GeeekPi 464

GeeekPi I2C 1602 LCD Display Module User Manual

Model: 464

1. Product Overview

The GeeekPi I2C 1602 LCD Display Module is a 16x2 character serial LCD with a blue backlight, designed for various DIY electronic projects. It utilizes the I2C protocol to minimize I/O port usage, simplifying wiring and integration with development boards.

Key Features:

- **I2C Protocol:** Reduces I/O port occupation for easier project integration.
- **16x2 Character Display:** Provides two lines of 16 characters each.
- **Blue Backlight:** Enhances visibility in various lighting conditions.
- **Adjustable Contrast:** Features a potentiometer for fine-tuning display contrast.
- **Wide Compatibility:** Compatible with popular development boards such as Arduino, Raspberry Pi, Tinkerboard, Nano Pi, Banana Pi, and STM32.
- **Power Supply:** Operates on 5V.
- **Default I2C Address:** 0x27 (Note: Some modules may use 0x3F).

Package Contents:

- 2 x GeeekPi I2C 1602 LCD Display Modules
- 8 x Jumper Wires

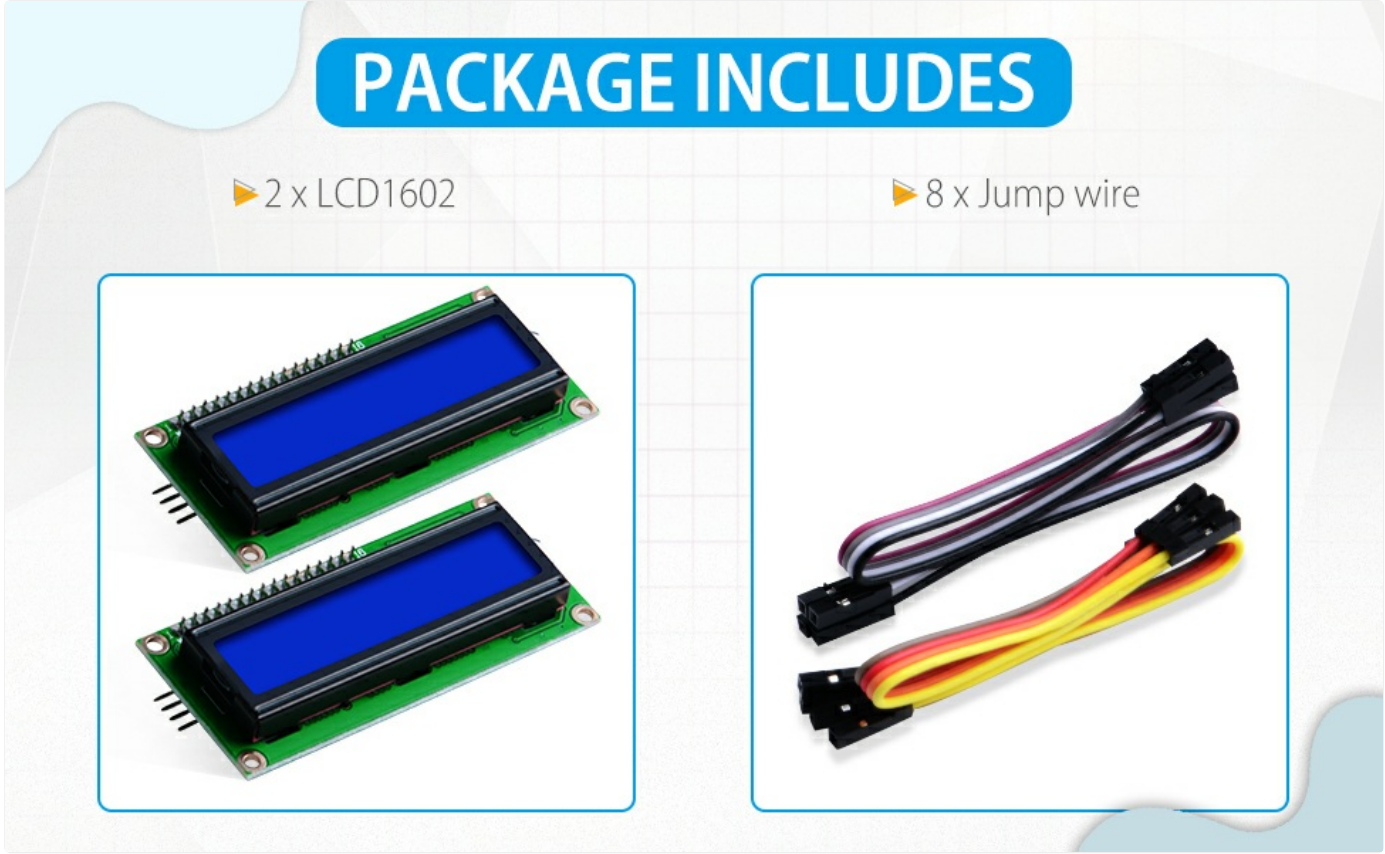


Image: Two GeeekPi 1602 LCD Display Modules and eight jumper wires, illustrating the typical package contents.

2. Setup and Connection

This section details the steps for connecting your GeeekPi I2C 1602 LCD Display Module to common development boards like Raspberry Pi and Arduino.

2.1 Module Pinout

The I2C 1602 LCD module typically has four pins for connection:

- **GND:** Ground connection.
- **VCC:** Power supply (5V).
- **SDA:** Serial Data Line (I2C data).
- **SCL:** Serial Clock Line (I2C clock).



Image: Front view of the LCD displaying "Hello, World!! GeeekPi" and a back view showing the I2C adapter board with GND, VCC, SDA, SCL pins, and a potentiometer.

2.2 Connecting to Raspberry Pi

Follow these steps to connect the LCD module to your Raspberry Pi:

1. Connect the **GND** pin of the LCD module to a Ground pin on the Raspberry Pi.
2. Connect the **VCC** pin of the LCD module to the 5V pin on the Raspberry Pi.
3. Connect the **SDA** pin of the LCD module to GPIO2 (SDA1) on the Raspberry Pi.
4. Connect the **SCL** pin of the LCD module to GPIO3 (SCL1) on the Raspberry Pi.

