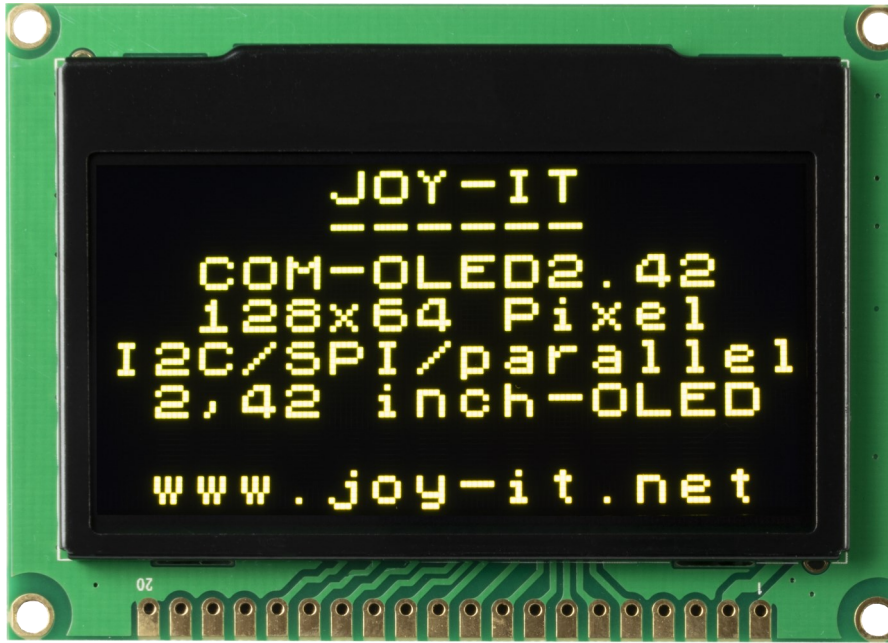


# OLED-DISPLAYMODUL

COM-OLED2.42



## 1. ALLGEMEINE INFORMATIONEN

Sehr geehrter Kunde,  
vielen Dank, dass Sie sich für unser Produkt entschieden haben. Im Folgenden zeigen wir Ihnen, was bei der Inbetriebnahme und der Verwendung zu beachten ist.

Sollten Sie während der Verwendung unerwartet auf Probleme stoßen, so können Sie uns selbstverständlich gerne kontaktieren.

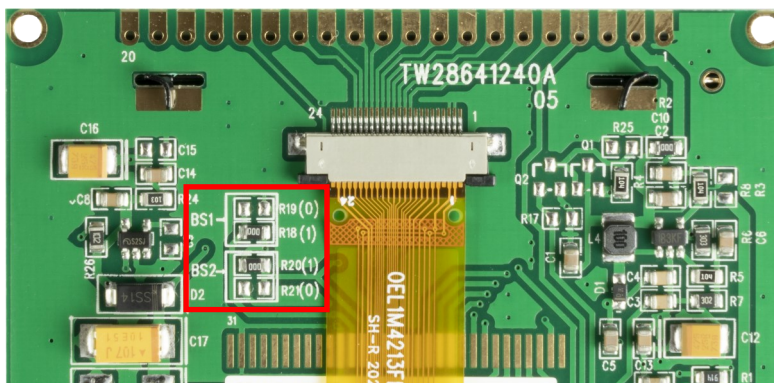
## 2. EINRICHTUNG DER DISPLAYSCHNITTSTELLE

Das Display kann auf 4 verschiedene Arten angesteuert werden, über I2C, SPI, 8-bit parallel 6800-Interface und 8-bit parallel 8080-Interface.

Das Display wird für die Ansteuerung über SPI vorkonfiguriert ausgeliefert. Wenn Sie eine der anderen Ansteuerungsarten verwenden möchten, müssen Sie die dafür vorgesehenen Widerstände BS1 und BS2 auf der Rückseite der Platine umlöten.

In der Tabelle können Sie sehen, wie die Widerstände für den jeweiligen Modus gesetzt werden müssen.

	6800-parallel	8080-parallel	I2C	SPI
BS1	0	1	1	0
BS2	1	1	0	0



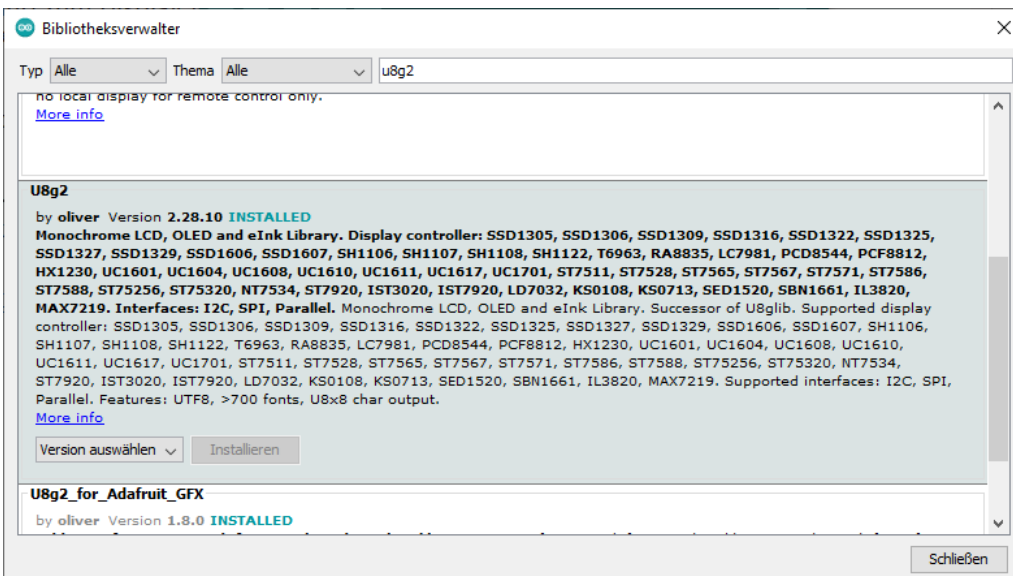
## 3. VEWRWENDUNG MIT EINEM ARDUINO

Da das Display mit einem 3 V Logiklevel arbeitet und die meisten Arduinos mit 5V, verwenden wir in diesem Beispiel einen Arduino Pro Mini 3,3 V.

Wenn Sie einen Arduino mit 5 V Logiklevel verwenden möchten, wie zum Beispiel einen Arduino Uno, müssen Sie alle Datenleitungen, die vom Arduino zum Display führen, mit einem Logiklevel-Konverter von 5 V auf 3,3 V herabsetzen.

Zunächst müssen Sie die benötigte Bibliothek in Ihrer Arduino IDE installieren.

