

## Manuals+

[Q & A](#) | [Deep Search](#) | [Upload](#)

[manuals.plus](#) /

- › [sb components](#) /
- › [sb Components Raspberry Pi Pico Breadboard Kit \(SKU20935\) - Instruction Manual](#)

## sb components SKU20935

# sb Components Raspberry Pi Pico Breadboard Kit (SKU20935) - Instruction Manual

Your comprehensive guide to setting up and operating your Raspberry Pi Pico Breadboard Kit.

[Overview](#)   [Components](#)   [Setup](#)   [Operating](#)  
[Instructions](#)   [Maintenance](#)   [Troubleshooting](#)   [Specifications](#)

## 1. PRODUCT OVERVIEW

The sb Components Raspberry Pi Pico Breadboard Kit (SKU20935) is a versatile multi-purpose kit designed for physical computing and prototyping. It integrates the powerful Raspberry Pi Pico microcontroller with a convenient 400-point half-size breadboard, along with essential programmable components like a buzzer, LEDs, and pushbuttons. This kit is ideal for electronics enthusiasts, students, and developers looking to experiment with MicroPython or C/C++ SDK for various projects.

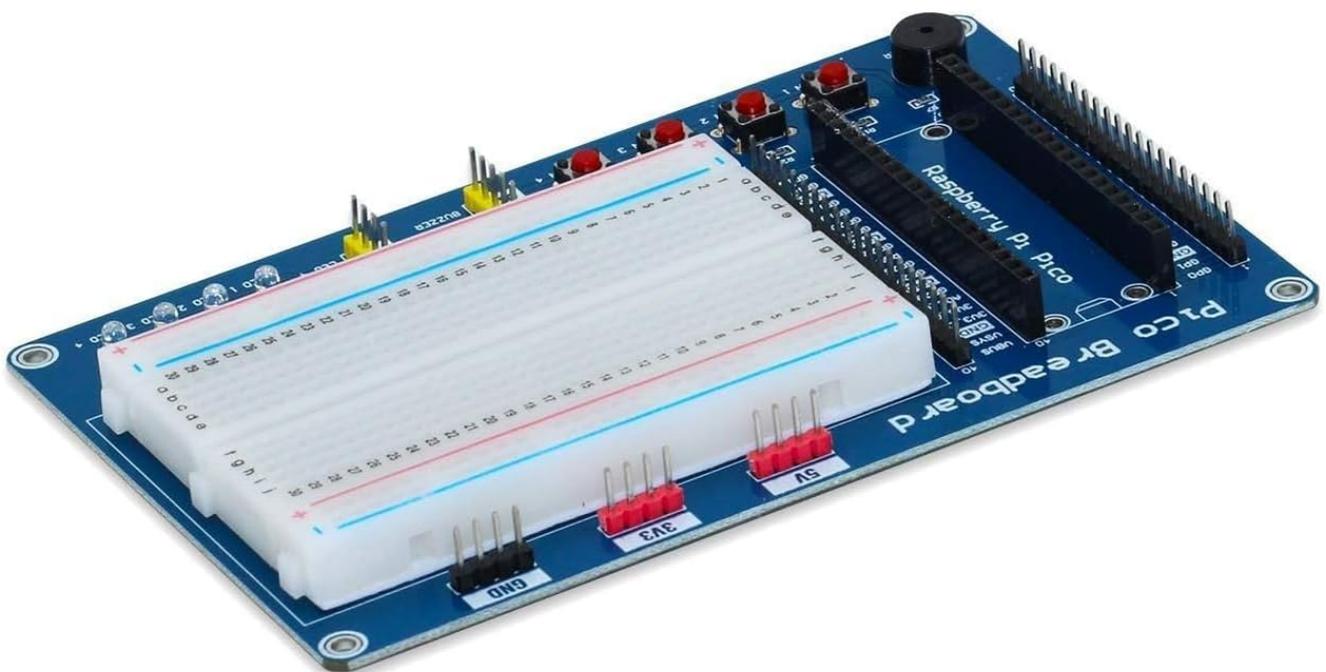


Image: An overview of the sb Components Raspberry Pi Pico Breadboard Kit, featuring the blue PCB with a white breadboard, a Raspberry Pi Pico board, buttons, LEDs, and a buzzer.