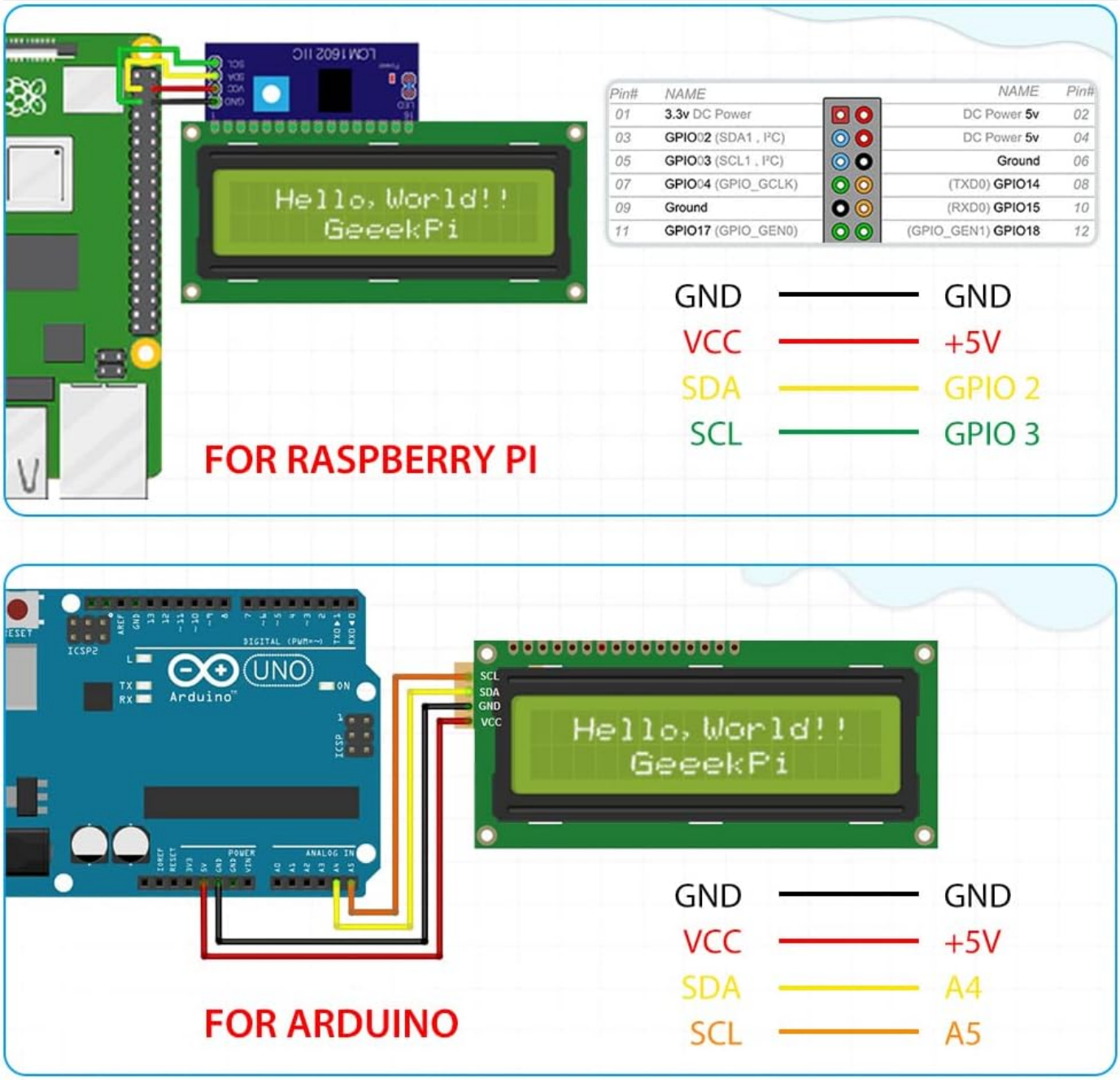


Image: A diagram showing the wiring connections for the I2C 1602 LCD module to both a Raspberry Pi and an Arduino board. For Raspberry Pi, GND connects to GND, VCC to +5V, SDA to GPIO2, and SCL to GPIO3.

2.3 Connecting to Arduino

Follow these steps to connect the LCD module to your Arduino board:

1. Connect the **GND** pin of the LCD module to a Ground pin on the Arduino.
2. Connect the **VCC** pin of the LCD module to the 5V pin on the Arduino.
3. Connect the **SDA** pin of the LCD module to Analog Pin A4 (SDA) on the Arduino Uno.
4. Connect the **SCL** pin of the LCD module to Analog Pin A5 (SCL) on the Arduino Uno.

Refer to the image above for visual guidance on Arduino wiring.

3. Operation

Once connected, the LCD module can be controlled via code on your development board. This section provides general operational guidelines.

3.1 Software Libraries

To communicate with the I2C LCD module, you will need an appropriate library for your development environment. For Arduino, the LiquidCrystal_I2C library is commonly used and can be installed via the Arduino IDE's Library Manager.

3.2 I2C Address Verification

The default I2C address for this module is **0x27**. However, some modules may use **0x3F** or other addresses. If your display does not initialize, use an I2C scanner sketch (available online) to determine the correct address for your specific module.

3.3 Adjusting Contrast and Backlight

The module features a blue potentiometer on the back of the I2C adapter board. This potentiometer controls the display contrast and backlight intensity.

- Turn the potentiometer **clockwise (CW)** to increase contrast/backlight.
- Turn the potentiometer **counter-clockwise (CCW)** to decrease contrast/backlight.