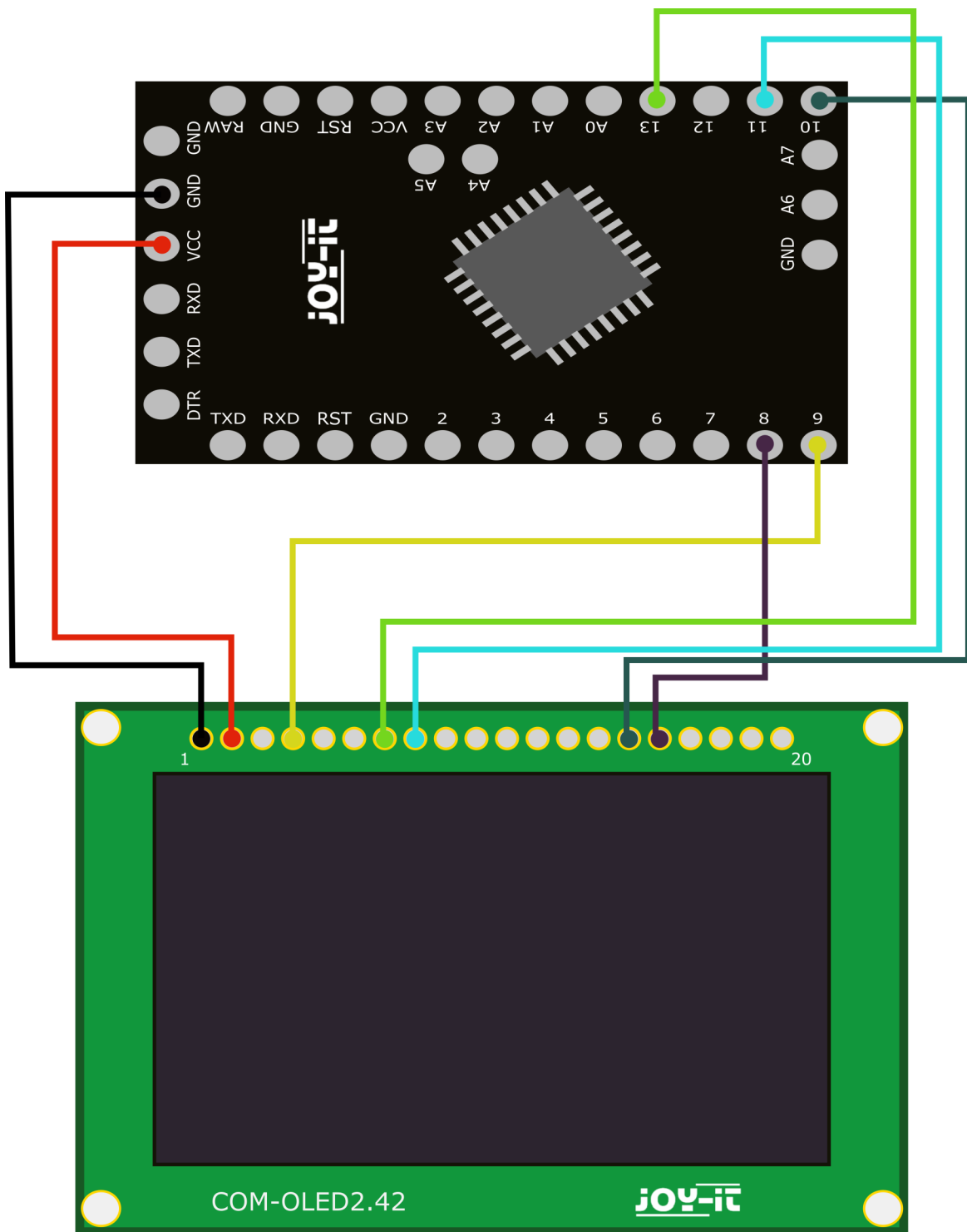
Suchen Sie dazu unter **Werkzeuge** -> **Bibliotheken verwalten...** nach **u8g2** und installieren Sie die Bibliothek **U8g2 by oliver**.



## SPI-Interface

Anschlussbelegung

<b>Display Pin</b>	1	2	4	7	8	15	16
<b>Arduino Pro Mini Pin</b>	GND	3,3V (VCC)	9	13	11	10	8



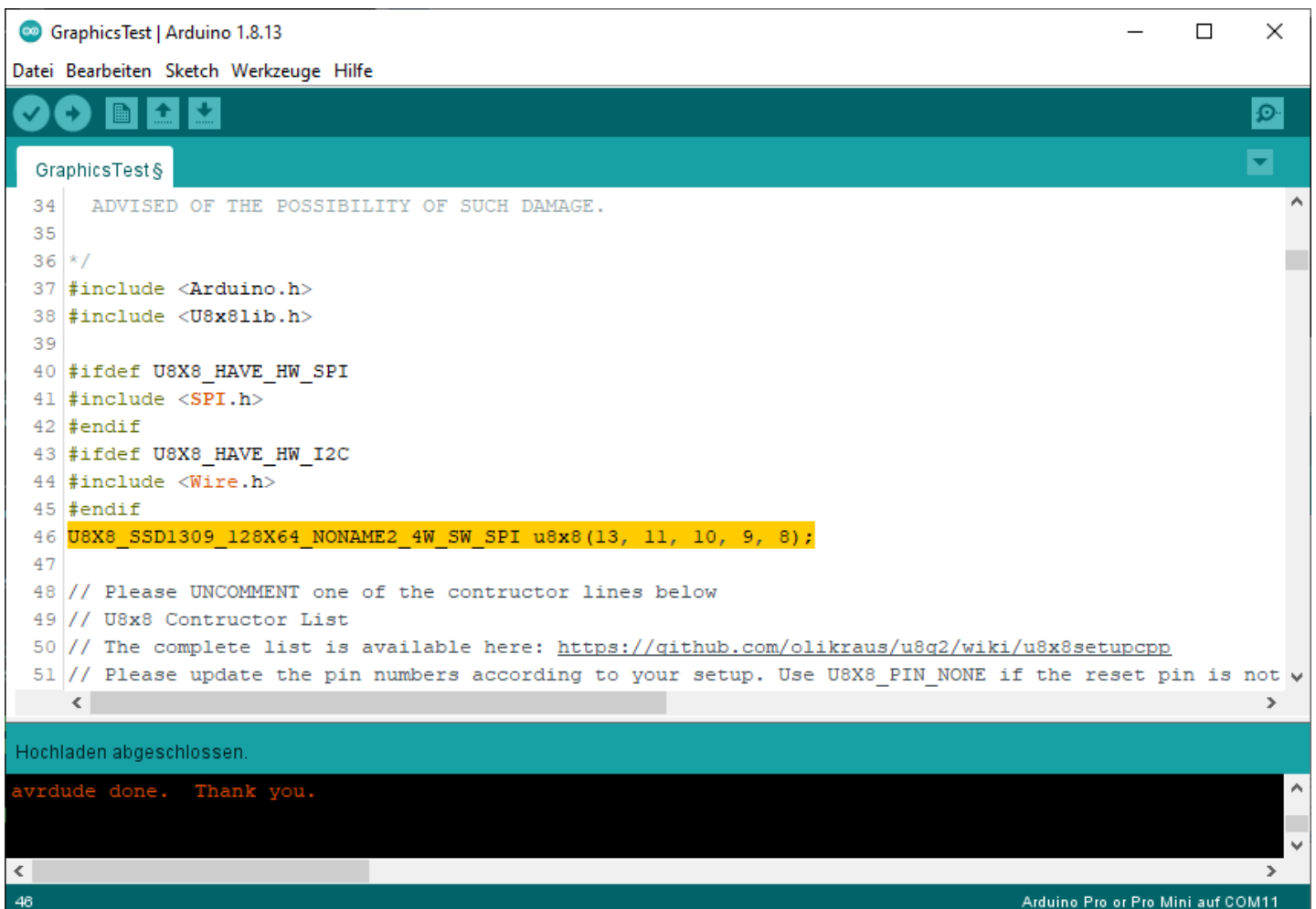
## SPI-Interface

Öffnen Sie nun das Codebeispiel GraphicTest der Bibliothek.  
Klicken Sie dazu auf:

**Datei -> Beispiele -> U8g2 -> u8x8-> GraphicTest**

Fügen Sie nun den folgenden Konstruktor für das Display in das Programm ein, wie im Bild unten zusehen:

```
U8X8_SSD1309_128X64_NONAME2_4W_SW_SPI u8x8(13, 11, 10, 9, 8);
```



```
GraphicsTest | Arduino 1.8.13
Datei Bearbeiten Sketch Werkzeuge Hilfe
GraphicsTest$
34 ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
35
36 */
37 #include <Arduino.h>
38 #include <U8x8lib.h>
39
40 #ifdef U8X8_HAVE_HW_SPI
41 #include <SPI.h>
42 #endif
43 #ifdef U8X8_HAVE_HW_I2C
44 #include <Wire.h>
45 #endif
46 U8X8_SSD1309_128X64_NONAME2_4W_SW_SPI u8x8(13, 11, 10, 9, 8);
47
48 // Please UNCOMMENT one of the constructor lines below
49 // U8x8 Constructor List
50 // The complete list is available here: https://github.com/olikraus/u8g2/wiki/u8x8setupcpp
51 // Please update the pin numbers according to your setup. Use U8X8_PIN_NONE if the reset pin is not
< >
```

Hochladen abgeschlossen.

```
avrdude done. Thank you.
```

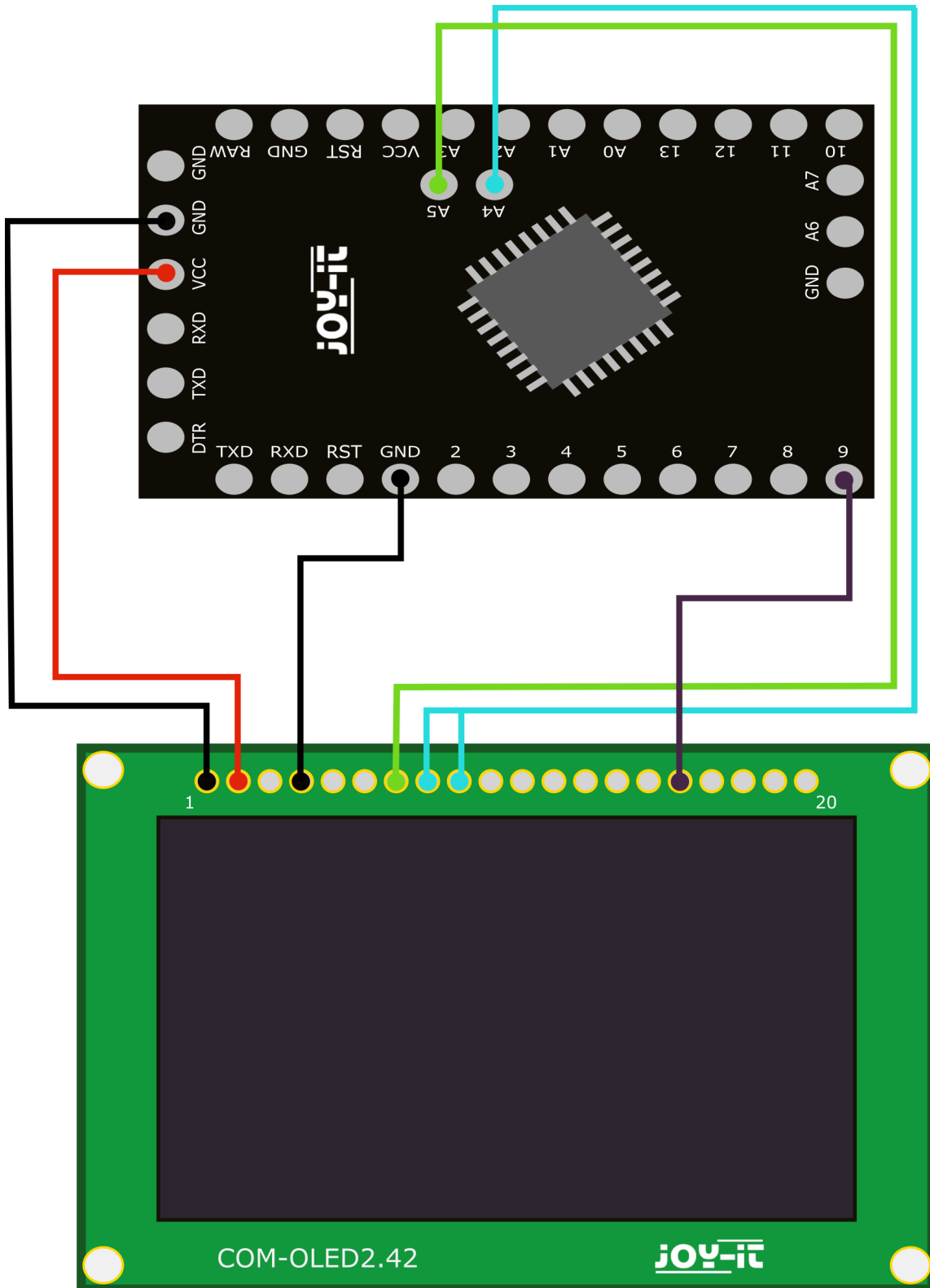
48 Arduino Pro or Pro Mini auf COM11

Nun können Sie das Beispiel auf Ihren Arduino hochladen.