## 2. KIT COMPONENTS

---

The sb Components Raspberry Pi Pico Breadboard Kit (SKU20935) includes the following items:

- **Raspberry Pi Pico Microcontroller Board:** Raspberry Pi's first microcontroller board, designed for physical computing.
- **400-point Half-Size Breadboard:** For solderless prototyping and circuit assembly.
- **Programmable Buzzer:** For generating audio feedback.
- **4 Programmable LEDs:** For visual indicators and light-based projects.
- **4 Pushbuttons:** For digital input and user interaction.
- **Dedicated Power Pins:** Convenient 5V, 3V3, and GND pins for easy power distribution.
- **USB Cable:** For connecting the Raspberry Pi Pico to a computer for power and data transfer.
- **Assorted Jumper Wires:** For making connections between components on the breadboard.
- **Additional Sensors and Components:** The kit may include various sensors (e.g., soil moisture, light, sound, ultrasonic, tilt, flame, MPU6050) and other components (e.g., potentiometer, rotary decoder, transistors, servo, display modules, resistors, LEDs) for diverse projects.

## Raspberry Pi Pico Breadboard Kit

Connectivity technology  
USB

Wireless Standard  
bluetooth

Included components  
Pico Breadboard Kit

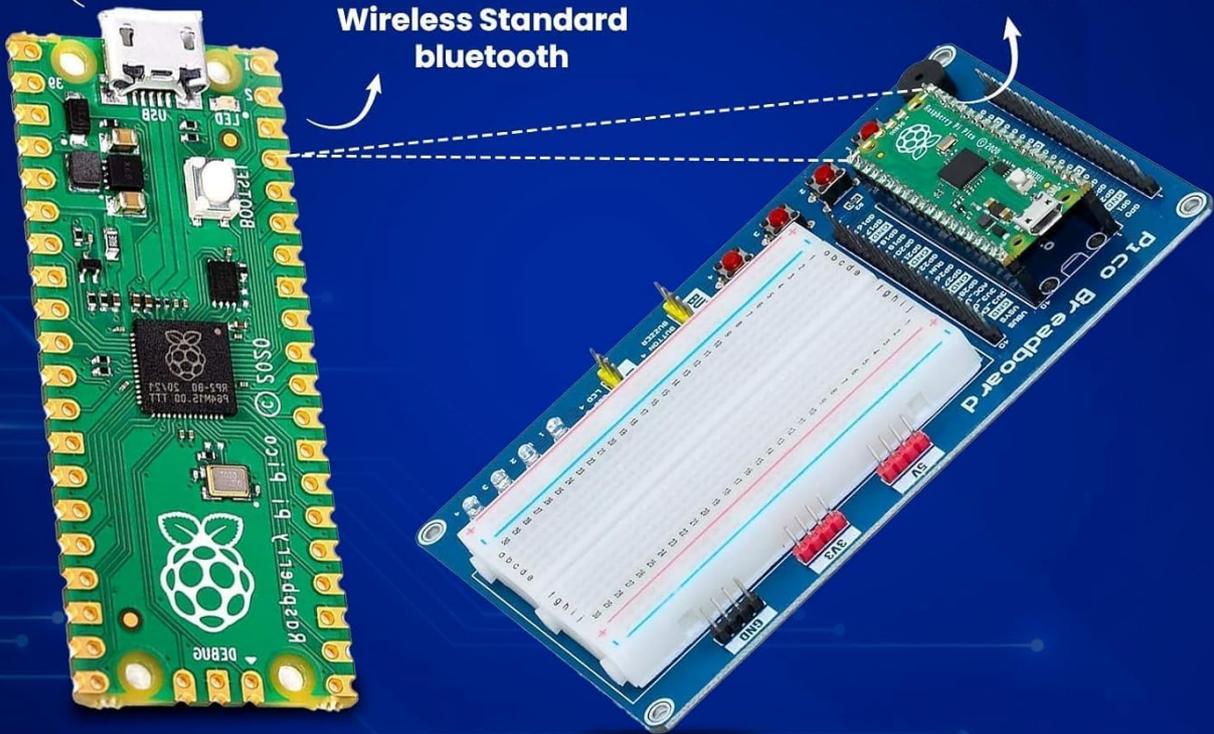


Image: A visual representation of the Raspberry Pi Pico Breadboard Kit, showing the Pico board, the breadboard kit, and various included components like sensors, wires, and display modules.

Your browser does not support the video tag.

Video: An unboxing and overview of the Raspberry Pi Pico Breadboard Kit, showcasing the various components included in the package.

### 3. SETUP INSTRUCTIONS

#### 3.1 Assembling the Pico Breadboard Kit

1. **Insert Raspberry Pi Pico:** Carefully align the Raspberry Pi Pico board with the pin headers on the Pico Breadboard Kit. Ensure all pins are correctly seated into the female headers.
2. **Secure the Pico:** Gently press down on the Pico until it is firmly in place on the breadboard kit.
3. **Attach Breadboard:** Peel the adhesive backing from the 400-point half-size breadboard and firmly attach it to the designated area on the Pico Breadboard Kit.
4. **Connect USB Cable:** Connect the Micro-USB cable to the Raspberry Pi Pico's Micro-USB port and the other

end to your computer or a suitable 5V power source.

