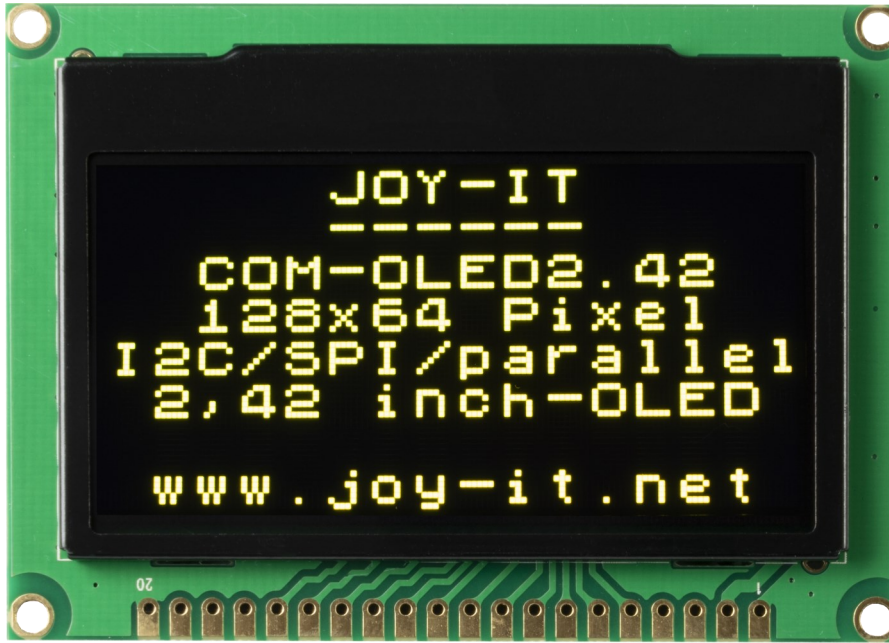


OLED-DISPLAY MODULE

COM-OLED2.42



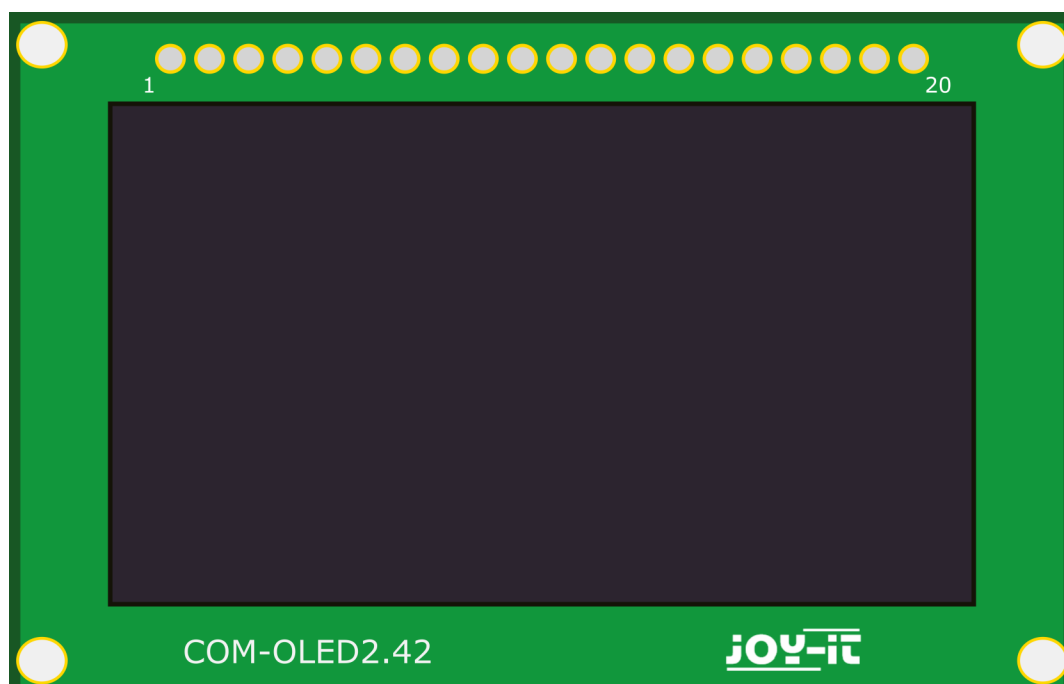
1. GENERAL INFORMATION

Dear customer,
thank you very much for choosing our product.
In the following, we will introduce you to what to observe while starting up and using this product.
Should you encounter any unexpected problems during use, please do

2. PIN ASSIGNMENT OF THE DISPLAY

Pin Designation	Pin number	I/O	Function
VSS	1	P	Logic circuit ground This is a ground pin. It also serves as a reference for the logic pins. It must be connected to the external ground.
VDD	2	3,3 - 5V	Power supply for display module circuit This is a power supply pin.
V0	3	-	Voltage supply for OEL panel This is the most positive voltage supply pin of the chip. Please do not connect it.
A0	4	I	Data/Command Control This pin is a data/command control pin. When the pin is pulled high, the input at D7~D0 is treated as display data. When the pin is pulled low, the input at D7~D0 is transferred to the command register.
/WR	5	I	Read/Write Select or Write This pin is an MCU interface input. When connected to a 68XX series microprocessor, this pin is used as a read/write select (R/W) input. Pull this pin high for read mode and pull it low for write mode. When the 80XX interface mode is selected, this pin is the write input (WR). The data write operation is initiated when this pin is pulled "Low" and CS is pulled "Low".
/RD	6	I	Read/Write Enable or Read This pin is an MCU interface input. When connected to a 68XX series microprocessor, this pin is used as an Enable(E) signal. The read/write operation is initiated when this pin is pulled high and CS is pulled low. When connected to an 80XX microprocessor, this pin receives the Read(RD) signal. The data read operation is initiated when this pin is pulled low and CS is pulled low.