Adjust this until the characters are clearly visible. If the screen appears blank, try adjusting the contrast first, as it might be set too high or too low.

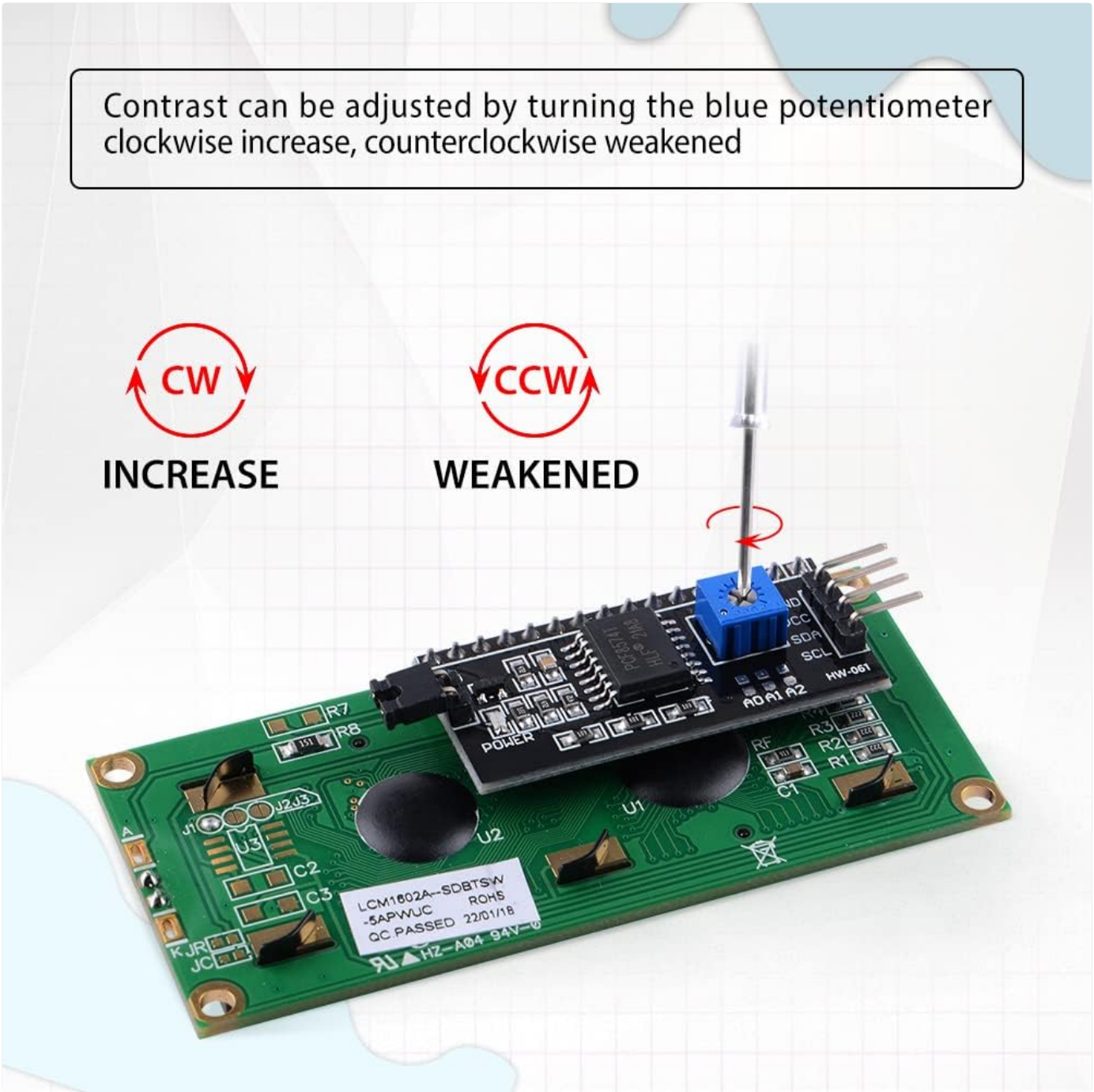


Image: A screwdriver adjusting the blue potentiometer on the back of the I2C LCD module, with arrows indicating clockwise for increase and counter-clockwise for weakened contrast.