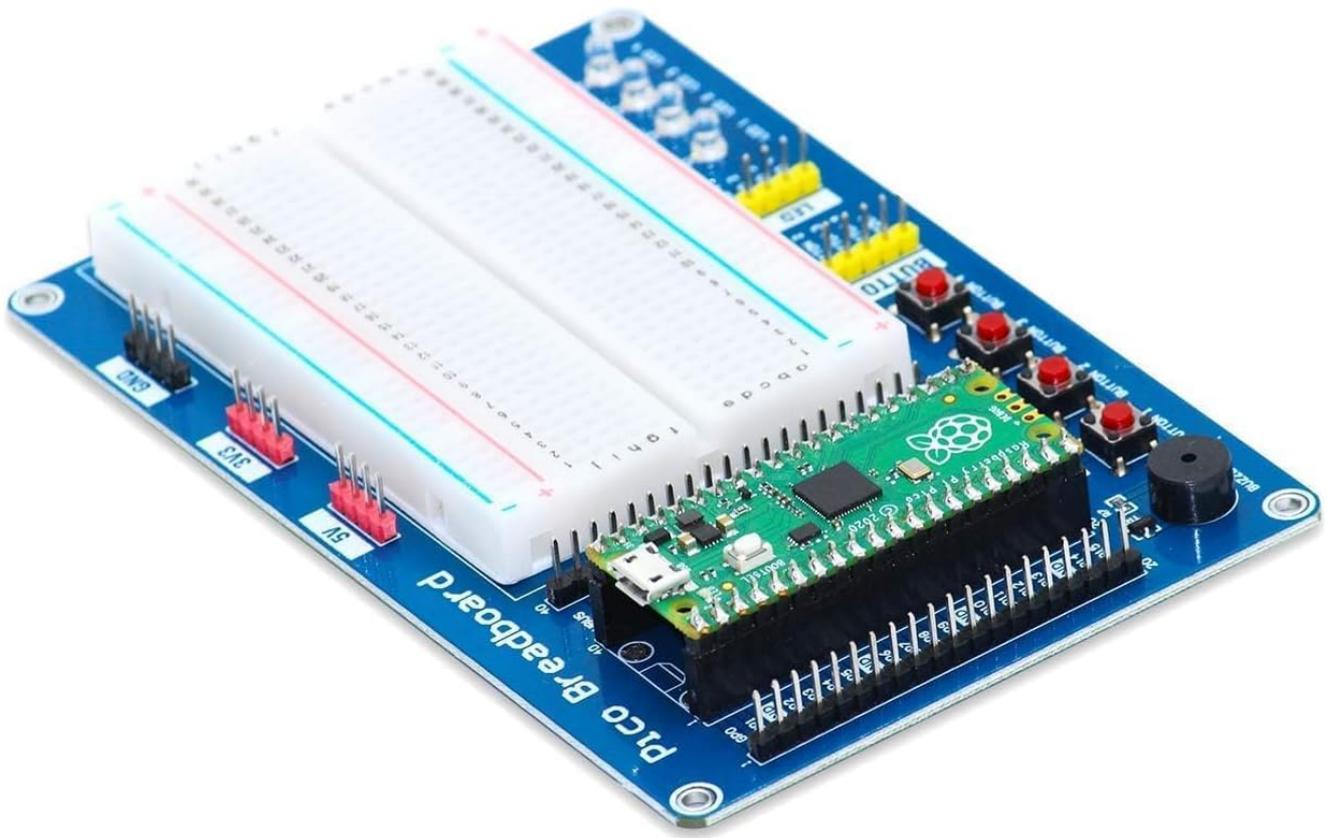


Image: A close-up view of the Raspberry Pi Pico board correctly installed onto the Pico Breadboard Kit, with the white breadboard attached.

### 3.2 Firmware Upload (MicroPython)

To program the Raspberry Pi Pico, you will typically need to upload the MicroPython firmware. Follow these steps:

1. **Enter Bootloader Mode:** Hold down the BOOTSEL button on the Raspberry Pi Pico while connecting it to your computer via the Micro-USB cable. Release the BOOTSEL button after the Pico is connected.
2. **Access RPI-RP2 Drive:** Your computer will recognize the Pico as a mass storage device named 'RPI-RP2'.
3. **Drag and Drop Firmware:** Download the latest MicroPython UF2 file from the official Raspberry Pi Pico website. Drag and drop this UF2 file onto the 'RPI-RP2' drive.
4. **Reboot:** The Pico will automatically reboot, and the 'RPI-RP2' drive will disappear. Your Pico is now running MicroPython.

Your browser does not support the video tag.