## I2C-Interface

Anschlussbelegung

Display Pin	1	2	4	7	8	9	16
Arduino Pro Mini Pin	GND	3,3V (VCC)	GND	A5	A4	A4	9



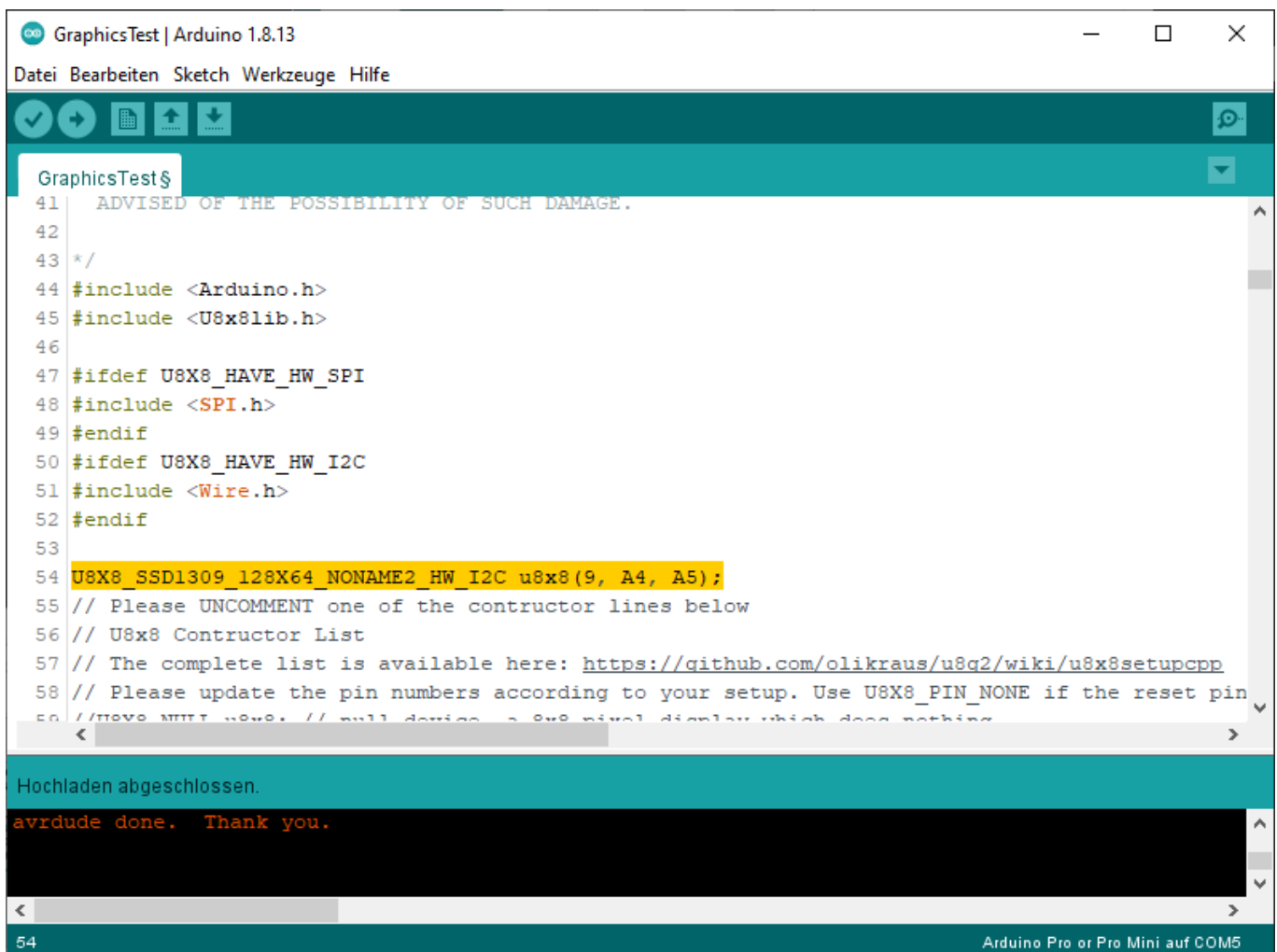
## I2C-Interface

Öffnen Sie nun das Codebeispiel GraphicTest der Bibliothek.  
Klicken Sie dazu auf:

**Datei -> Beispiele -> U8g2 -> u8x8-> GraphicTest**

Fügen Sie nun den folgenden Konstruktor für das Display in das Programm ein, wie im Bild unten zusehen:

```
U8X8_SSD1309_128X64_NONAME2_HW_I2C u8x8(9, A4, A5);
```



```
GraphicsTest | Arduino 1.8.13
Datei Bearbeiten Sketch Werkzeuge Hilfe

GraphicsTest$
41  ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
42
43  */
44  #include <Arduino.h>
45  #include <U8x8lib.h>
46
47  #ifdef U8X8_HAVE_HW_SPI
48  #include <SPI.h>
49  #endif
50  #ifdef U8X8_HAVE_HW_I2C
51  #include <Wire.h>
52  #endif
53
54  U8X8_SSD1309_128X64_NONAME2_HW_I2C u8x8(9, A4, A5);
55  // Please UNCOMMENT one of the constructor lines below
56  // U8x8 Constructor List
57  // The complete list is available here: https://github.com/olikraus/u8g2/wiki/u8x8setupcpp
58  // Please update the pin numbers according to your setup. Use U8X8_PIN_NONE if the reset pin
59  // does not exist. // null device = 8x8 pixel display which does nothing

Hochladen abgeschlossen.
avrdude done. Thank you.

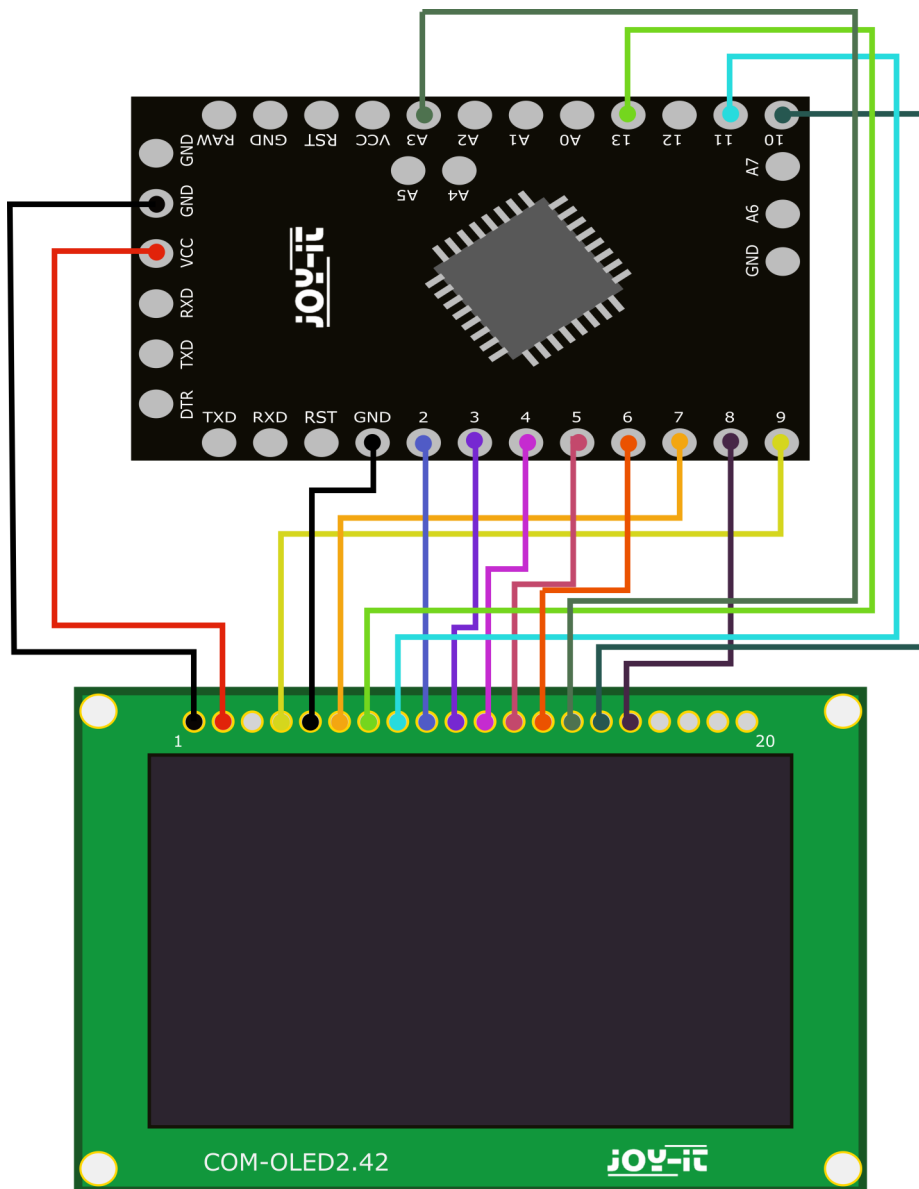
54 Arduino Pro or Pro Mini auf COM5
```

Nun können Sie das Beispiel auf Ihren Arduino hochladen.