4. Maintenance

The GeeekPi I2C 1602 LCD Display Module requires minimal maintenance.

- **Cleaning:** Use a soft, dry cloth to gently wipe the display surface if it becomes dusty. Avoid using liquid cleaners directly on the module.
- **Storage:** Store the module in a dry, anti-static environment when not in use to prevent damage from static discharge or moisture.
- **Handling:** Handle the module by its edges to avoid touching the display area or the electronic components on the PCB.

5. Troubleshooting

If you encounter issues with your I2C 1602 LCD module, consider the following troubleshooting steps:

5.1 Display is Blank or Shows Garbled Characters

- **Check Wiring:** Ensure all four pins (GND, VCC, SDA, SCL) are correctly connected to your development board as per the setup instructions.
- **Adjust Contrast:** Use a small screwdriver to slowly turn the blue potentiometer on the back of the module. The contrast might be set too high or too low, making the characters invisible.
- **Verify I2C Address:** The default address is 0x27, but some modules use 0x3F. Use an I2C scanner sketch to confirm the correct address for your module and update your code accordingly.
- **Power Supply:** Ensure the module is receiving a stable 5V power supply. Insufficient power can lead to erratic behavior or no display.

5.2 No Communication with Development Board

- **Library Installation:** Confirm that the correct I2C LCD library is installed and properly included in your code.
- **I2C Bus Conflict:** If other I2C devices are connected, ensure there are no address conflicts.
- **Pin Assignments:** Double-check that SDA and SCL pins in your code match the physical pins on your development board (e.g., for Raspberry Pi Pico, physical pins 4 and 5 correspond to logical pins GP2 and GP3 for I2C1 SDA/SCL).

5.3 Display Refresh Rate Issues

- **Update Frequency:** Avoid updating the display too frequently, especially if writing to both lines. Updating more than once per second can sometimes lead to unreadable text.

