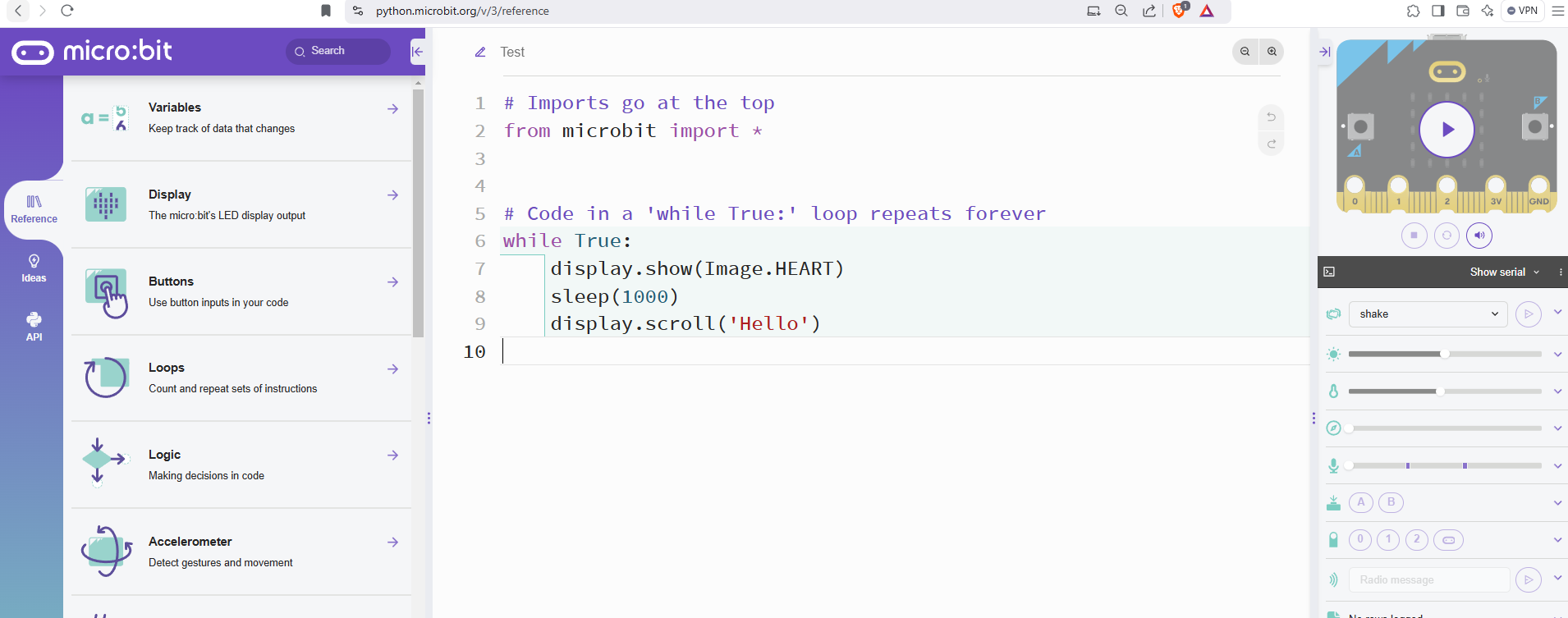
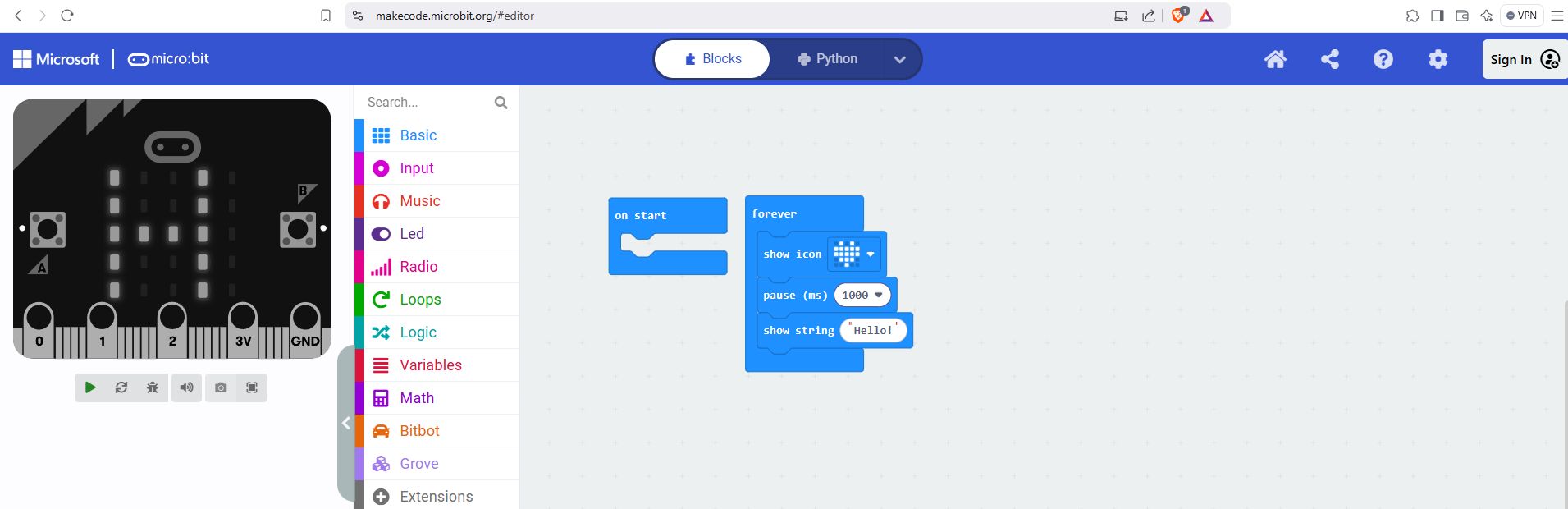
Preps

**Debugging microbit**

* All the debugging should be done using the **default code** as shown below.
* Please **disconnect** the microbit from **MLB basic or MLB+ or breakout board** if connected before the debugging starts.



OR



**Precautions**

1. If the parent is available, ask the parent to follow the debugging steps and the child can stay near and observe.
2. Do not rush or hurry.
3. Do not make the customer uncomfortable. Give them ample time to follow your instructions. It is a new platform for the parents and they might not be having a good knowledge of technical things.
4. If the parent is not available, the child can also follow the instructions.
5. The customer might take time to look for an alternative wire or computer. Give them time and do not rush things..
6. The customer might not even know what microbit is. Keep your hardware handy so that you can show it to them and help them identify the components.
7. Keep the document open for smooth debugging.

**Micro:bit Debugging Document: In-Depth Analysis and Additional Issues**

### **1. Overview**

This document aims to provide a detailed reference for troubleshooting common issues encountered while using the Micro:bit in educational or development environments. The guide includes documented error cases, observed symptoms, step-by-step solutions, and preventive tips to ensure smoother operation.

### **2. Key Guidelines for Debugging**

* Always start with visual inspection (LED indicators, cable connection, PC detection).
* Cross-check on another laptop if the issue persists.
* Confirm the board is powered using a proper micro-USB cable (not a power-only cable).
* Maintain the latest version of firmware and use updated MakeCode or Python editors.
* Use official USB ports (avoid USB hubs when testing faulty boards).

### **3. Documented Cases and Solutions**

#### **Case 1: Yellow Light Not Glowing / Micro:bit Drive Not Visible**

**Symptoms:**

* Yellow LED on Micro:bit does not blink when connected.
* MICROBIT drive does not appear in File Explorer (Windows) or Finder (Mac).

**Causes:**

* USB cable is power-only (no data transfer).
* Port malfunction or poor contact.
* Faulty Micro:bit or USB cable.

**Solution:**

* Use a known working data cable.
* Change the USB port or try on a different laptop.
* Reset Micro:bit by pressing the reset button.
* Use a paperclip to short GND and RST (extreme cases).