## 8 bit Parallel 6800-Interface

Anschlussbelegung

Display Pin	1	2	4	5	6	7	8	9	10	11	12	13	14	15	16
Arduino Pro Mini Pin	GND	3,3V (VCC)	9	GND	7	13	11	2	3	4	5	6	A3	10	8



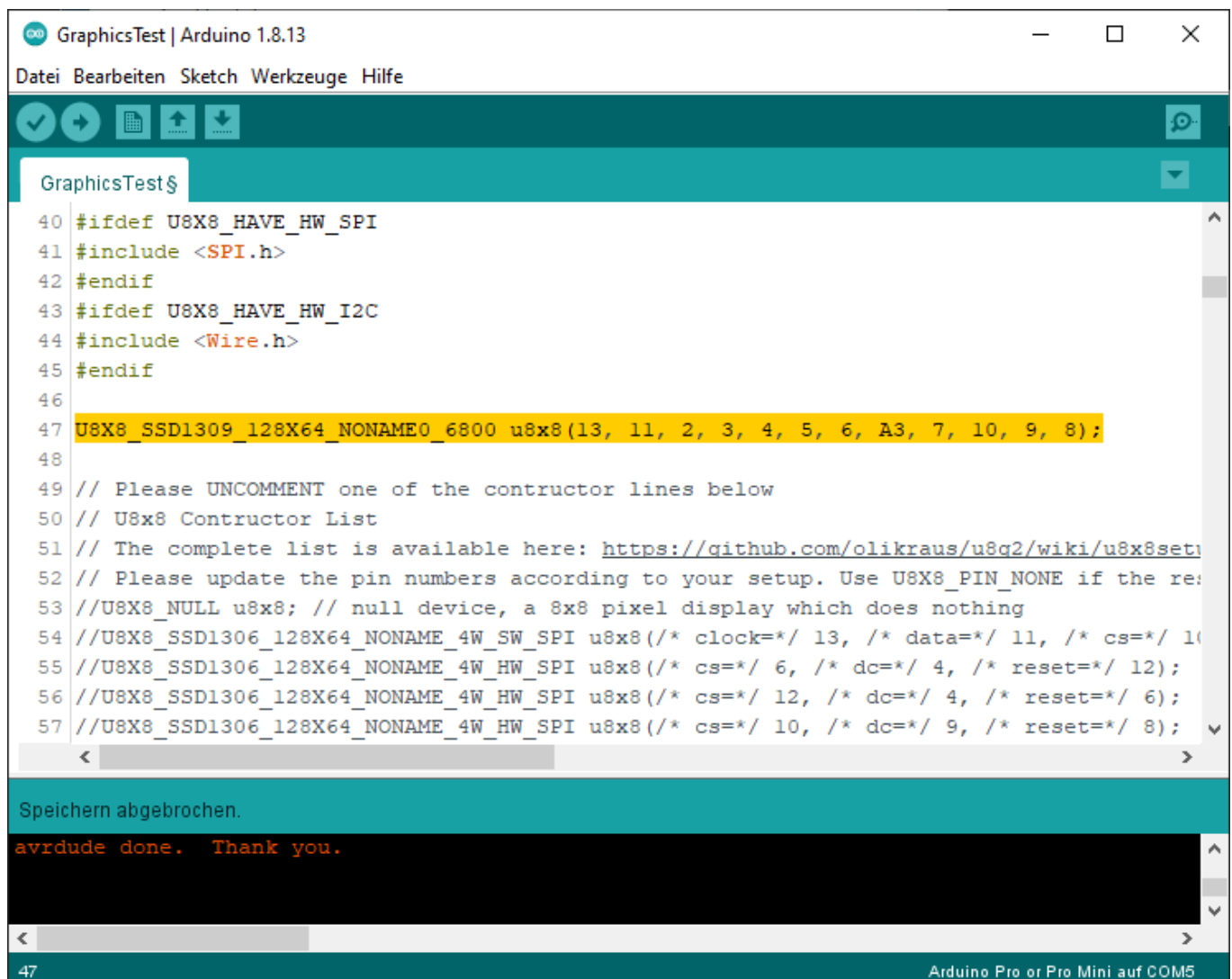
## 8 bit Parallel 6800-Interface

Öffnen Sie nun das Codebeispiel GraphicTest der Bibliothek.  
Klicken Sie dazu auf:

**Datei -> Beispiele -> U8g2 -> u8x8 -> GraphicTest**

Fügen Sie nun den folgenden Konstruktor für das Display in das Programm ein, wie im Bild unten zusehen:

```
U8X8_SSD1309_128X64_NONAME0_6800 u8x8(13, 11, 2, 3, 4, 5, 6,
A3, 7, 10, 9, 8);
```



```
GraphicsTest | Arduino 1.8.13
Datei Bearbeiten Sketch Werkzeuge Hilfe

GraphicsTest$
40 #ifndef U8X8_HAVE_HW_SPI
41 #include <SPI.h>
42 #endif
43 #ifndef U8X8_HAVE_HW_I2C
44 #include <Wire.h>
45 #endif
46
47 U8X8_SSD1309_128X64_NONAME0_6800 u8x8(13, 11, 2, 3, 4, 5, 6, A3, 7, 10, 9, 8);
48
49 // Please UNCOMMENT one of the constructor lines below
50 // U8x8 Constructor List
51 // The complete list is available here: https://github.com/olikraus/u8g2/wiki/u8x8setu
52 // Please update the pin numbers according to your setup. Use U8X8_PIN_NONE if the res
53 //U8X8_NULL u8x8; // null device, a 8x8 pixel display which does nothing
54 //U8X8_SSD1306_128X64_NONAME_4W_SW_SPI u8x8(/* clock=*/ 13, /* data=*/ 11, /* cs=*/ 10
55 //U8X8_SSD1306_128X64_NONAME_4W_HW_SPI u8x8(/* cs=*/ 6, /* dc=*/ 4, /* reset=*/ 12);
56 //U8X8_SSD1306_128X64_NONAME_4W_HW_SPI u8x8(/* cs=*/ 12, /* dc=*/ 4, /* reset=*/ 6);
57 //U8X8_SSD1306_128X64_NONAME_4W_HW_SPI u8x8(/* cs=*/ 10, /* dc=*/ 9, /* reset=*/ 8);

Speichern abgebrochen.
avrdupe done. Thank you.

47 Arduino Pro or Pro Mini auf COM5
```