Pin Designation	Pin number	I/O	Function															
DB0	7	I/O	Host data input/output bus These pins are bidirectional 8-bit data buses that connect to the microprocessor data bus. When serial mode is selected, D1 is the SDIN serial data input and D0 is the SCLK serial clock input.															
DB1	8	I/O																
DB2	9	I/O																
DB3	10	I/O																
DB4	11	I/O																
DB5	12	I/O																
DB6	13	I/O																
DB7	14	I/O																
/CS	15	I	Chip-Select This pin is the chip select input. The chip is only enabled for MCU communication when CS# is pulled low.															
/RESET	16	I	Power Reset for Controller and Driver This pin is a reset signal input. When the pin is low, the initialization of the chip is performed.															
NC (BS1)	17	H/L	Communication protocol selection															
NC (BS2)	18	H/L	These pins are inputs for selecting the MCU interface. See the following table:															
			<table><tr><th></th><th>68XX-parallel</th><th>80XX-parallel</th><th>I2C</th><th>Serial</th></tr><tr><td>BS1</td><td>0</td><td>1</td><td>1</td><td>0</td></tr><tr><td>BS2</td><td>1</td><td>1</td><td>0</td><td>0</td></tr></table>		68XX-parallel	80XX-parallel	I2C	Serial	BS1	0	1	1	0	BS2	1	1	0	0
	68XX-parallel	80XX-parallel	I2C	Serial														
BS1	0	1	1	0														
BS2	1	1	0	0														
NC	19	-	NC or connection to VSS.															
FG	20	0V	It must be connected to external ground.															



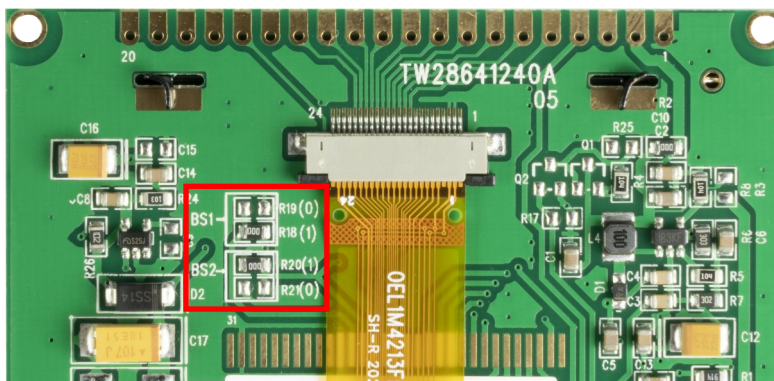
2. 1 SETUP OF THE DISPLAY INTERFACE

The display can be controlled in 4 different ways, via I2C, SPI, 8-bit parallel 6800 interface and 8-bit parallel 8080 interface.

The display is delivered pre-configured for control via SPI. If you want to use one of the other control methods, you have to re-solder the resistors BS1 and BS2 on the back of the board.

In the table, you can see how the resistors must be set for the respective mode.

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



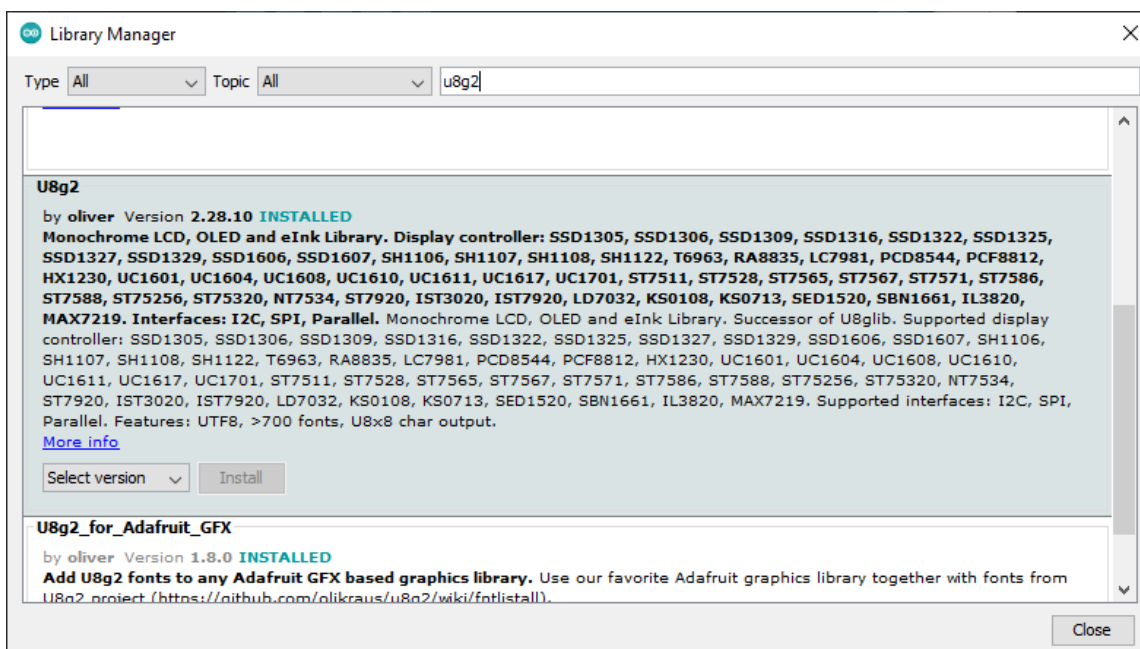
3. USE WITH AN ARDUINO

As the display works with a 3V logic level and most Arduinos with 5V, we use an Arduino Pro Mini 3.3V in this example.

If you want to use an Arduino with a 5V logic level, such as an Arduino Uno, you have to reduce all data lines leading from the Arduino to the display from 5V to 3.3V with a logic level converter.

First you need to install the required library in your Arduino IDE.