#### **Case 2: Code Uploads, But No Output Displayed**

**Symptoms:**

* Hex file uploads successfully.
* No display on Micro:bit LEDs.

**Causes:**

* Bug in the code (infinite loops, missing display calls).
* Logic error: condition never true.

**Solution:**

* Add a basic “Hello!” message at the start to check display.
* Simplify code and test blocks individually.
* Restart Micro:bit and re-upload minimal test code.

#### **Case 3: Heart Icon Only Displayed (Tier 1 Specific)**

**Symptoms:**

* Only heart icon shows even after uploading new code.

**Causes:**

* Micro:bit is stuck in demo mode from factory firmware.

**Solution:**

* Upload a new hex file using MakeCode.
* If issue persists, hard reset the board and upload again.

#### **Case 4: WebUSB Error 503/504**

**Symptoms:**

* Error shown when trying to connect via WebUSB on MakeCode.

**Causes:**

* Browser caching or port issue.

**Solution:**

* Close all tabs using Micro:bit.
* Clear browser cache.
* Restart browser and reconnect.

#### **Case 5: WebUSB Already in Use in Another Tab**

**Symptoms:**

* MakeCode shows error that device is connected in another tab.

**Solution:**

* Close other tabs/windows using WebUSB.
* Refresh MakeCode tab.
* Disconnect and reconnect Micro:bit.

#### **Case 6: OS Error During I2C Communication**

**Symptoms:**

* MicroPython code throws OS Error during sensor communication.

**Causes:**

* Sensor is not connected properly.
* Wrong I2C address used in code.

**Solution:**

* Check sensor wiring (especially SDA/SCL).
* Verify correct address using datasheet or scanner code.
* Add a try/except block in MicroPython to handle error gracefully.

### **4. Firmware Issues and Updating**

**How to Check Firmware Version:**

* Plug in the Micro:bit, open MICROBIT drive.
* Read the DETAILS.TXT file for version info.

**How to Update Firmware:**

1. Go to: https://microbit.org/get-started/user-guide/firmware/
2. Follow the instructions for your board version.
3. Drag the .hex firmware file onto the MICROBIT drive.
4. Wait for flashing to complete.

**Useful Links:**

* Firmware updates: https://microbit.org/get-started/user-guide/firmware/
* Editors: https://makecode.microbit.org or https://python.microbit.org

### **5. Additional Common Issues (Under Research)**

#### **Issue A: Micro:bit Not Visible on File Explorer/Finder**

**Symptoms:**

* Micro:bit not detected on PC/Mac.

**Potential Fixes:**

* Try a different USB cable and port.
* Hold reset button and connect USB to enter bootloader mode.

#### **Issue B: Yellow & Red Lights Both ON**

**Symptoms:**

* Both status LEDs are continuously ON.

**Indication:**

* Board might be in a locked or bricked state.

**Action Plan:**

* Press and hold reset for 10 seconds.
* Reflash firmware if possible.
* If not resolved, consider board replacement.

#### **Issue C: Incorrect Wiring by Students**

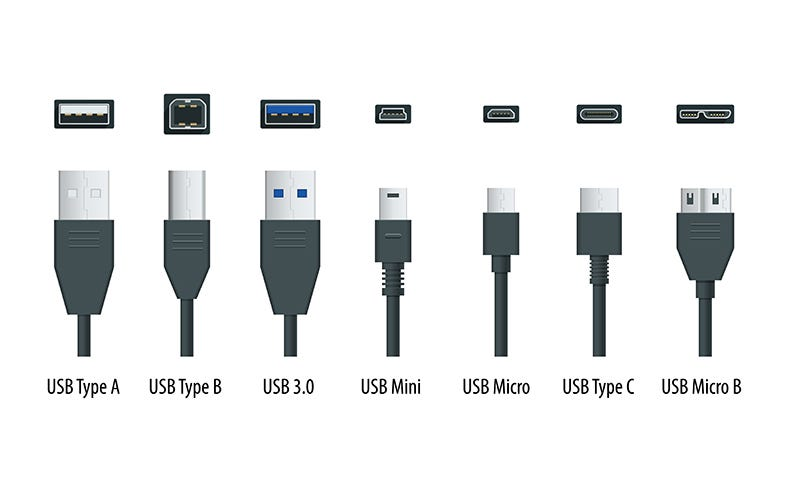
**Symptoms:**

* Code logic fails due to incorrect GPIO usage.

**Fixes:**

* Double-check pin numbers in MakeCode/MicroPython.
* Compare circuit against documentation.
* Use multimeter to test circuit continuity if necessary.

### **6. Recommendations**

* Maintain at least 2 spare working Micro:bits for testing.
* Keep a labeled collection of known-good USB cables.
* Use visuals for explaining cable types and port usage.
* Encourage students to document their circuit setup.
* Track repeated issues in a shared spreadsheet.
* Provide step-by-step handouts to reduce support time.  
  

**Case 2: The code uploads fine and the desired output is not displayed on the microbit.**

1. Click on the save button to download the .hex file. Go to file explorer/ finder-> downloads and copy paste the .hex file in the microbit drive. The output should now be visible and you can change the code in your simulator to show some other icon and try again to upload the code using the simulator now. It should work!! Follow the next step.
2. The microbit kit will have at least two cables(black-USB type A and white-USB type C). Ask the user to look for another cable in the kit (stage 1 kit) and try to fit it in the computer. It is very much possible that the user will be having both types of ports in the laptop/pc. Test the microbit using the new cable and check if the desired output is there or not by transferring the .hex file. If the desired output is there, the problem is solved. If not, tell the user we will send you a replacement microbit within the next 4-5 business days.

NOTE: If the user says that the other wire does not fit in the computer, ask them if they have an adaptor to connect the wire. If they are not sure about the adaptor, you should not force it.

1. One last thing to do is, check with the user if they have another laptop for testing. If they have, then quickly ask the user to connect the microbit on another device using both the cables or the one cable that connects. Check if the .hex file is transferring or not. If it does, the code should upload. If it does not, tell the user that we will send a replacement microbit to you within the next 4-5 business days.

**Case 3: Specific to Tier 1. The test code(as shown at the top) uploads fine, but only a heart icon is displayed on the microbit and it does not do anything else. It should display “hello” as well in a loop. But that does not happen.**

This is not fixable for now. We are working on it. But still try the following method

1. Click on the save button to download the .hex file. Go to file explorer/ finder-> downloads and copy paste the .hex file in the microbit drive. The output should now be visible and you can change the code in your simulator to show some other icon and try again to upload the code using the simulator now. It should work!! Follow the next step.
2. The microbit kit will have at least two cables(black-USB type A and white-USB type C). Ask the user to look for another cable in the kit (stage 1 kit) and try to fit it in the computer. It is very much possible that the user will be having both types of ports in the laptop/pc. Test the microbit using the new cable and check if the desired output is there or not by transferring the .hex file. If the desired output is there, the problem is solved. If not, tell the user that we will send a replacement microbit to you within the next 4-5 business days.

NOTE: If the user says that the other wire does not fit in the computer, ask them if they have an adaptor to connect the wire. If they are not sure about the adaptor, you should not force it.

1. One last thing to do is, check with the user if they have another laptop for testing. If they have, then quickly ask the user to connect the microbit on another device using both the cables or the one cable that connects. Check if the .hex file is transferring or not. If it does, the code should upload. If it does not, tell the user that we will send a replacement microbit to you within the next 4-5 business days.

**Case 4: Microbit code is not uploading, even after upgrading the firmware , error is 504/503; Web USB error**

Showing both in Chrome/Edge, tried the original microbit cable as well and tried replacing the microbit as well still the same error

Solution: it got fixed after using a different laptop

**Case 5:error WebUSB is being used in another tab/ Web USB Error**

Solution.