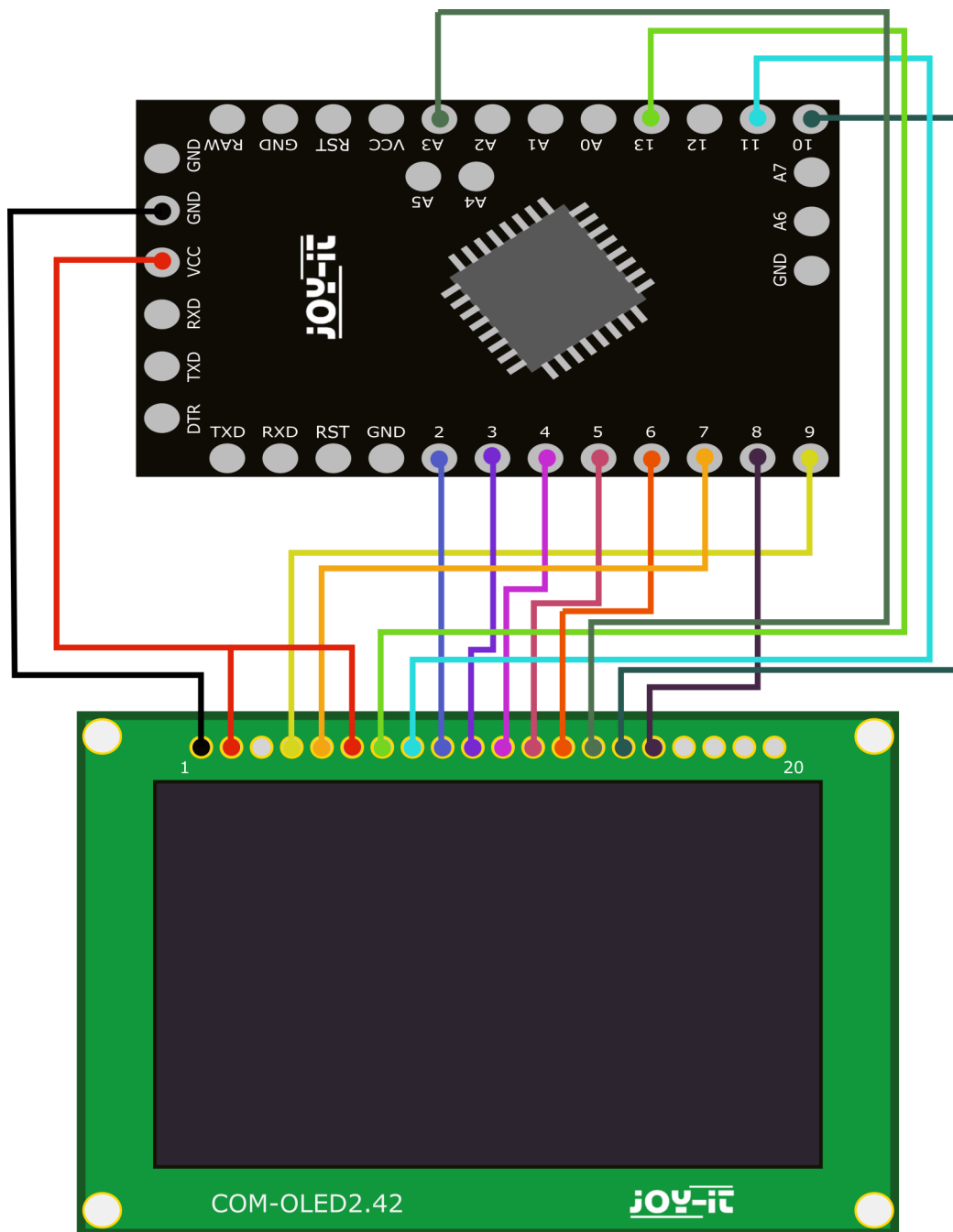
Nun können Sie das Beispiel auf Ihren Arduino hochladen.



## 8 bit Parallel 8080-Interface

Anschlussbelegung

Display Pin	1	2	4	5	6	7	8	9	10	11	12	13	14	15	16
Arduino Pro	GND	3,3V	9	7	3,3V	13	11	2	3	4	5	6	A3	10	8



## 8 bit Parallel 8080-Interface

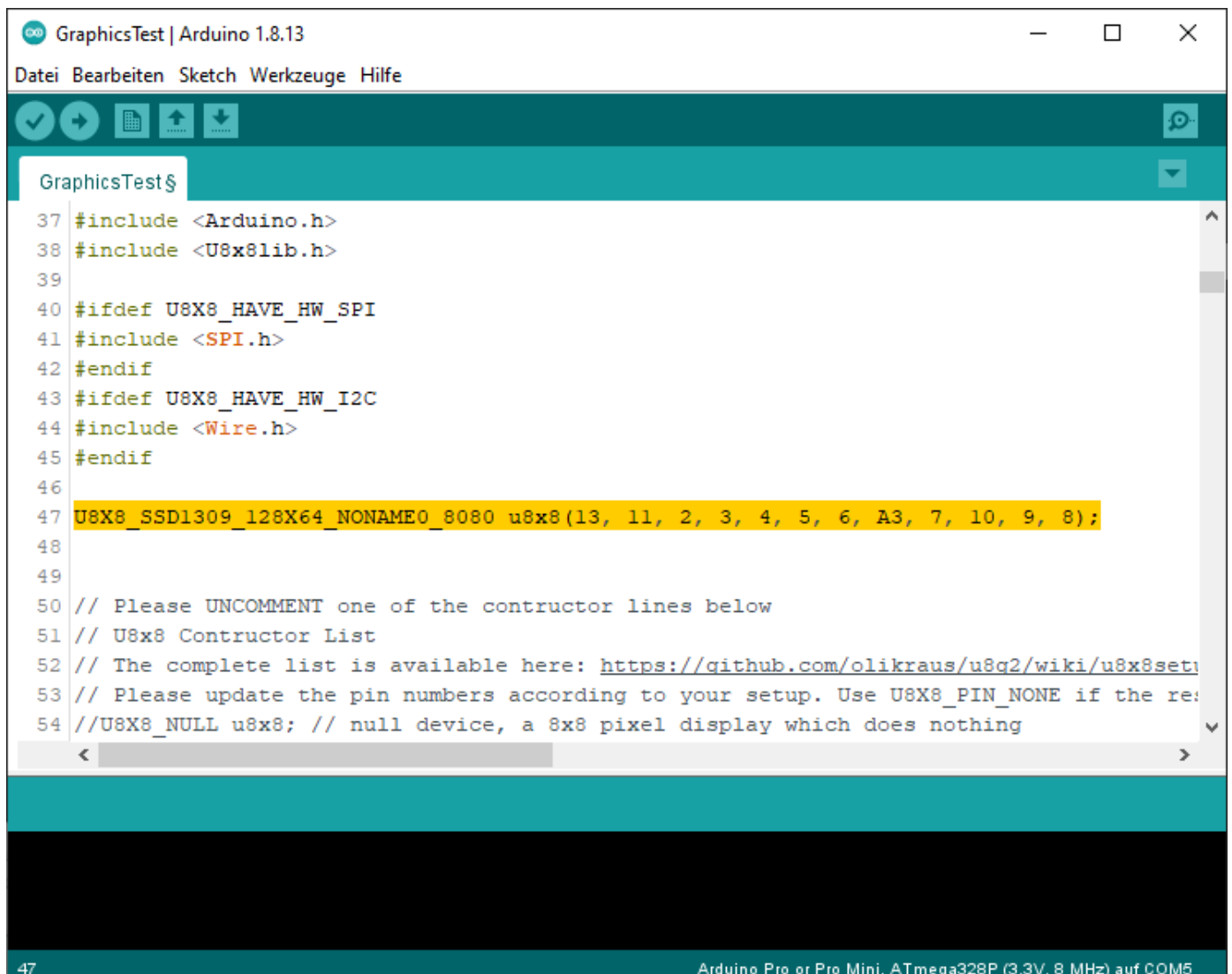
Öffnen Sie nun das Codebeispiel GraphicTest der Bibliothek.