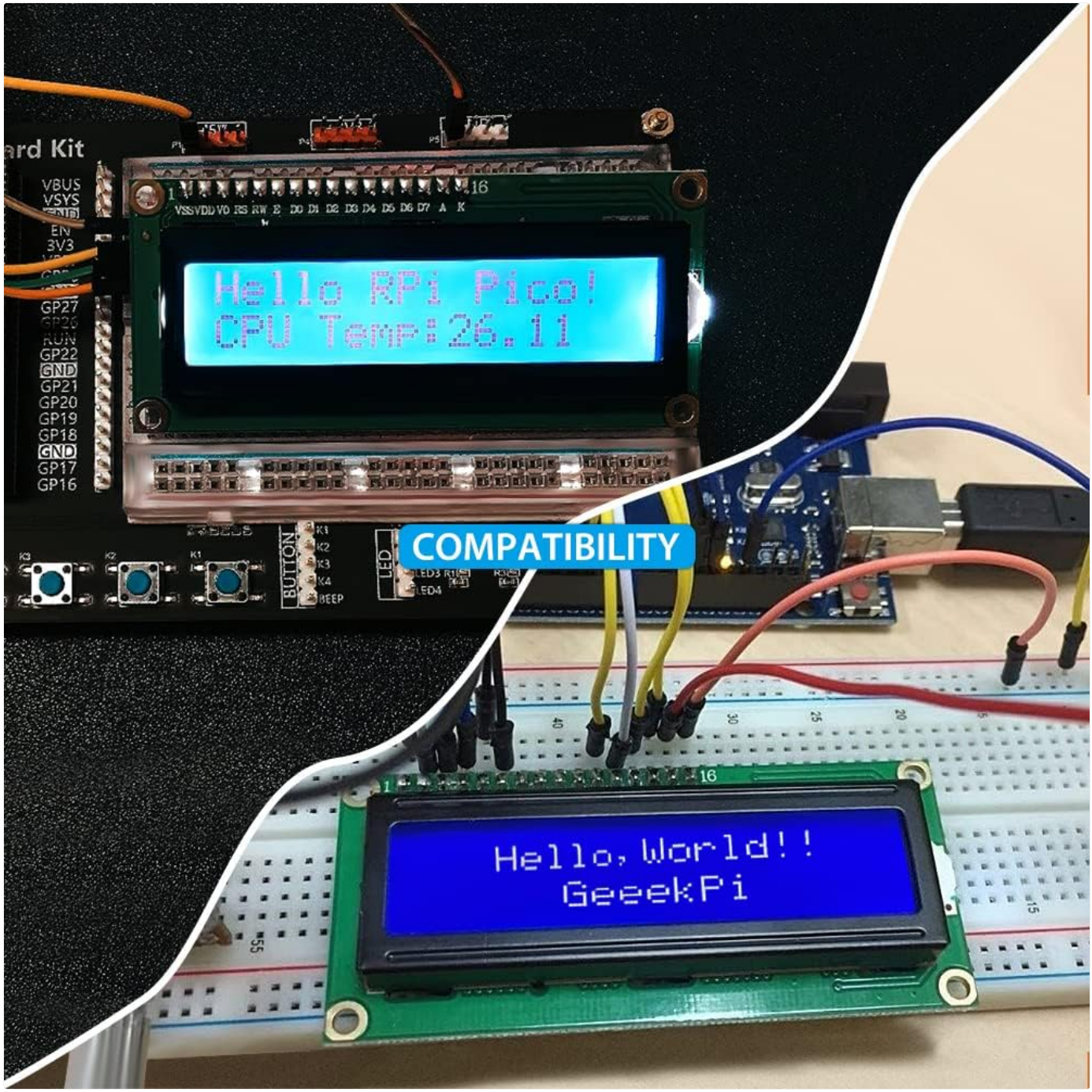


Image: The GeeekPi I2C 1602 LCD module displaying text while connected to a Raspberry Pi Pico and another module connected to an Arduino on a breadboard, demonstrating compatibility.

6. Specifications

Feature	Specification
Display Type	LCD Character Module
Display Format	16 Characters x 2 Lines
Backlight Color	Blue
Interface	I2C (IIC/TWI)
Operating Voltage	5V
Default I2C Address	0x27 (can vary, check with I2C scanner)

Feature	Specification
Module Dimensions (L x W x H)	80mm x 35mm x 11mm (approx. 3.14 x 1.38 x 0.43 inches)
Weight	Approximately 5.3 ounces (for the pack)
Compatibility	Arduino, Raspberry Pi, Tinkerboard, Nano Pi, Banana Pi, STM32