1. Click on the save button to download the .hex file. Go to file explorer/ finder-> downloads and copy paste the .hex file in the microbit drive. The output should now be visible.
2. The microbit kit will have at least two cables(black-USB type A and white-USB type C). Ask the user to look for another cable in the kit (stage 1 kit) and try to fit it in the computer. It is very much possible that the user will be having both types of ports in the laptop/pc. Test the microbit using the new cable and check if the desired output is there or not by transferring the .hex file. If the desired output is there, the problem is solved.
3. First make sure both the lights are ON, then open the device manager (windows) and uninstall two drivers (1. WEBUSB and 2.CMSIS DAP drivers ) then unplug the microbit from both the ends, when you plug the microbit again drivers will reinstall and the code will be uploaded through makecode or micropython

**CASE 6: OS ERROR**

OS error will occur when you are writing a code to make mirobit work with devices through I2C,

For ex LCD, Moonrover KIT and the microbit is not connected to theses devices.

Points to Note:

1. Whenever you are sending the code make sure all the connections are correct and microbit is connected to the breakout board (in case of moontinker) and moonrover kit.
2. If still the issue is same , redo all the connections and unplug and plug the microbit again.

### [**https://microbit.org/get-started/user-guide/firmware/**](https://microbit.org/get-started/user-guide/firmware/)

### **Checking your firmware version**

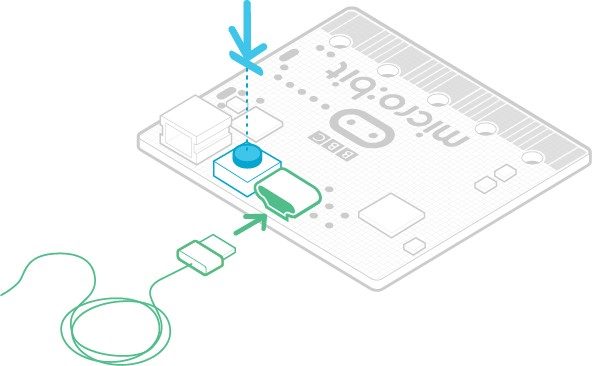
To find out what version of the firmware you have on your micro:bit:

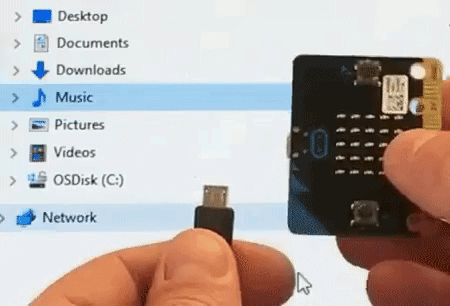
* Plug it in to a computer using the USB cable
* Open the DETAILS.TXT file on the MICROBIT drive
* Look for the number on the line that begins 'Interface Version'. For example, a micro:bit with firmware version 0249 will read:  
    
  Interface Version: 0249

### **How to update the firmware**

If you need to update the firmware to access a new feature or troubleshoot a problem, here is how to do it:

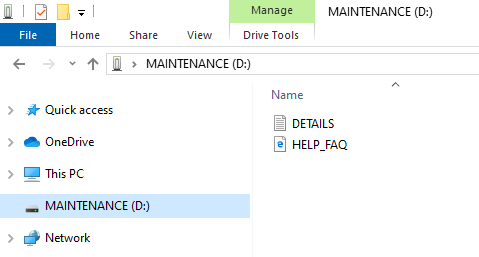
* Disconnect the USB cable and battery pack from the micro:bit.
* Hold the reset button at the back of the micro:bit and plug the USB lead into the micro:bit and a computer. You should see a drive appear in your file manager called MAINTENANCE (instead of MICROBIT).



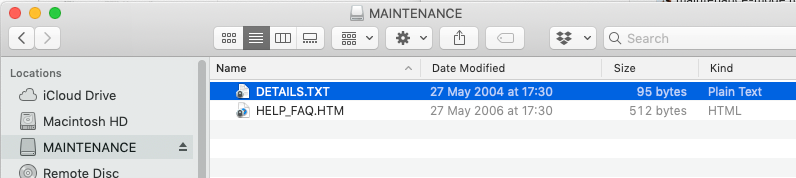


* The MAINTENANCE drive will look like this, depending on your computer:

### **Windows**

****

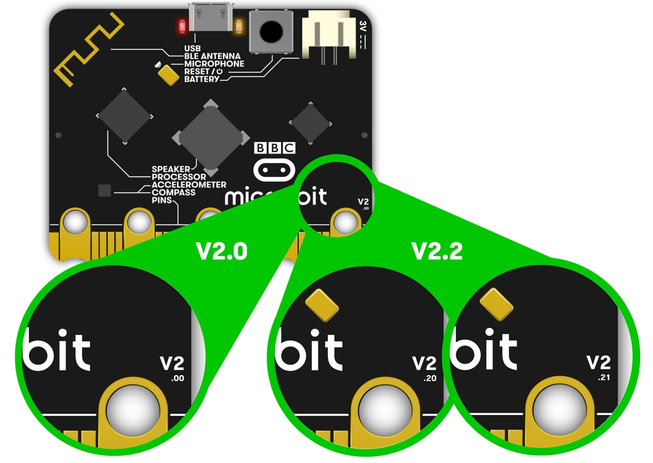
### **macOS**

****

* Download the .hex file appropriate for your version of the micro:bit from this page to your computer. The version number is printed in the bottom right-hand corner on the rear of the device.

#### **micro:bit V2**

If you have a micro:bit V2 (with speaker and microphone), there are two possible versions of the firmware; V2.00 and V2.20/2.21. Please select the firmware appropriate to your board.



* Drag and drop the .hex file you downloaded from this page onto the MAINTENANCE drive and wait for the yellow LED on the back of the device to stop flashing. When the upgrade is completed, the micro:bit will reset, ejecting itself from the computer and re-appear in normal MICROBIT drive mode.

**Link for the firmware v2.21/2.20:**

[**https://drive.google.com/file/d/1MDf5oNpClpNUppN-Spkn\_Rb2puuiKCR9/view?usp=sharing**](https://drive.google.com/file/d/1MDf5oNpClpNUppN-Spkn_Rb2puuiKCR9/view?usp=sharing)

Firmware Link:

<https://microbit.org/get-started/user-guide/firmware/>

Makecode Link: <https://makecode.microbit.org/#>

Micro- Python Editor Link: <https://python.microbit.org/v/3>

.

**Troubleshooting Guidelines**

**General Best Practices:**

* Keep this document readily accessible throughout the debugging process for quick reference.
* Be patient and avoid rushing the customer. Allow sufficient time for them to understand and follow your instructions.
* Recognize that the customer may be unfamiliar with the particular platform and technical concepts.
* If the customer needs to locate alternative equipment (e.g., wires, computer), grant them the necessary time without creating a sense of urgency.
* Have a Microcontroller device on hand to visually aid the customer in identifying components, especially if they are unfamiliar with the hardware.

**Parental Involvement:**

* Ideally, if a parent or guardian is present, request their assistance in performing the debugging steps while the child observes.
* If a parent is unavailable, children can also follow the provided instructions.

**Customer Comfort:**

* Prioritize the customer's comfort throughout the interaction. Avoid making them feel pressured or inadequate due to their technical knowledge.

**Debugging Guide**

**1. Micro: bit**

**Micro: bit Debugging Resources**

**Official Documentation:**

* Micro: bit Firmware Updates [2]
* Micro:bit Troubleshooting (Sphero) [2]
* Setting up Debug Environment (micro: bit v2)[3]

**Case 1: Yellow Light Not Glowing / Drive Not Visible**

**Symptoms:** The Yellow LED does not blink, and the MICROBIT drive is not detected.

**Causes:** Power-only USB cable, faulty port, poor contact, or faulty board/cable.

**Solution:**

o Use a known working data cable.

o Change USB port or try a different laptop.

o Reset the board by pressing the reset button.

**Resource:** [micro:bit Troubleshooting (Sphero)][2]

**Case 2: Code Uploads, But No Output Displayed**

**Symptoms:** Hex file uploads, no output on LEDs.

**Causes:** Bug in code (infinite loops, missing display calls), logic error.

**Solution:**

o Add a basic “Hello!” message at the start.

o Simplify code and test blocks individually.

o Restart and re-upload the minimal test code.