Klicken Sie dazu auf:

**Datei -> Beispiele -> U8g2 -> u8x8 -> GraphicTest**

Fügen Sie nun den folgenden Konstruktor für das Display in das Programm ein, wie im Bild unten zusehen:

```
U8X8_SSD1309_128X64_NONAME0_8080 u8x8(13, 11, 2, 3, 4, 5, 6,
A3, 7, 10, 9, 8);
```



```
GraphicsTest | Arduino 1.8.13
Datei Bearbeiten Sketch Werkzeuge Hilfe
GraphicsTest$
37 #include <Arduino.h>
38 #include <U8x8lib.h>
39
40 #ifdef U8X8_HAVE_HW_SPI
41 #include <SPI.h>
42 #endif
43 #ifdef U8X8_HAVE_HW_I2C
44 #include <Wire.h>
45 #endif
46
47 U8X8_SSD1309_128X64_NONAME0_8080 u8x8(13, 11, 2, 3, 4, 5, 6, A3, 7, 10, 9, 8);
48
49
50 // Please UNCOMMENT one of the constructor lines below
51 // U8x8 Constructor List
52 // The complete list is available here: https://github.com/olikraus/u8g2/wiki/u8x8set
53 // Please update the pin numbers according to your setup. Use U8X8_PIN_NONE if the res
54 //U8X8_NULL u8x8; // null device, a 8x8 pixel display which does nothing
```

Nun können Sie das Beispiel auf Ihren Arduino hochladen.

## 4. VERWENDUNG MIT DEM RASPBERRY PI

Um die Verwendung des Displays mit dem Raspberry Pi besonders einfach zu gestalten, verwenden wir die `luma.oled` Bibliothek. Die zur Installation benötigten Abhängigkeiten können Sie mit folgenden Befehlen installieren:

```
sudo apt-get update
```

```
sudo apt-get install git python3 python3-dev python3-pip python3-pip python3-pil libjpeg-dev zlib1g-dev libfreetype6-dev liblcms2-dev libopenjp2-7 libtiff5 build-essential libsdl-dev libportmidi-dev libsdl-ttf2.0-dev libsdl-mixer1.2-dev libsdl-image1.2-dev
```

Zunächst installieren wir die Bibliothek und laden die Beispieldateien herunter, dies geschieht mit folgenden Befehlen:

```
sudo -H pip3 install --upgrade luma.oled
```

```
sudo git clone https://github.com/rm-hull/luma.examples.git
```

Als nächstes müssen Sie dem Benutzer (in diesem Fall „pi“) die Berechtigung geben um auf die benötigte Hardware zuzugreifen. Sollten Sie nicht den Benutzer „pi“ verwenden, passen Sie den Benutzer am Ende des Befehls entsprechend an.

```
sudo usermod -a -G spi,gpio,i2c pi
```

Anschließend starten Sie Ihren Raspberry Pi mit folgendem Befehl neu:

```
sudo reboot
```

Nach dem Neustart geben Sie nun folgenden Befehl in das Terminal ein:

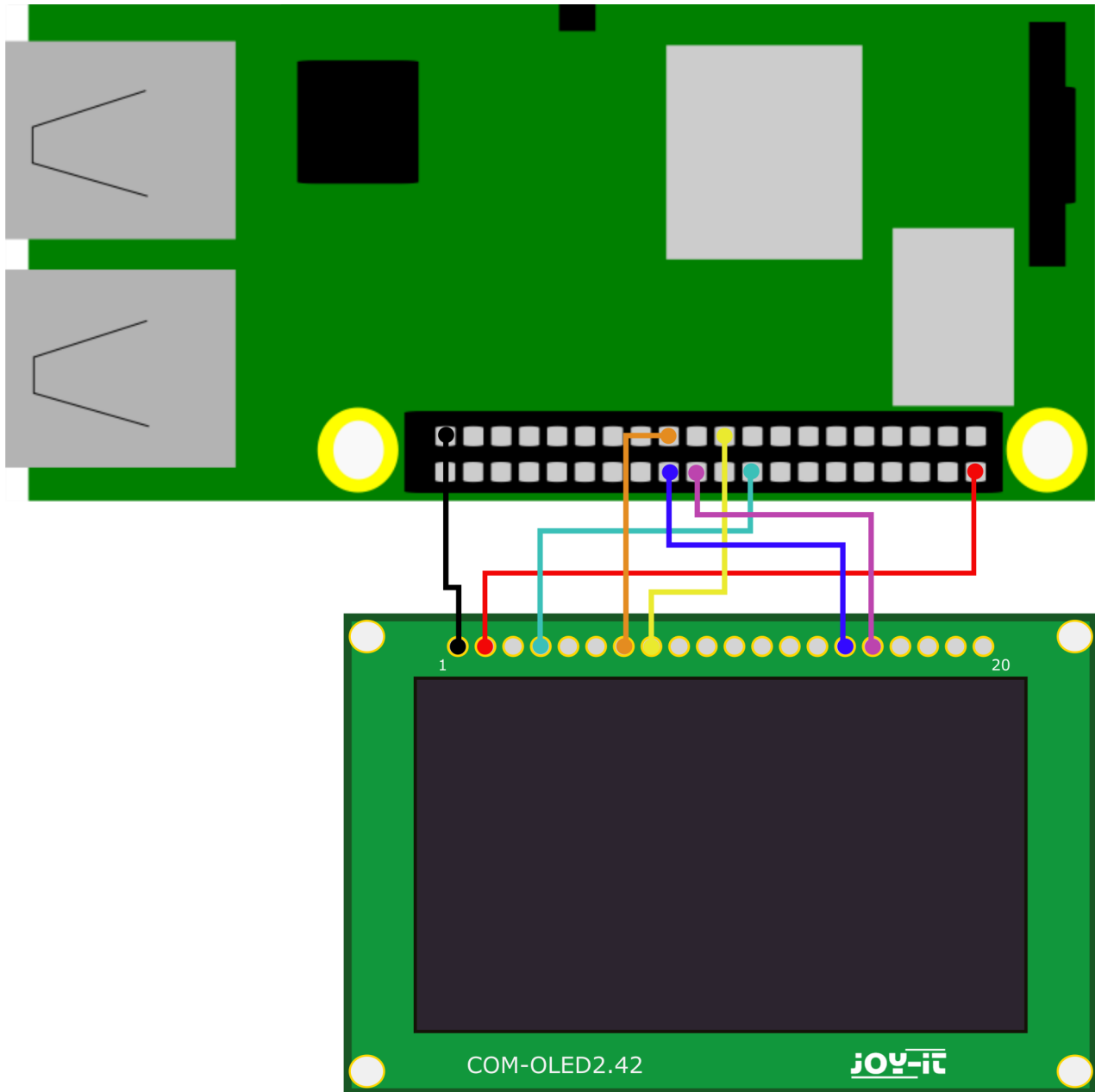
```
sudo raspi-config
```

Dort können Sie nun unter **3 Interface Options SPI** und **I2C** aktivieren, damit Sie beide Schnittstellen verwenden können.