Video: A tutorial demonstrating the process of uploading MicroPython firmware to the Raspberry Pi Pico and setting up the development environment.

## 4. OPERATING INSTRUCTIONS

The Pico Breadboard Kit provides a versatile platform for various electronic projects. Here are general guidelines for operation:

### 4.1 Using the Breadboard

The 400-point breadboard allows for solderless prototyping. Components can be easily inserted and removed, making it ideal for experimenting with circuits.

- **Power Rails:** The breadboard has dedicated power rails (red for positive, blue for ground) along the sides. Connect the 5V, 3V3, and GND pins from the Pico Breadboard Kit to these rails as needed using jumper wires.
- **Component Placement:** Insert component leads into the holes. Rows of 5 holes are typically connected horizontally, while the power rails are connected vertically.
- **Jumper Wires:** Use jumper wires to connect different components and the Raspberry Pi Pico's GPIO pins to the breadboard.

## 4.2 Programming with Thonny IDE

Thonny IDE is recommended for programming the Raspberry Pi Pico with MicroPython. It provides a user-friendly interface for writing, running, and debugging code.

1. **Install Thonny:** If not already installed, download and install Thonny IDE on your computer.
2. **Connect Pico:** Connect your Raspberry Pi Pico to your computer via USB.
3. **Configure Thonny:** In Thonny, go to *Tools > Options > Interpreter*. Select "MicroPython (Raspberry Pi Pico)" as the interpreter and choose the correct COM port.
4. **Write and Run Code:** You can now write MicroPython code in the editor and click the "Run" button to execute it on the Pico.

Your browser does not support the video tag.

Video: A comprehensive video tutorial showcasing various projects and functionalities that can be built using the Raspberry Pi Pico and its associated kit components.

## 4.3 Using Onboard Components

The Pico Breadboard Kit features several integrated components for quick prototyping:

- **Programmable Buzzer:** Connect to a GPIO pin on the Pico to generate sounds based on your code.
- **Programmable LEDs:** The four onboard LEDs can be controlled individually via GPIO pins for visual feedback or light patterns.
- **Pushbuttons:** The four pushbuttons provide digital input. Connect them to GPIO pins and program your Pico to respond to button presses.

## 5. MAINTENANCE

- **Keep Dry:** Ensure the kit and its components are kept dry to prevent damage from moisture and corrosion.
- **Cleanliness:** Regularly clean the breadboard and components with a soft, dry cloth or a soft brush to remove dust and debris. Avoid using liquids directly on electronic parts.
- **Proper Storage:** Store all components in the provided storage box or similar organized containers to prevent loss or damage. Keep small parts separated.
- **Handle with Care:** Electronic components are delicate. Avoid excessive force when inserting or removing them from the breadboard. Bending pins can cause permanent damage.
- **Firmware Updates:** Periodically check the official Raspberry Pi website for updated MicroPython firmware or SDKs to ensure optimal performance and access to new features.

## 6. TROUBLESHOOTING

- **Pico Not Detected by Computer:**

- Ensure the Micro-USB cable is fully inserted into both the Pico and the computer.
- Try a different Micro-USB cable; some cables are for charging only and do not support data transfer.
- Verify the Pico is in bootloader mode (BOOTSEL button held down during connection).

- **Code Not Running/Errors in Thonny:**

- Check your wiring connections on the breadboard for any loose or incorrect placements.
- Review your MicroPython code for syntax errors or logical mistakes.
- Ensure Thonny IDE is configured to use the correct MicroPython interpreter and COM port for your Pico.
- Confirm the MicroPython firmware is correctly installed on the Pico.

- **Components Not Responding:**

- Double-check all power and ground connections.
- Verify the correct GPIO pins are being used in your code and match your wiring.
- Test individual components separately if possible to isolate the issue.

- **Buzzer is Constantly On/Buzzing:**

- This can occur if the buzzer pin is directly connected to a constant voltage without proper control. Ensure it's connected to a GPIO pin and controlled by your code.
- Check for short circuits on the breadboard.

## 7. SPECIFICATIONS

Feature	Detail
Product Name	sb Components Raspberry Pi Pico Breadboard Kit
Model Number	SKU20935
Microcontroller	Raspberry Pi Pico (RP2040 microcontroller with 2MB Flash)
Breadboard Type	400 points half-size breadboard
Onboard Components	Programmable buzzer, 4 programmable LEDs, 4 pushbuttons
Power Pins	Dedicated 5V, 3V3, and GND pins
Connectivity	Micro-USB for power and data
Operating System (for programming)	Linux (via Raspberry Pi or other computer)
Programming Languages	C/C++ SDK, MicroPython
Item Weight	6.2 ounces

Feature	Detail
Package Dimensions	9.8 x 9.8 x 1.22 inches