**Resource:** [micro:bit Troubleshooting (Sphero)][2]

**Case 3: Heart Icon Only Displayed**

· **Symptoms:** Only the heart icon appears after uploading new code.

· **Causes:** Board stuck in factory demo mode, firmware issue.

· **Solution:**

o Upload a new hex file using MakeCode.

o Hard reset the board and upload again.

· **Resource:** [micro: bit Support][1]

**Case 4: WebUSB Error 503/504**

· **Symptoms:** Error when connecting via WebUSB on MakeCode.

· **Causes:** Browser caching, port issue.

· **Solution:**

o Close all tabs using Micro: bit.

o Clear browser cache.

o Restart the browser and reconnect.

· **Resource:** [micro: bit Support][1]

**Case 5: WebUSB Already in Use in Another Tab**

· **Symptoms:** Error that the device is connected to another tab.

· **Causes:** Multiple tabs/windows using WebUSB.

· **Solution:**

o Close other tabs/windows using WebUSB.

o Refresh the MakeCode tab.

o Disconnect and reconnect Micro: bit.

· **Resource:** [micro: bit Support][1]

**Case 6: OS Error During I2C Communication**

· **Symptoms:** MicroPython throws an OS error during sensor communication.

· **Causes:** Sensor wiring issue, wrong I2C address.

· **Solution:**

o Check wiring (SDA/SCL).

o Verify correct address using datasheet or scanner code.

o Use a try/except block in MicroPython.

· **Resource:** [micro:bit Troubleshooting (Sphero)][2]

**Case 7: Board Not Visible on File Explorer/Finder**

· **Symptoms:** Micro: bit not detected on PC/Mac.

· **Causes:** Cable/port issue, bootloader mode.

· **Solution:**

o Try a different USB cable and port.

o Hold the reset button and connect the USB to enter bootloader mode.

· **Resource:** [micro: bit Support][1]

**Case 8: Yellow & Red Lights Both ON**

· **Symptoms:** Both status LEDs are continuously on.

· **Causes:** Board locked/bricked.

· **Solution:**

o Press and hold reset for 10 seconds.

o Reflash firmware.

o Consider board replacement if unresolved.

· **Resource:** [micro: bit Support][1]

**2. Arduino Uno**

**Official Resources:**

· [Arduino Uno Wikipedia][4]

· [Common Arduino Issues & Solutions][5]

· [Debugging Arduino Uno with DebugWire][6]

**Case 1: Bootloader Problem**

· **Symptoms:** RX/TX LEDs stuck on, upload fails.

· **Causes:** Bootloader corruption, upload timing issues.

· **Solution:**

o Press reset before upload.

o Reinstall the bootloader if needed.

· **Resource:** [Common Arduino Issues][5]

**Case 2: Wrong COM Port**

· **Symptoms:** Board not detected.

· **Causes:** Driver issue, port conflict.

· **Solution:**

o Select the correct port in the IDE.

o Restart IDE/reconnect board.

· **Resource:** [Common Arduino Issues][5]

**Case 3: Outdated Drivers**

· **Symptoms:** Board not recognized by PC.

· **Causes:** Outdated or missing drivers.

· **Solution:**

o Update drivers via Device Manager.

· **Resource:** [Common Arduino Issues][5]

**Case 4: Errors In Connection**

· **Symptoms:** Errors during connection or upload.

· **Causes:** Faulty cable, loose connection.

· **Solution:**

o Check cable and connections.

o Try a different USB port.

· **Resource:** [Common Arduino Issues][5]

**Case 5: Serial Monitor Not Working**

· **Symptoms:** No output in the serial monitor.

· **Causes:** Incorrect baud rate, code not using Serial.begin().

· **Solution:**

o Set the correct baud rate.

o Ensure Serial.begin() is used.

· **Resource:** [Common Arduino Issues][5]

**Case 6: Servo Problems**

· **Symptoms:** The Servo motor does not start.

· **Causes:** Insufficient current, voltage drop.

· **Solution:**

o Use external power for the servo.

o Check wiring and connections.

· **Resource:** [Common Arduino Issues][5]

**Case 7: Grounding Errors**

· **Symptoms:** Unstable behavior, resets.

· **Causes:** Poor grounding, floating pins.

· **Solution:**

o Ensure proper grounding.

o Connect unused pins to ground if needed.

· **Resource:** [Common Arduino Issues][5]

**Case 8: Programming Errors**

· **Symptoms:** Code upload fails, unexpected behavior.

· **Causes:** Syntax errors, logic errors.

· **Solution:**

o Check code for errors.

o Test code in small blocks.

· **Resource:** [Common Arduino Issues][5]

**Case 9: Debugging Not Supported**

· **Symptoms:** "Debugging is not supported by 'Arduino Uno'" in the IDE.

· **Causes:** ATmega328P does not support native debugging in Arduino IDE 2.

· **Solution:**

o Use DebugWire with the compatible debugger (e.g., Adafruit CH552 QT Py).

o Modify fuses to enable debugging (advanced).

· **Resource:** [Debugging Arduino Uno with DebugWire][6]

**3. Raspberry Pi Pico**

**Official Resources:**

· [Raspberry Pi Pico Wiki (Waveshare)][7]

· [Debugging Raspberry Pi Pico on Windows][8]

**Case 1: Board Not Detected**

· **Symptoms:** Pico is not recognized by PC.

· **Causes:** Driver issue, USB port/cable problem.

· **Solution:**

o Try a different port/cable.

o Check for driver updates.

o Reset board.

· **Resource:** [Raspberry Pi Pico Wiki][7]

**Case 2: Code Upload Failure**

· **Symptoms:** Error during upload.

· **Causes:** Wrong board selected, code errors.

· **Solution:**

o Select the correct board in the IDE.

o Check code for errors.

o Reset the board before upload.

· **Resource:** [Raspberry Pi Pico Wiki][7]

**Case 3: No Output on Pins**

· **Symptoms:** No response from GPIO.

· **Causes:** Wrong PIN, wiring error.

· **Solution:**

o Check wiring and verify PINs.

o Test with simple LED blink code.

· **Resource:** [Raspberry Pi Pico Wiki][7]

**Case 4: Arduino Labs MicroPython Issues**

· **Symptoms:** REPL not working, code not running.

· **Causes:** Syntax errors, interpreter misconfiguration.

· **Solution:**

o Reconnect the board, restart Arduino Labs.

o Check for syntax errors.

· **Resource:** [Raspberry Pi Pico Wiki][7]

**Case 5: Debugging Setup**

· **Symptoms:** Need to debug code at runtime.

· **Causes:** Complex code, need for breakpoints.

· **Solution:**

o Use Visual Studio Code with OpenOCD/pyOCD.

o Set breakpoints and step through code.

· **Resource:** [Debugging Raspberry Pi Pico on Windows][8]

**4. NodeMCU**

**Case 1: Board Not Detected**

**Symptoms:** NodeMCU is not recognized by the PC.

**Causes:** Missing CH340/CP2102 drivers, USB port/cable issue.

**Solution:**

o Install the correct USB drivers.

o Try a different port/cable.

**Resource:** [Arduino IDE NodeMCU Setup Guide (external)][5]

**Case 2: Code Upload Failure**

**Symptoms:** Error during upload.

**Causes:** Wrong board/port selected, code errors.

**Solution:**

o Select the correct board and port in the IDE.

o Check code for errors.

o Reset the board before upload.

**Resource:** [Arduino IDE NodeMCU Setup Guide (external)][5]

**Case 3: Wi-Fi Connection Issues**

**Symptoms:** Unable to connect to the network.

**Causes:** Incorrect SSID/password, network interference.

**Solution:**

o Check SSID/password.

o Ensure Wi-Fi is enabled.

o Test with basic example.

**Resource:** [NodeMCU Official Documentation (external)]

**Case 4: Serial Monitor Issues**

**Symptoms:** No output in the serial monitor.

**Causes:** Incorrect baud rate, code not using Serial.begin().

**Solution:**

o Set the correct baud rate.

o Ensure Serial.begin() is used.