## SPI-Interface

Anschlussbelegung

<b>Display Pin</b>	1	2	4	7	8	15	16
<b>Raspberry Pin</b>	GND	5V	Pin 18	Pin 23	Pin 19	Pin 24	Pin 22



Nachdem Sie das Display angeschlossen haben, können Sie mit den folgenden zwei Befehlen ein Beispiel Programm ausführen:

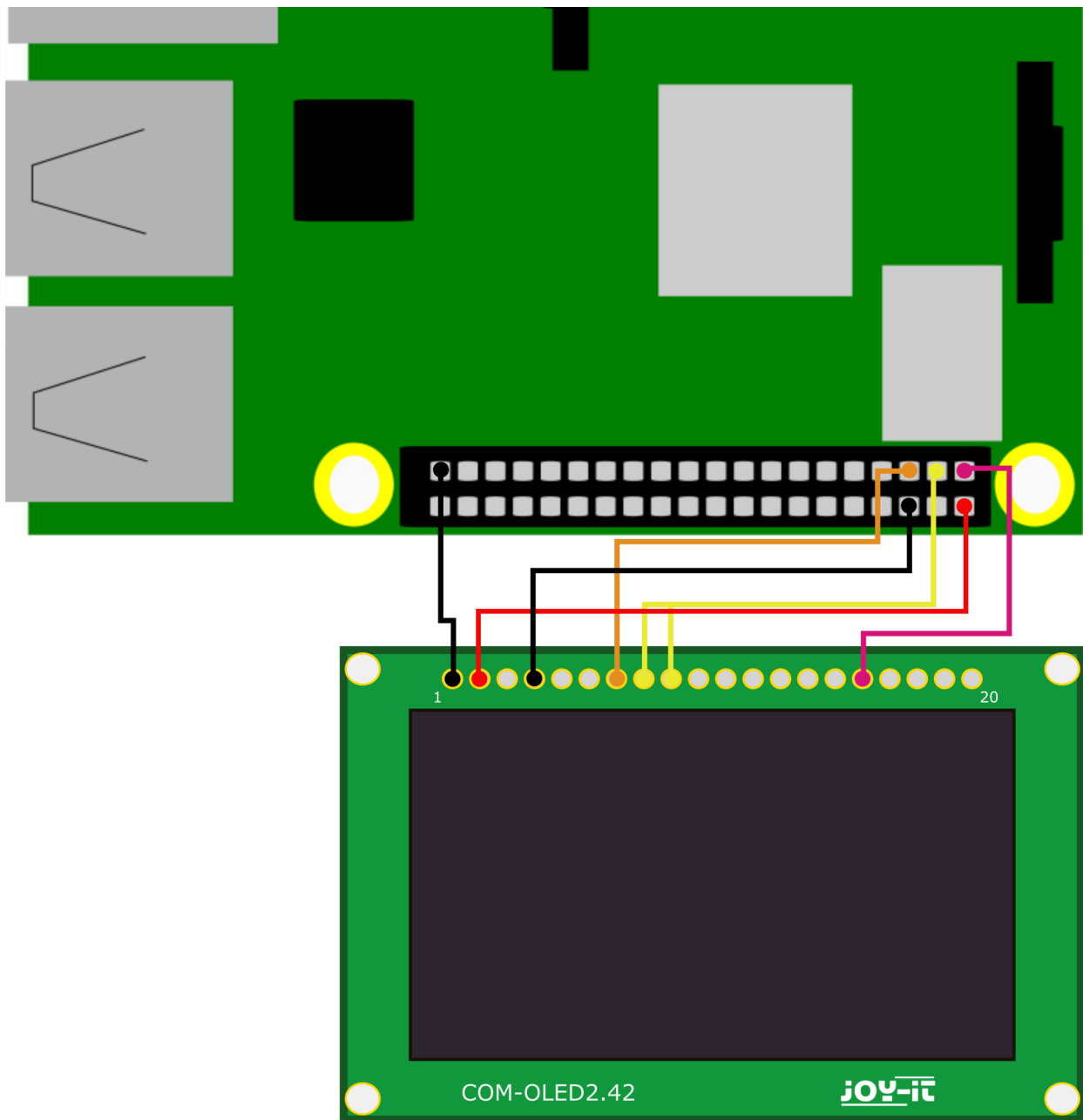
```
cd ~/luma.examples/examples/
```

```
sudo python3 demo.py -i spi
```

## I2C-Interface

Anschlussbelegung

<b>Display Pin</b>	1	2	4	7	8	9	16
<b>Raspberry Pin</b>	GND	5V	GND	Pin 5	Pin 3	Pin 3	3,3V



Nachdem Sie das Display angeschlossen haben, können Sie mit den folgenden zwei Befehlen ein Beispiel Programm ausführen:

```
cd ~/luma.examples/examples/
```

```
sudo python3 demo.py
```

## 5. SONSTIGE INFORMATIONEN

Unsere Informations- und Rücknahmepflichten nach dem Elektroggesetz (ElektroG)



### Symbol auf Elektro- und Elektronikgeräten:

Diese durchgestrichene Mülltonne bedeutet, dass Elektro- und Elektronikgeräte **nicht** in den Hausmüll gehören. Sie müssen die Altgeräte an einer Erfassungsstelle abgeben. Vor der Abgabe haben Sie Altbatterien und Altakkumulatoren, die nicht vom Altgerät umschlossen sind, von diesem zu trennen.

### Rückgabemöglichkeiten:

Als Endnutzer können Sie beim Kauf eines neuen Gerätes, Ihr Altgerät (das im Wesentlichen die gleiche Funktion wie das bei uns erworbene neue erfüllt) kostenlos zur Entsorgung abgeben. Kleingeräte bei denen keine äußere Abmessungen größer als 25 cm sind können unabhängig vom Kauf eines Neugerätes in haushaltsüblichen Mengen abgeben werden.

### Möglichkeit Rückgabe an unserem Firmenstandort während der Öffnungszeiten:

SIMAC Electronics GmbH, Pascalstr. 8, D-47506 Neukirchen-Vluyn

### Möglichkeit Rückgabe in Ihrer Nähe:

Wir senden Ihnen eine Paketmarke zu mit der Sie das Gerät kostenlos an uns zurücksenden können. Hierzu wenden Sie sich bitte per E-Mail an [Service@joy-it.net](mailto:Service@joy-it.net) oder per Telefon an uns.

### Informationen zur Verpackung:

Verpacken Sie Ihr Altgerät bitte transportsicher, sollten Sie kein geeignetes Verpackungsmaterial haben oder kein eigenes nutzen möchten kontaktieren Sie uns, wir lassen Ihnen dann eine geeignete Verpackung zukommen.

## 6. SUPPORT

Wir sind auch nach dem Kauf für Sie da. Sollten noch Fragen offen bleiben oder Probleme auftauchen stehen wir Ihnen auch per E-Mail, Telefon und Ticket-Supportsystem zur Seite.

E-Mail: [service@joy-it.net](mailto:service@joy-it.net)

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

Telefon: +49 (0)2845 98469 – 66 (10 - 17 Uhr)

Für weitere Informationen besuchen Sie unsere Website:

[www.joy-it.net](http://www.joy-it.net)