
//the other address-line
LiquidCrystal_I2C lcd(0x27,20,4);
//LiquidCrystal_I2C lcd(0x3F,20,4);


void setup()
{
  lcd.init();
}

void loop()
{
  lcd.backlight();

  //Print message
  lcd.setCursor(0,0);
  lcd.print("    joy-IT");
  lcd.setCursor(0,1);
  lcd.print(" ");
  lcd.setCursor(0,2);
  lcd.print("     I2C Serial");
  lcd.setCursor(0,3);
  lcd.print("        LCD");

}
```

2. Using with a Raspberry Pi

2.1 Installing the software

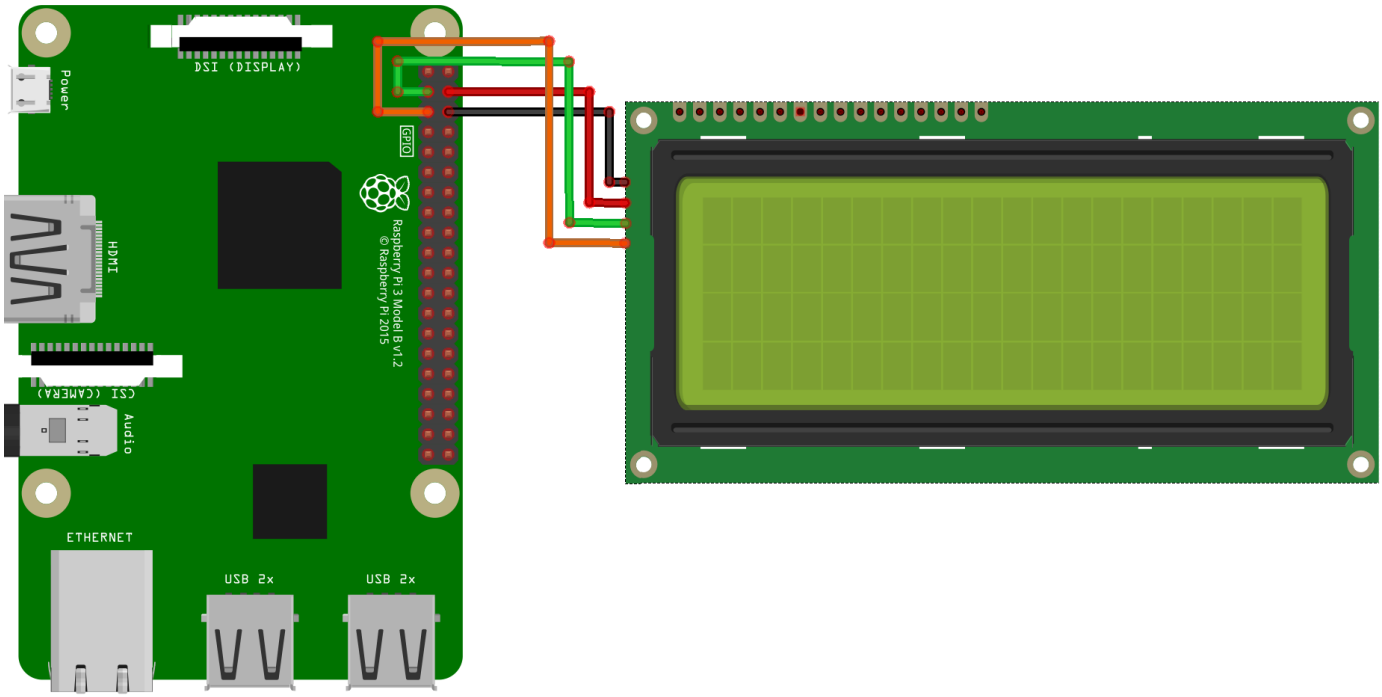
You can skip this step and continue with step 2.3 if you are already using the latest Raspian-System on your Raspberry Pi.

With the help of the „Win32 Disk Imager“-Program, install the latest Raspbian image which you can download here.



2.2 Connecting the display

Connect the display, as seen in the following image and table, to the pins of your Raspberry Pi.



Raspberry Pi	I2C 20x4 LCD
Pin 6 (Ground)	GND
Pin 4 (5V)	VCC
Pin 3 (BCM 2 / SDA)	SDA
Pin 5 (BCM 3 / SCL)	SCL

2.3 Installing the libraries

When you finished the Raspbian-Installation and started your system, open up the Terminal-Console and enter the following commands:

```
sudo apt-get install python-pip python-dev build-essential
sudo pip install RPi.GPIO
```

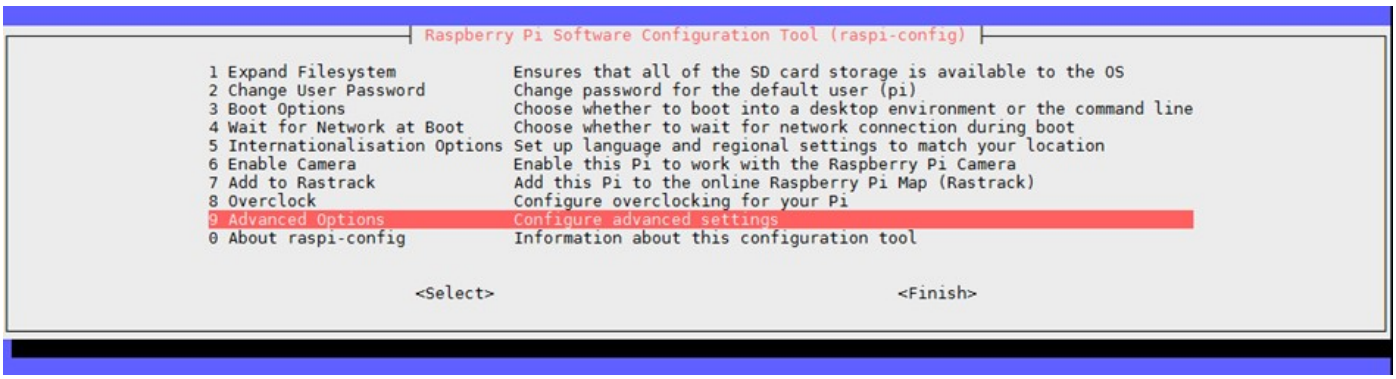
```
sudo apt-get install python-imaging
```