**Resource:** [NodeMCU Official Documentation (external)]

**Note:** For NodeMCU, since official resources were not included in your provided links, I have referenced the standard community documentation. Please consult the [NodeMCU official documentation](https://nodemcu.readthedocs.io/en/latest/) for the most accurate and up-to-date information. For the Arduino IDE NodeMCU setup, the [Random Nerd Tutorials guide](https://randomnerdtutorials.com/how-to-install-esp8266-board-arduino-ide/) is widely recommended.

1. [Teaching with micro: bit : Help & Support](https://support.microbit.org/support/solutions/articles/19000070627-where-do-i-find-more-information-on-teaching-with-micro-bit-)

2. <https://sdk.sphero.com/troubleshooting/microbit-troubleshooting>

3. <https://microbit.c272.org/setup/debug-setup-guide/>

4. <https://en.wikipedia.org/wiki/Arduino_Uno>

5. <https://chipwired.com/10-common-arduino-issues-and-how-to-fix-them/>

6. [Debug Arduino Uno/ATmega328P with Adafruit CH552 QT Py and debugWire](https://learn.adafruit.com/debug-arduino-uno-atmega328p-with-adafruit-ch552-qt-py-and-debugwire?view=all)

7. <https://www.waveshare.com/wiki/Raspberry_Pi_Pico>

8. [Debugging the Raspberry Pi Pico on Windows 10 - element14 Community](https://community.element14.com/products/raspberry-pi/b/blog/posts/debugging-the-raspberry-pi-pico-on-windows-10)

Precautions

**Precautions**

1. If the parent is available, ask the parent to follow the debugging steps and the child can stay near and observe.
2. Do not rush or hurry.
3. Do not make the customer uncomfortable. Give them ample time to follow your instructions. It is a new platform for the parents and they might not be having a good knowledge of technical things.
4. If the parent is not available, the child can also follow the instructions.
5. The customer might take time to look for an alternative wire or computer. Give them time and do not rush things..
6. The customer might not even know what microbit is. Keep your hardware handy so that you can show it to them and help them identify the components.
7. Keep the document open for smooth debugging.

Microbit

**Micro:bit Debugging Document: In-Depth Analysis and Additional Issues**

### **1. Overview**

This document aims to provide a detailed reference for troubleshooting common issues encountered while using the Micro:bit in educational or development environments. The guide includes documented error cases, observed symptoms, step-by-step solutions, and preventive tips to ensure smoother operation.

### **2. Key Guidelines for Debugging**

* Always start with visual inspection (LED indicators, cable connection, PC detection).
* Cross-check on another laptop if the issue persists.
* Confirm the board is powered using a proper micro-USB cable (not a power-only cable).
* Maintain the latest version of firmware and use updated MakeCode or Python editors.
* Use official USB ports (avoid USB hubs when testing faulty boards).

### **3. Documented Cases and Solutions**

#### **Case 1: Yellow Light Not Glowing / Micro:bit Drive Not Visible**

**Symptoms:**

* Yellow LED on Micro:bit does not blink when connected.
* MICROBIT drive does not appear in File Explorer (Windows) or Finder (Mac).

**Causes:**

* USB cable is power-only (no data transfer).
* Port malfunction or poor contact.
* Faulty Micro:bit or USB cable.

**Solution:**

* Use a known working data cable.
* Change the USB port or try on a different laptop.
* Reset Micro:bit by pressing the reset button.
* Use a paperclip to short GND and RST (extreme cases).

#### **Case 2: Code Uploads, But No Output Displayed**

**Symptoms:**

* Hex file uploads successfully.
* No display on Micro:bit LEDs.

**Causes:**

* Bug in the code (infinite loops, missing display calls).
* Logic error: condition never true.

**Solution:**

* Add a basic “Hello!” message at the start to check display.
* Simplify code and test blocks individually.
* Restart Micro:bit and re-upload minimal test code.

#### **Case 3: Heart Icon Only Displayed (Tier 1 Specific)**

**Symptoms:**

* Only heart icon shows even after uploading new code.

**Causes:**

* Micro:bit is stuck in demo mode from factory firmware.

**Solution:**

* Upload a new hex file using MakeCode.
* If issue persists, hard reset the board and upload again.

#### **Case 4: WebUSB Error 503/504**

**Symptoms:**

* Error shown when trying to connect via WebUSB on MakeCode.

**Causes:**

* Browser caching or port issue.

**Solution:**

* Close all tabs using Micro:bit.
* Clear browser cache.
* Restart browser and reconnect.

#### **Case 5: WebUSB Already in Use in Another Tab**

**Symptoms:**

* MakeCode shows error that device is connected in another tab.

**Solution:**

* Close other tabs/windows using WebUSB.
* Refresh MakeCode tab.
* Disconnect and reconnect Micro:bit.

#### **Case 6: OS Error During I2C Communication**

**Symptoms:**

* MicroPython code throws OS Error during sensor communication.

**Causes:**

* Sensor is not connected properly.
* Wrong I2C address used in code.

**Solution:**

* Check sensor wiring (especially SDA/SCL).
* Verify correct address using datasheet or scanner code.
* Add a try/except block in MicroPython to handle error gracefully.

### **4. Firmware Issues and Updating**

**How to Check Firmware Version:**

* Plug in the Micro:bit, open MICROBIT drive.
* Read the DETAILS.TXT file for version info.

**How to Update Firmware:**

1. Go to: https://microbit.org/get-started/user-guide/firmware/
2. Follow the instructions for your board version.
3. Drag the .hex firmware file onto the MICROBIT drive.
4. Wait for flashing to complete.

**Useful Links:**

* Firmware updates: https://microbit.org/get-started/user-guide/firmware/
* Editors: https://makecode.microbit.org or https://python.microbit.org

### **5. Additional Common Issues (Under Research)**

#### **Issue A: Micro:bit Not Visible on File Explorer/Finder**

**Symptoms:**

* Micro:bit not detected on PC/Mac.

**Potential Fixes:**

* Try a different USB cable and port.
* Hold reset button and connect USB to enter bootloader mode.

#### **Issue B: Yellow & Red Lights Both ON**

**Symptoms:**

* Both status LEDs are continuously ON.

**Indication:**

* Board might be in a locked or bricked state.

**Action Plan:**

* Press and hold reset for 10 seconds.
* Reflash firmware if possible.
* If not resolved, consider board replacement.

#### **Issue C: Incorrect Wiring by Students**

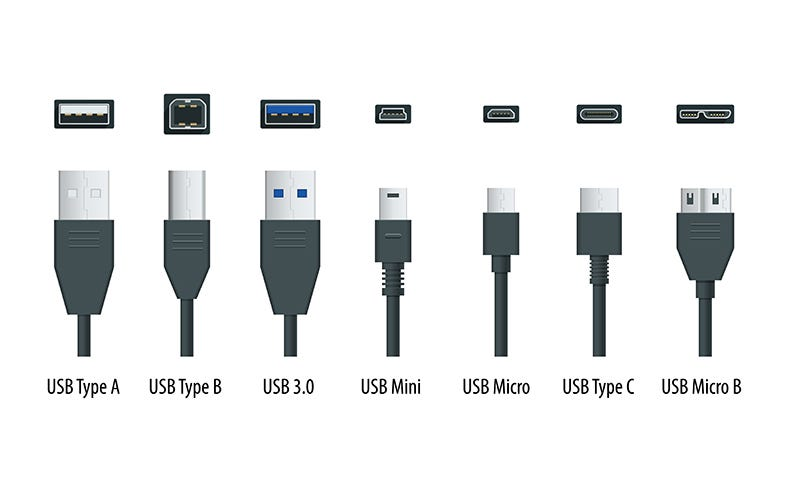
**Symptoms:**

* Code logic fails due to incorrect GPIO usage.

**Fixes:**

* Double-check pin numbers in MakeCode/MicroPython.
* Compare circuit against documentation.
* Use multimeter to test circuit continuity if necessary.