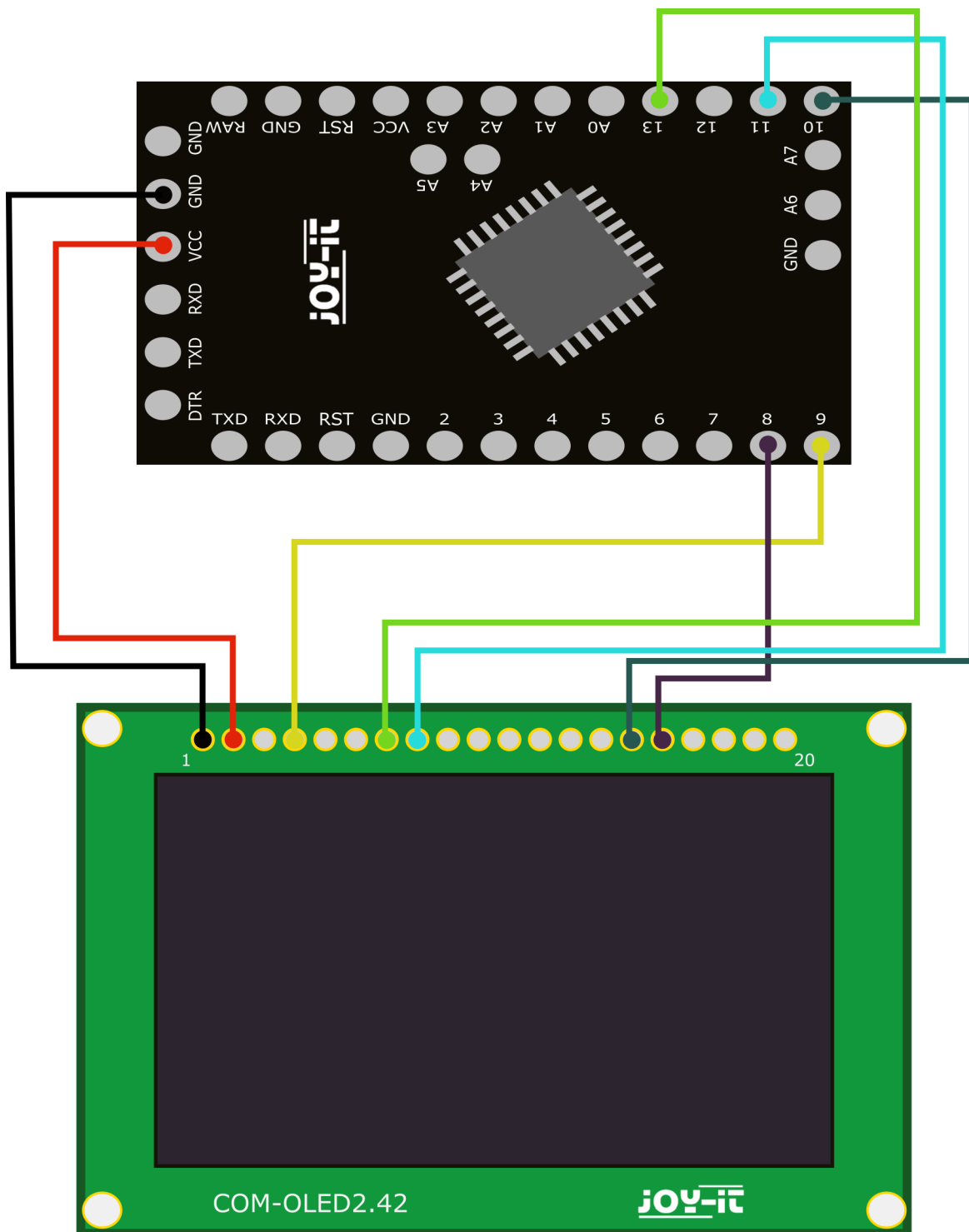
To do this, go to **Tools -> Manage Libraries...** Search for **u8g2** and install the library **U8g2 by oliver**



SPI-Interface

Wiring

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




SPI-Interface

Now open the GraphicTest code sample of the library.
To do this, click on:

File -> Examples -> U8g2 -> u8x8-> GraphicTest

Now insert the following constructor for the display into the programme,
as shown in the picture below:

```
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.

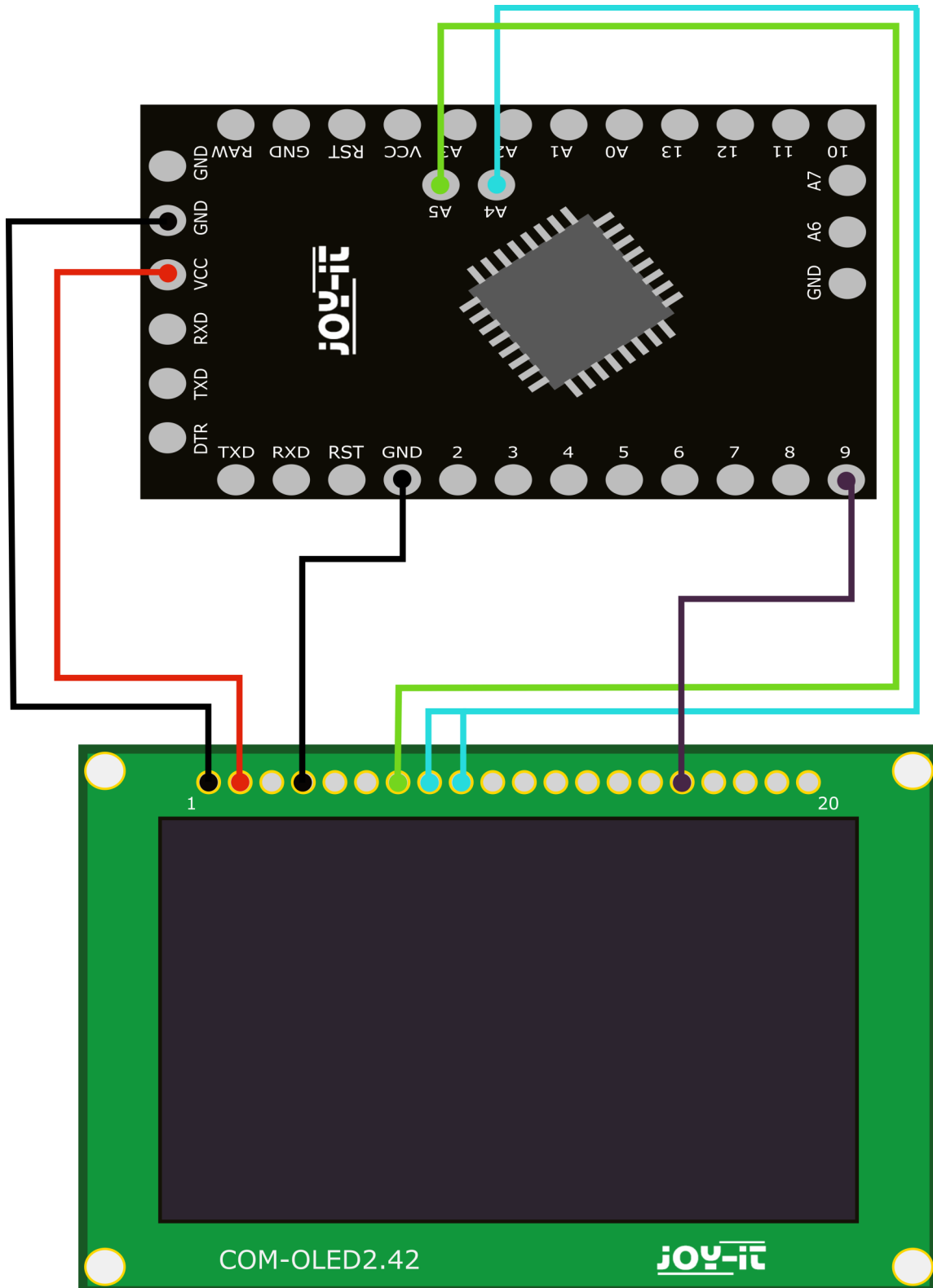
48 Arduino Pro or Pro Mini auf COM11
```

Now you can upload the example to your Arduino.

I2C-Interface

Wiring

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



I2C-Interface

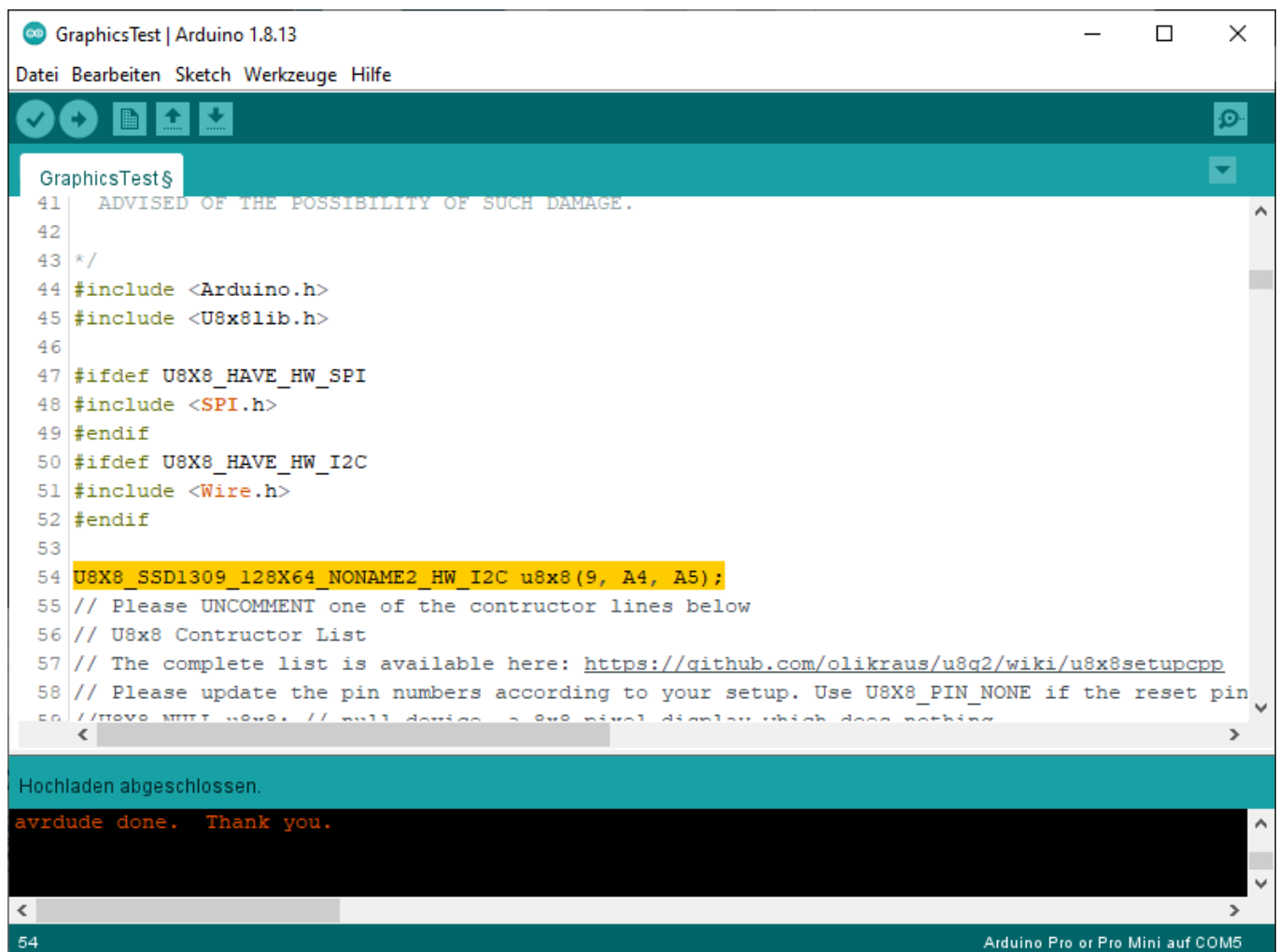
Now open the GraphicTest code sample of the library.