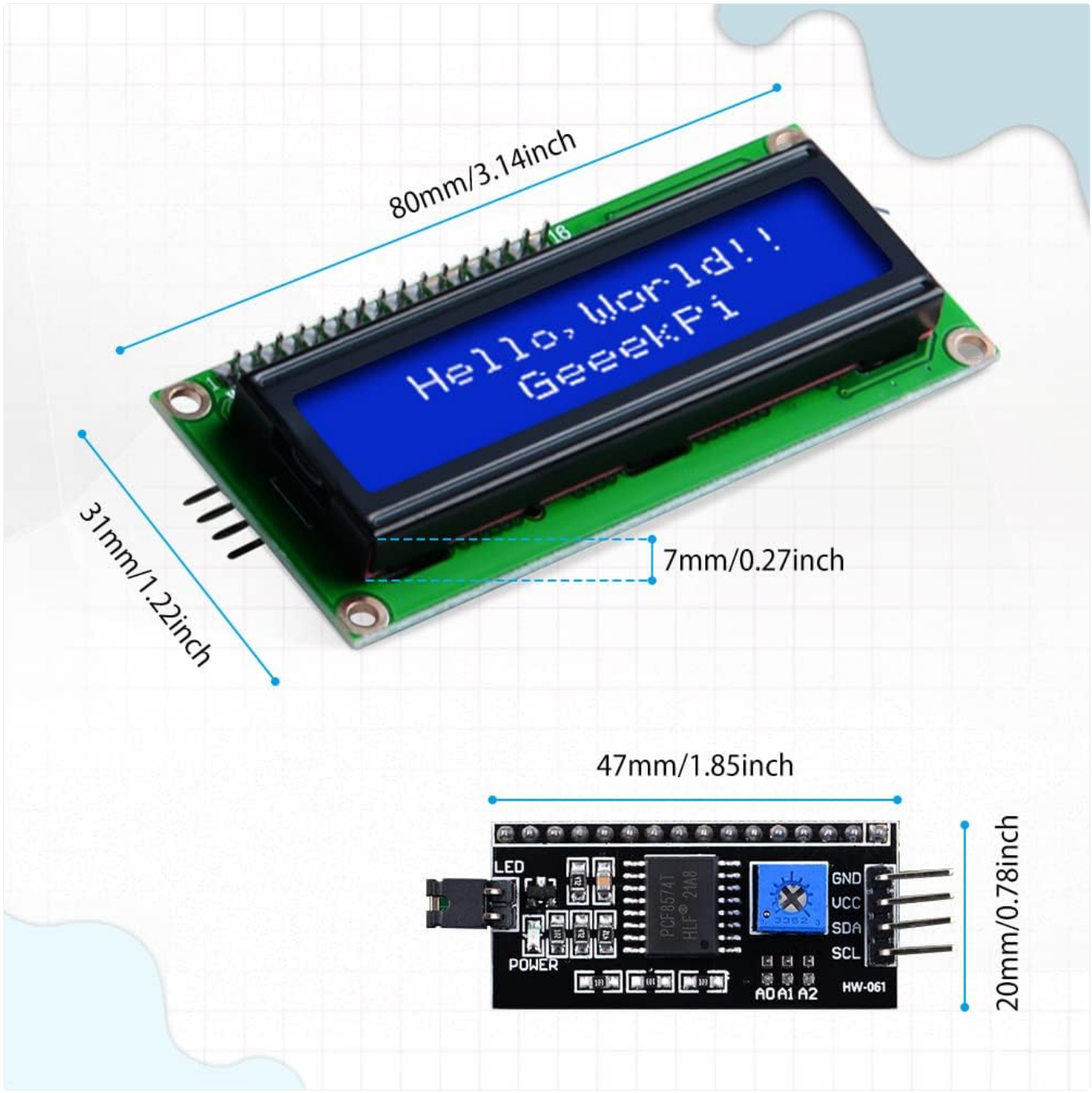


Image: Detailed dimensions of the GeeekPi I2C 1602 LCD module, showing its length, width, and height.

7. Warranty Information

Specific warranty details for the GeeekPi I2C 1602 LCD Display Module are not provided in this manual. Please refer to the product packaging, the seller's website, or contact GeeekPi directly for warranty terms and conditions.

8. Customer Support

For technical assistance, additional resources, or inquiries regarding your GeeekPi I2C 1602 LCD Display Module, please visit the official GeeekPi website or contact their customer support channels. You may also find helpful community forums and online tutorials for Raspberry Pi and Arduino projects involving I2C LCDs.

Ask a question about this manual

Ask about setup, troubleshooting, compatibility, parts, safety, or missing instructions. Manuals+ will review the question and use this page's manual context to help answer it.

Name

Anonymous

Email

you@example.com

After a successful submission, we securely hash the supplied account or form email before sending it to Microsoft Advertising for conversion attribution. Privacy details.

Question

Example: How do I reset this device or fix this error?

Details

Model number, symptoms, what you tried, or the section of the manual you are using.

Submit question