### **6. Recommendations**

* Maintain at least 2 spare working Micro:bits for testing.
* Keep a labeled collection of known-good USB cables.
* Use visuals for explaining cable types and port usage.
* Encourage students to document their circuit setup.
* Track repeated issues in a shared spreadsheet.
* Provide step-by-step handouts to reduce support time.  
  

Case 2

**Case 2: The code uploads fine and the desired output is not displayed on the microbit.**

1. Click on the save button to download the .hex file. Go to file explorer/ finder-> downloads and copy paste the .hex file in the microbit drive. The output should now be visible and you can change the code in your simulator to show some other icon and try again to upload the code using the simulator now. It should work!! Follow the next step.
2. The microbit kit will have at least two cables(black-USB type A and white-USB type C). Ask the user to look for another cable in the kit (stage 1 kit) and try to fit it in the computer. It is very much possible that the user will be having both types of ports in the laptop/pc. Test the microbit using the new cable and check if the desired output is there or not by transferring the .hex file. If the desired output is there, the problem is solved. If not, tell the user we will send you a replacement microbit within the next 4-5 business days.

NOTE: If the user says that the other wire does not fit in the computer, ask them if they have an adaptor to connect the wire. If they are not sure about the adaptor, you should not force it.

1. One last thing to do is, check with the user if they have another laptop for testing. If they have, then quickly ask the user to connect the microbit on another device using both the cables or the one cable that connects. Check if the .hex file is transferring or not. If it does, the code should upload. If it does not, tell the user that we will send a replacement microbit to you within the next 4-5 business days.

Case 3

**Case 3: Specific to Tier 1. The test code(as shown at the top) uploads fine, but only a heart icon is displayed on the microbit and it does not do anything else. It should display “hello” as well in a loop. But that does not happen.**

This is not fixable for now. We are working on it. But still try the following method

1. Click on the save button to download the .hex file. Go to file explorer/ finder-> downloads and copy paste the .hex file in the microbit drive. The output should now be visible and you can change the code in your simulator to show some other icon and try again to upload the code using the simulator now. It should work!! Follow the next step.
2. The microbit kit will have at least two cables(black-USB type A and white-USB type C). Ask the user to look for another cable in the kit (stage 1 kit) and try to fit it in the computer. It is very much possible that the user will be having both types of ports in the laptop/pc. Test the microbit using the new cable and check if the desired output is there or not by transferring the .hex file. If the desired output is there, the problem is solved. If not, tell the user that we will send a replacement microbit to you within the next 4-5 business days.

NOTE: If the user says that the other wire does not fit in the computer, ask them if they have an adaptor to connect the wire. If they are not sure about the adaptor, you should not force it.

1. One last thing to do is, check with the user if they have another laptop for testing. If they have, then quickly ask the user to connect the microbit on another device using both the cables or the one cable that connects. Check if the .hex file is transferring or not. If it does, the code should upload. If it does not, tell the user that we will send a replacement microbit to you within the next 4-5 business days.

Case 4

**Case 4: Microbit code is not uploading, even after upgrading the firmware , error is 504/503; Web USB error**

Showing both in Chrome/Edge, tried the original microbit cable as well and tried replacing the microbit as well still the same error

Solution: it got fixed after using a different laptop

Case 5

**Case 5:error WebUSB is being used in another tab/ Web USB Error**

Solution.

1. Click on the save button to download the .hex file. Go to file explorer/ finder-> downloads and copy paste the .hex file in the microbit drive. The output should now be visible.
2. The microbit kit will have at least two cables(black-USB type A and white-USB type C). Ask the user to look for another cable in the kit (stage 1 kit) and try to fit it in the computer. It is very much possible that the user will be having both types of ports in the laptop/pc. Test the microbit using the new cable and check if the desired output is there or not by transferring the .hex file. If the desired output is there, the problem is solved.
3. First make sure both the lights are ON, then open the device manager (windows) and uninstall two drivers (1. WEBUSB and 2.CMSIS DAP drivers ) then unplug the microbit from both the ends, when you plug the microbit again drivers will reinstall and the code will be uploaded through makecode or micropython

OS ERROR

**CASE 6: OS ERROR**

OS error will occur when you are writing a code to make mirobit work with devices through I2C,

For ex LCD, Moonrover KIT and the microbit is not connected to theses devices.

Points to Note:

1. Whenever you are sending the code make sure all the connections are correct and microbit is connected to the breakout board (in case of moontinker) and moonrover kit.
2. If still the issue is same , redo all the connections and unplug and plug the microbit again.

Microbit Firmware Upgrade Steps

### [**https://microbit.org/get-started/user-guide/firmware/**](https://microbit.org/get-started/user-guide/firmware/)

### **Checking your firmware version**

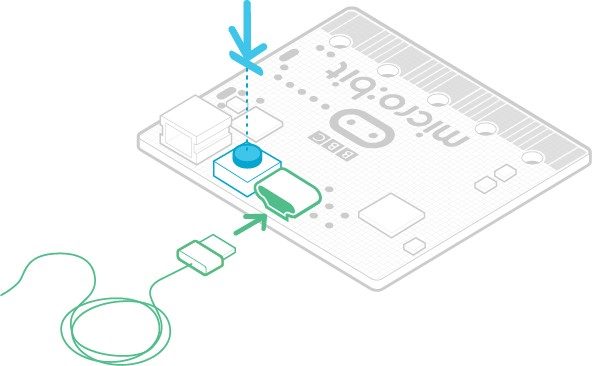
To find out what version of the firmware you have on your micro:bit:

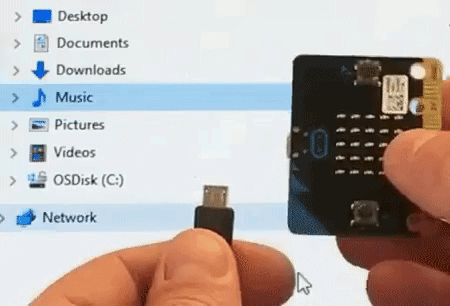
* Plug it in to a computer using the USB cable
* Open the DETAILS.TXT file on the MICROBIT drive
* Look for the number on the line that begins 'Interface Version'. For example, a micro:bit with firmware version 0249 will read:  
    
  Interface Version: 0249

### **How to update the firmware**

If you need to update the firmware to access a new feature or troubleshoot a problem, here is how to do it:

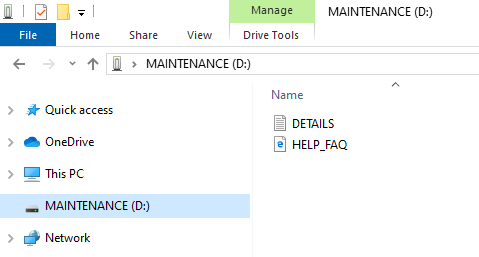
* Disconnect the USB cable and battery pack from the micro:bit.
* Hold the reset button at the back of the micro:bit and plug the USB lead into the micro:bit and a computer. You should see a drive appear in your file manager called MAINTENANCE (instead of MICROBIT).



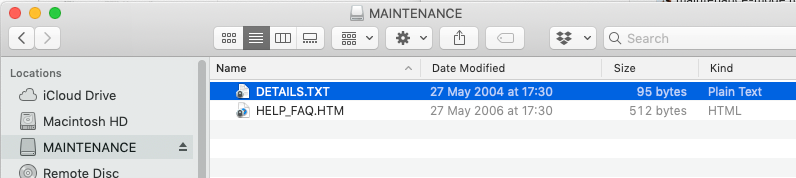


* The MAINTENANCE drive will look like this, depending on your computer:

### **Windows**

****

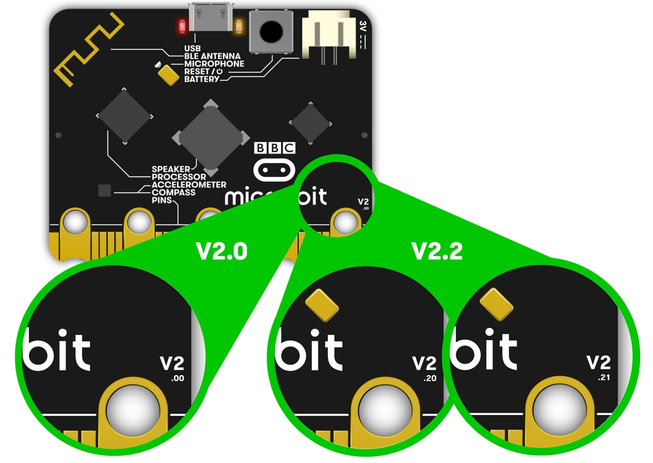
### **macOS**

****

* Download the .hex file appropriate for your version of the micro:bit from this page to your computer. The version number is printed in the bottom right-hand corner on the rear of the device.