To do this, click on:

File -> Examples -> U8g2 -> u8x8-> GraphicTest

Now insert the following constructor for the display into the programme, as shown in the picture below:

```
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 <U8g2lib.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 // U8X8_PIN_NONE u8x8; // null device - a 8x8 pixel display which does nothing

Hochladen abgeschlossen.
avrdude done. Thank you.

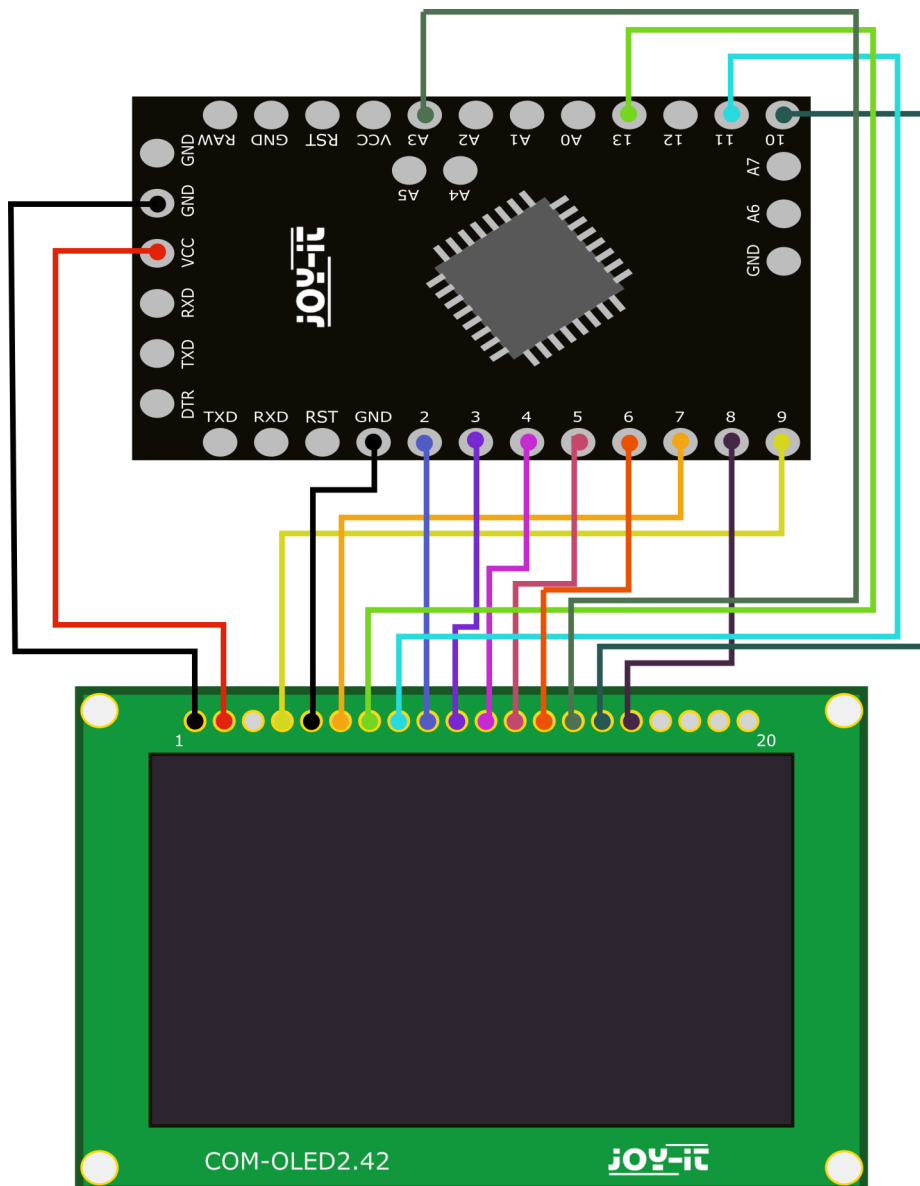
54 Arduino Pro or Pro Mini auf COM5
```

Now you can upload the example to your Arduino.

8 bit Parallel 6800-Interface

Wiring

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

Now open the GraphicTest code sample of the library.

To do this, click on:

File -> Examples -> U8g2 -> u8x8-> GraphicTest

Now insert the following constructor for the display into the programme, as shown in the picture below:

```
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/u8x8set
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.
avrdude done. Thank you.

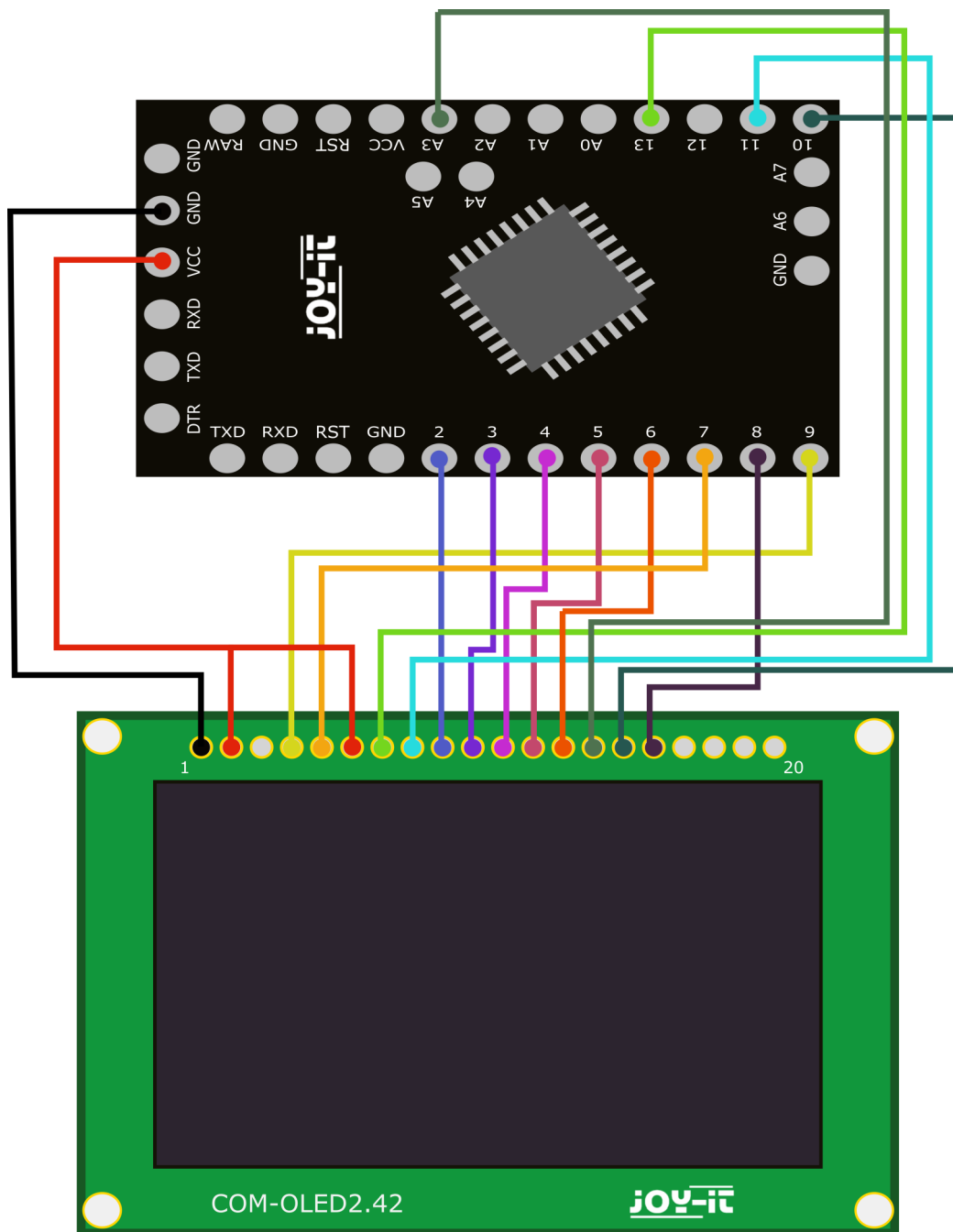
47 Arduino Pro or Pro Mini auf COM5
```

Now you can upload the example to your Arduino.

8 bit Parallel 8080-Interface

Wiring

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	7	3,3V (VCC)	13	11	2	3	4	5	6	A3	10	8



8 bit Parallel 8080-Interface

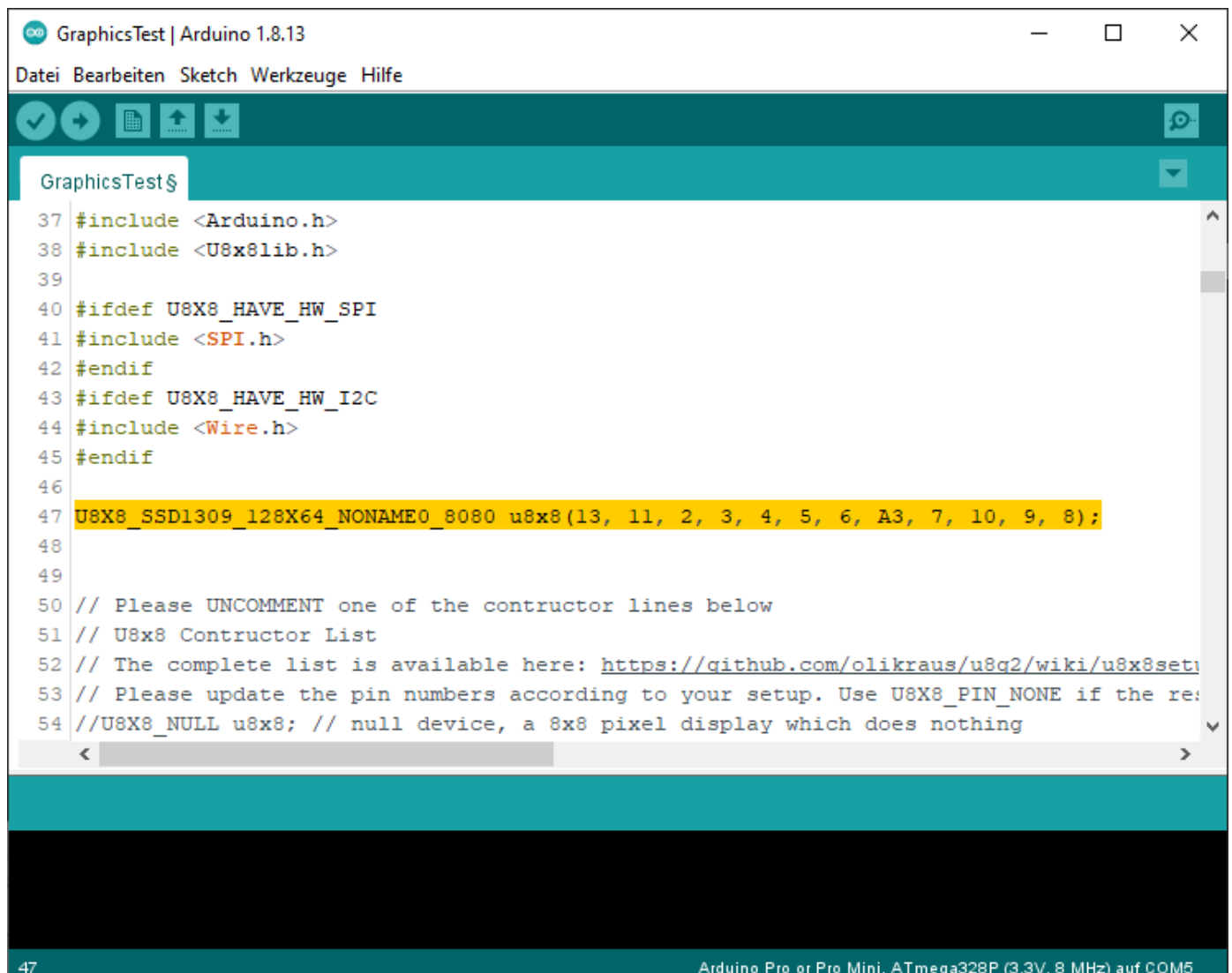
Now open the GraphicTest code sample of the library.