```
sudo apt-get install python-smbus i2c-tools
```

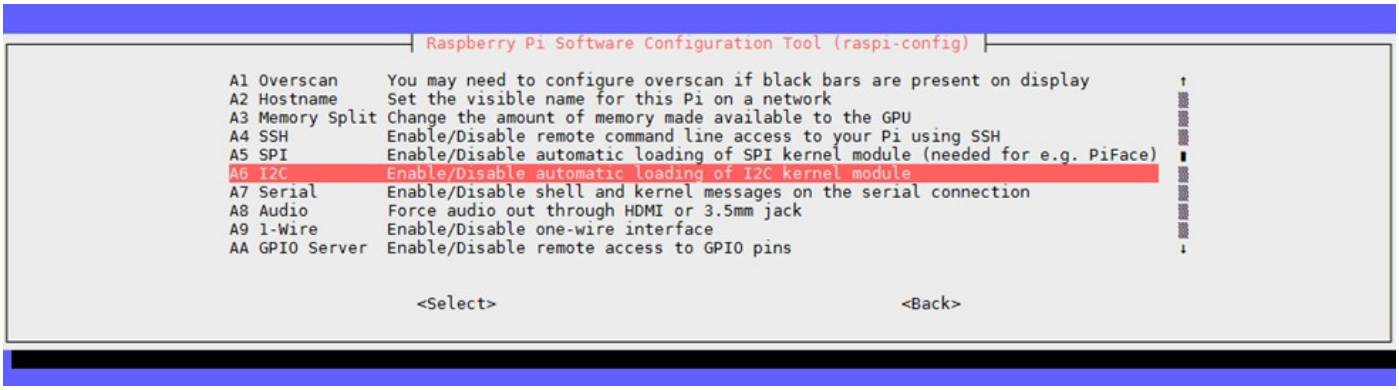
If you haven‘t already enabled I2C on your Raspberry, you need to catch this up in the configuration. Open up the configuration with the following command:

```
sudo raspi-config
```

In the now opened window, enter the **Advanced Options** section.



Choose and enable the **I2C** option.



Now you need to add two entries in the modules-file.
Therefore you need to open this file:

```
sudo nano /etc/modules
```

```
i2c-bcm2708
i2c-dev
```

Append the following two lines at the very end of the file:

Save the file with the combination **CTRL + O** and leave the editor with the combination **CTRL+X**.

```
sudo reboot
```

Now you need to restart your Raspberry Pi with the following command:

Now we can download the LCD driver.
We are using the **Raspberry Pi LCD 4x20 I²C python driver** to allow you a quick and easy use.
These drivers are made by **CaptainStouf** and are published under the **GNU General Public License v2.0** and are also available here.

At first, create a new folder and download the driver-package:

```
mkdir I2C-LCD && cd I2C-LCD
wget http://tutorials-raspberrypi.de/wp-content/uploads/scripts/
hd44780_i2c.zip
unzip hd44780_i2c.zip
```

Now we start our first communication with the display.
Please note, that the display needs to be connected.

```
sudo i2cdetect -y 1
```

The output should look like the following example:

```
     0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f
00:          -- -- -- -- -- -- -- -- -- -- -- -- --
10: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
20: -- -- -- -- -- -- -- 27 -- -- -- -- -- -- -- --
30: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
40: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
50: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
60: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
70: -- -- -- -- -- -- -- --
```

The executed check has detected an I2C-Signal at the address **27**.
This is the displays default address.
But the address may vary in some cases.
If your address is not 27, you need to modify your address in the driver-settings.

Therefore, open up the driver-settings-file:

```
sudo nano lcddriver.py
```

Modify the command **ADDRESS = 0x27** and change the **27** to match your address-number, which you have seen in the I2C-check before.
After that, you can save and leave the editor with the usual combination.

2.4 Communication with the display

The display-installation is now finished.

```
sudo nano LCD.py
```

The last step is to create a new file and to start the operation with the display:

Enter the following code-example.
The message inside the command **lcd.lcd_display_string** sets the shown text on the display.
The number next to it addresses the specific line.
Of course you can modify the text to your needs.

```
import lcddriver
from time import *

lcd = lcddriver.lcd()
lcd.lcd_clear()

lcd.lcd_display_string("     joy-IT", 1)
lcd.lcd_display_string("", 2)
lcd.lcd_display_string("      I2C Serial", 3)
lcd.lcd_display_string("         LCD", 4)
```

After entering the example, save and leave the editor with the usual combination.

```
sudo python LCD.py
```

3. Support

We also support you after your purchase.
If there are any questions left or if you encounter any problems please feel free to contact us my mail, phone or by our ticket-supportsystem on our website.

Mail: service@joy-it.net

Ticket-System: http://support.joy-it.net

Phone: +49 (0)2845 98469 – 66 (11- 18 Uhr)

Please visit our website for more informations:

www.joy-it.net