#### **micro:bit V2**

If you have a micro:bit V2 (with speaker and microphone), there are two possible versions of the firmware; V2.00 and V2.20/2.21. Please select the firmware appropriate to your board.



* Drag and drop the .hex file you downloaded from this page onto the MAINTENANCE drive and wait for the yellow LED on the back of the device to stop flashing. When the upgrade is completed, the micro:bit will reset, ejecting itself from the computer and re-appear in normal MICROBIT drive mode.

**Link for the firmware v2.21/2.20:**

[**https://drive.google.com/file/d/1MDf5oNpClpNUppN-Spkn\_Rb2puuiKCR9/view?usp=sharing**](https://drive.google.com/file/d/1MDf5oNpClpNUppN-Spkn_Rb2puuiKCR9/view?usp=sharing)

Google Form

**Please fill out the google form at the end of the debugging session**

<https://docs.google.com/forms/d/e/1FAIpQLSeXQpWEqnxzxsOcnTdKYWP9JYda6BvBjB1VazWsKqAZvUjSsA/viewform?usp=sharing>

Firmware Link

Firmware Link:

<https://microbit.org/get-started/user-guide/firmware/>

Makecode Link: <https://makecode.microbit.org/#>

Micro- Python Editor Link: <https://python.microbit.org/v/3>

Debugging Training Video

**Debugging Training Video**

**Link:** [**Video link click here**](https://drive.google.com/file/d/1dnFddYR4IJ47JZBP5SM03wCvRXZc_anR/view?usp=sharing)

Tab 11

Microbit is not visible on drive/finder (what needs to be done )Yellow light and red light both turned ON

Students usually do the connections incorrectly for the microbit and the other components.

Webusb is being used by another tab or device and the code is not being sent

But the microbit is getting detected

New Guide in Progress

.

**Troubleshooting Guidelines**

**General Best Practices:**

* Keep this document readily accessible throughout the debugging process for quick reference.
* Be patient and avoid rushing the customer. Allow sufficient time for them to understand and follow your instructions.
* Recognize that the customer may be unfamiliar with the particular platform and technical concepts.
* If the customer needs to locate alternative equipment (e.g., wires, computer), grant them the necessary time without creating a sense of urgency.
* Have a Microcontroller device on hand to visually aid the customer in identifying components, especially if they are unfamiliar with the hardware.

**Parental Involvement:**

* Ideally, if a parent or guardian is present, request their assistance in performing the debugging steps while the child observes.
* If a parent is unavailable, children can also follow the provided instructions.

**Customer Comfort:**

* Prioritize the customer's comfort throughout the interaction. Avoid making them feel pressured or inadequate due to their technical knowledge.

**Debugging Guide**

**1. Micro: bit**

**Micro: bit Debugging Resources**

**Official Documentation:**

* Micro: bit Firmware Updates [2]
* Micro:bit Troubleshooting (Sphero) [2]
* Setting up Debug Environment (micro: bit v2)[3]

**Case 1: Yellow Light Not Glowing / Drive Not Visible**

**Symptoms:** The Yellow LED does not blink, and the MICROBIT drive is not detected.

**Causes:** Power-only USB cable, faulty port, poor contact, or faulty board/cable.

**Solution:**

o Use a known working data cable.

o Change USB port or try a different laptop.

o Reset the board by pressing the reset button.

**Resource:** [micro:bit Troubleshooting (Sphero)][2]

**Case 2: Code Uploads, But No Output Displayed**

**Symptoms:** Hex file uploads, no output on LEDs.

**Causes:** Bug in code (infinite loops, missing display calls), logic error.

**Solution:**

o Add a basic “Hello!” message at the start.

o Simplify code and test blocks individually.

o Restart and re-upload the minimal test code.

**Resource:** [micro:bit Troubleshooting (Sphero)][2]

**Case 3: Heart Icon Only Displayed**

· **Symptoms:** Only the heart icon appears after uploading new code.

· **Causes:** Board stuck in factory demo mode, firmware issue.

· **Solution:**

o Upload a new hex file using MakeCode.

o Hard reset the board and upload again.

· **Resource:** [micro: bit Support][1]

**Case 4: WebUSB Error 503/504**

· **Symptoms:** Error when connecting via WebUSB on MakeCode.

· **Causes:** Browser caching, port issue.

· **Solution:**

o Close all tabs using Micro: bit.

o Clear browser cache.

o Restart the browser and reconnect.

· **Resource:** [micro: bit Support][1]

**Case 5: WebUSB Already in Use in Another Tab**

· **Symptoms:** Error that the device is connected to another tab.

· **Causes:** Multiple tabs/windows using WebUSB.

· **Solution:**

o Close other tabs/windows using WebUSB.

o Refresh the MakeCode tab.

o Disconnect and reconnect Micro: bit.

· **Resource:** [micro: bit Support][1]

**Case 6: OS Error During I2C Communication**

· **Symptoms:** MicroPython throws an OS error during sensor communication.

· **Causes:** Sensor wiring issue, wrong I2C address.

· **Solution:**

o Check wiring (SDA/SCL).

o Verify correct address using datasheet or scanner code.

o Use a try/except block in MicroPython.

· **Resource:** [micro:bit Troubleshooting (Sphero)][2]

**Case 7: Board Not Visible on File Explorer/Finder**

· **Symptoms:** Micro: bit not detected on PC/Mac.

· **Causes:** Cable/port issue, bootloader mode.

· **Solution:**

o Try a different USB cable and port.

o Hold the reset button and connect the USB to enter bootloader mode.

· **Resource:** [micro: bit Support][1]

**Case 8: Yellow & Red Lights Both ON**

· **Symptoms:** Both status LEDs are continuously on.

· **Causes:** Board locked/bricked.

· **Solution:**

o Press and hold reset for 10 seconds.

o Reflash firmware.

o Consider board replacement if unresolved.

· **Resource:** [micro: bit Support][1]

**2. Arduino Uno**

**Official Resources:**

· [Arduino Uno Wikipedia][4]

· [Common Arduino Issues & Solutions][5]

· [Debugging Arduino Uno with DebugWire][6]

**Case 1: Bootloader Problem**

· **Symptoms:** RX/TX LEDs stuck on, upload fails.

· **Causes:** Bootloader corruption, upload timing issues.

· **Solution:**

o Press reset before upload.

o Reinstall the bootloader if needed.

· **Resource:** [Common Arduino Issues][5]

**Case 2: Wrong COM Port**

· **Symptoms:** Board not detected.

· **Causes:** Driver issue, port conflict.

· **Solution:**

o Select the correct port in the IDE.

o Restart IDE/reconnect board.

· **Resource:** [Common Arduino Issues][5]

**Case 3: Outdated Drivers**

· **Symptoms:** Board not recognized by PC.

· **Causes:** Outdated or missing drivers.

· **Solution:**

o Update drivers via Device Manager.

· **Resource:** [Common Arduino Issues][5]

**Case 4: Errors In Connection**

· **Symptoms:** Errors during connection or upload.

· **Causes:** Faulty cable, loose connection.

· **Solution:**

o Check cable and connections.

o Try a different USB port.

· **Resource:** [Common Arduino Issues][5]

**Case 5: Serial Monitor Not Working**

· **Symptoms:** No output in the serial monitor.