To do this, click on:

File -> Examples -> U8g2 -> u8x8-> GraphicTest

Now insert the following constructor for the display into the programme,

```
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/u8x8setu
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
```

Now you can upload the example to your Arduino.

4. USE WITH THE RASPBERRY PI



These instructions were written under Raspberry Pi OS Bookworm for the Raspberry Pi 4 and 5. No checks have been carried out with other/newer operating systems or hardware.

To make using the display with the Raspberry Pi particularly easy, we use the `luma.oled` library. You can install the dependencies required for installation with the following commands:

```
sudo apt install git python3-dev python3-pip python3-numpy  
libfreetype6-dev libjpeg-dev build-essential
```

```
sudo apt install libSDL2-dev libSDL2-image-dev libSDL2-mixer-  
dev libSDL2-ttf-dev libportmidi-dev
```

Now activate the required interfaces by entering the following command:

```
sudo raspi-config
```

You can now activate **SPI** and **I2C** under **3 Interface Options** so that you can use both interfaces.

You must now create the virtual environment for this project. To do this, enter the following commands:

```
mkdir your_project
```

```
cd your_project
```

```
python -m venv --system-site-packages env
```

```
source env/bin/activate
```

Now install the `luma` library with this command:

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

Download the sample files with the following command:

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

Now enter the following commands for installation:

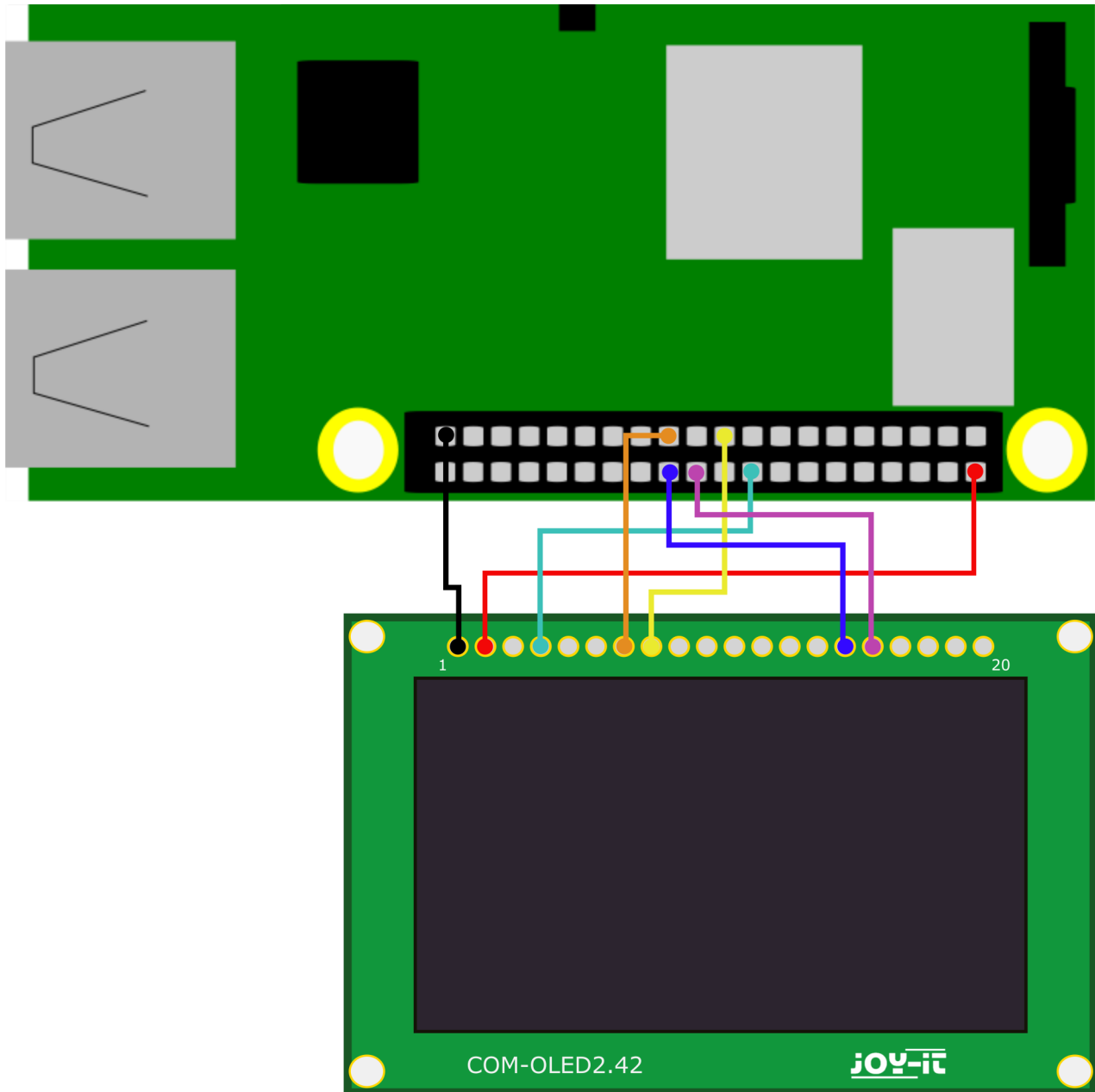
```
cd luma.examples
```

```
python3 setup.py install
```

SPI-Interface

Wiring

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



After you have connected the display, you can execute a sample programme with the following two commands:

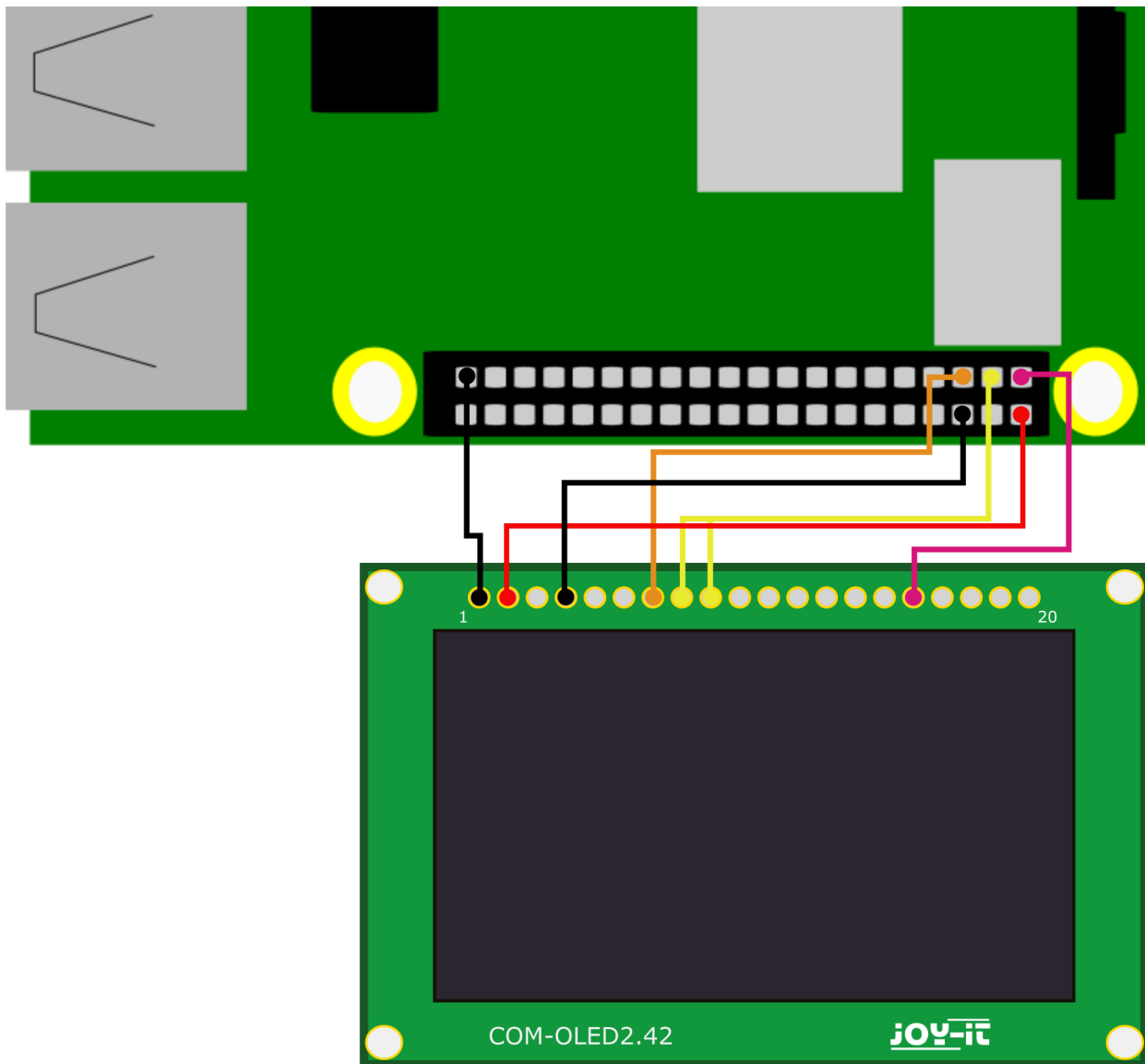
```
cd ~/your_project/luma.examples/examples/
```

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

I2C-Interface

Wiring

Display Pin	1	2	4	7	8	9	16
Raspberry Pin	GND	5V	GND	Pin 5	Pin 3	Pin 3	3,3V



After you have connected the display, you can execute a sample programme with the following two commands:

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

```
python3 demo.py
```

5. ADDITIONAL INFORMATION

Our information and take-back obligations according to the Electrical and Electronic Equipment Act (ElektroG)



Symbol on electrical and electronic equipment:

This crossed-out dustbin means that electrical and electronic appliances do not belong in the household waste. You must return the old appliances to a collection point.

Before handing over waste batteries and accumulators that are not enclosed by waste equipment must be separated from it.

Return options:

As an end user, you can return your old device (which essentially fulfils the same function as the new device purchased from us) free of charge for disposal when you purchase a new device.

Small appliances with no external dimensions greater than 25 cm can be disposed of in normal household quantities independently of the purchase of a new appliance.

Possibility of return at our company location during opening hours:

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

Possibility of return in your area:

We will send you a parcel stamp with which you can return the device to us free of charge. Please contact us by email at Service@joy-it.net or by telephone.

Information on packaging:

If you do not have suitable packaging material or do not wish to use your own, please contact us and we will send you suitable packaging.

6. SUPPORT

If there are still any issues pending or problems arising after your purchase, we will support you by e-mail, telephone and with our ticket support system.

Email: service@joy-it.net

Ticket system: <https://support.joy-it.net>

Telephone: +49 (0)2845 9360-50 (Mon - Thur: 09:00 - 17:00 o'clock CET,
Fri: 09:00 - 14:30 o'clock CET)

For further information please visit our website:

www.joy-it.net