· **Causes:** Incorrect baud rate, code not using Serial.begin().

· **Solution:**

o Set the correct baud rate.

o Ensure Serial.begin() is used.

· **Resource:** [Common Arduino Issues][5]

**Case 6: Servo Problems**

· **Symptoms:** The Servo motor does not start.

· **Causes:** Insufficient current, voltage drop.

· **Solution:**

o Use external power for the servo.

o Check wiring and connections.

· **Resource:** [Common Arduino Issues][5]

**Case 7: Grounding Errors**

· **Symptoms:** Unstable behavior, resets.

· **Causes:** Poor grounding, floating pins.

· **Solution:**

o Ensure proper grounding.

o Connect unused pins to ground if needed.

· **Resource:** [Common Arduino Issues][5]

**Case 8: Programming Errors**

· **Symptoms:** Code upload fails, unexpected behavior.

· **Causes:** Syntax errors, logic errors.

· **Solution:**

o Check code for errors.

o Test code in small blocks.

· **Resource:** [Common Arduino Issues][5]

**Case 9: Debugging Not Supported**

· **Symptoms:** "Debugging is not supported by 'Arduino Uno'" in the IDE.

· **Causes:** ATmega328P does not support native debugging in Arduino IDE 2.

· **Solution:**

o Use DebugWire with the compatible debugger (e.g., Adafruit CH552 QT Py).

o Modify fuses to enable debugging (advanced).

· **Resource:** [Debugging Arduino Uno with DebugWire][6]

**3. Raspberry Pi Pico**

**Official Resources:**

· [Raspberry Pi Pico Wiki (Waveshare)][7]

· [Debugging Raspberry Pi Pico on Windows][8]

**Case 1: Board Not Detected**

· **Symptoms:** Pico is not recognized by PC.

· **Causes:** Driver issue, USB port/cable problem.

· **Solution:**

o Try a different port/cable.

o Check for driver updates.

o Reset board.

· **Resource:** [Raspberry Pi Pico Wiki][7]

**Case 2: Code Upload Failure**

· **Symptoms:** Error during upload.

· **Causes:** Wrong board selected, code errors.

· **Solution:**

o Select the correct board in the IDE.

o Check code for errors.

o Reset the board before upload.

· **Resource:** [Raspberry Pi Pico Wiki][7]

**Case 3: No Output on Pins**

· **Symptoms:** No response from GPIO.

· **Causes:** Wrong PIN, wiring error.

· **Solution:**

o Check wiring and verify PINs.

o Test with simple LED blink code.

· **Resource:** [Raspberry Pi Pico Wiki][7]

**Case 4: Arduino Labs MicroPython Issues**

· **Symptoms:** REPL not working, code not running.

· **Causes:** Syntax errors, interpreter misconfiguration.

· **Solution:**

o Reconnect the board, restart Arduino Labs.

o Check for syntax errors.

· **Resource:** [Raspberry Pi Pico Wiki][7]

**Case 5: Debugging Setup**

· **Symptoms:** Need to debug code at runtime.

· **Causes:** Complex code, need for breakpoints.

· **Solution:**

o Use Visual Studio Code with OpenOCD/pyOCD.

o Set breakpoints and step through code.

· **Resource:** [Debugging Raspberry Pi Pico on Windows][8]

**4. NodeMCU**

**Case 1: Board Not Detected**

**Symptoms:** NodeMCU is not recognized by the PC.

**Causes:** Missing CH340/CP2102 drivers, USB port/cable issue.

**Solution:**

o Install the correct USB drivers.

o Try a different port/cable.

**Resource:** [Arduino IDE NodeMCU Setup Guide (external)][5]

**Case 2: Code Upload Failure**

**Symptoms:** Error during upload.

**Causes:** Wrong board/port selected, code errors.

**Solution:**

o Select the correct board and port in the IDE.

o Check code for errors.

o Reset the board before upload.

**Resource:** [Arduino IDE NodeMCU Setup Guide (external)][5]

**Case 3: Wi-Fi Connection Issues**

**Symptoms:** Unable to connect to the network.

**Causes:** Incorrect SSID/password, network interference.

**Solution:**

o Check SSID/password.

o Ensure Wi-Fi is enabled.

o Test with basic example.

**Resource:** [NodeMCU Official Documentation (external)]

**Case 4: Serial Monitor Issues**

**Symptoms:** No output in the serial monitor.

**Causes:** Incorrect baud rate, code not using Serial.begin().

**Solution:**

o Set the correct baud rate.

o Ensure Serial.begin() is used.

**Resource:** [NodeMCU Official Documentation (external)]

**Note:** For NodeMCU, since official resources were not included in your provided links, I have referenced the standard community documentation. Please consult the [NodeMCU official documentation](https://nodemcu.readthedocs.io/en/latest/) for the most accurate and up-to-date information. For the Arduino IDE NodeMCU setup, the [Random Nerd Tutorials guide](https://randomnerdtutorials.com/how-to-install-esp8266-board-arduino-ide/) is widely recommended.

1. [Teaching with micro: bit : Help & Support](https://support.microbit.org/support/solutions/articles/19000070627-where-do-i-find-more-information-on-teaching-with-micro-bit-)

2. <https://sdk.sphero.com/troubleshooting/microbit-troubleshooting>

3. <https://microbit.c272.org/setup/debug-setup-guide/>

4. <https://en.wikipedia.org/wiki/Arduino_Uno>

5. <https://chipwired.com/10-common-arduino-issues-and-how-to-fix-them/>

6. [Debug Arduino Uno/ATmega328P with Adafruit CH552 QT Py and debugWire](https://learn.adafruit.com/debug-arduino-uno-atmega328p-with-adafruit-ch552-qt-py-and-debugwire?view=all)

7. <https://www.waveshare.com/wiki/Raspberry_Pi_Pico>

8. [Debugging the Raspberry Pi Pico on Windows 10 - element14 Community](https://community.element14.com/products/raspberry-pi/b/blog/posts/debugging-the-raspberry-pi-pico-on-windows-10)

Tab 15

from docx import Document

from langchain.text\_splitter import RecursiveCharacterTextSplitter

from langchain.embeddings import OpenAIEmbeddings

from langchain.vectorstores import FAISS

from langchain.chains import RetrievalQA

from langchain.chat\_models import ChatOpenAI

import os

# Load your OpenAI API key

os.environ["OPENAI\_API\_KEY"] = "sk-proj-45rAz\_ZpkdGw1FQqSwvjxnE2-aEGKjkB23hyYwbEZXsKV6x6NyjxqG2JBH6jouHEkrH2NB9FajT3BlbkFJ8x\_DMLZG6rZDKcBiK-J4zXh7idFr6oD6s9ZbNA3YkqhzbqwRS9T0JRnCArfm7BewLxDxPZh74A"

# Load the DOCX

def load\_docx(path):

doc = Document(path)

return "\n".join([para.text for para in doc.paragraphs if para.text.strip() != ""])

text\_data = load\_docx("Debugging microbit.docx")

# Split the text into chunks

splitter = RecursiveCharacterTextSplitter(chunk\_size=500, chunk\_overlap=50)

chunks = splitter.split\_text(text\_data)

# Create FAISS vector store

embeddings = OpenAIEmbeddings()

vectorstore = FAISS.from\_texts(chunks, embedding=embeddings)

vectorstore.save\_local("microbit\_faiss\_index")

# Load the vector store

vectorstore = FAISS.load\_local("microbit\_faiss\_index", embeddings)

# Create retrieval-based QA chain

retriever = vectorstore.as\_retriever(search\_kwargs={"k": 3})

llm = ChatOpenAI(temperature=0.2, model="gpt-4")

qa\_chain = RetrievalQA.from\_chain\_type(llm=llm, retriever=retriever)

# Ask a question

while True:

query = input("\nAsk your Microbit debugging question (or type 'exit'): ")

if query.lower() == "exit":

break

answer = qa\_chain.run(query)

print(f"\n Answer: {answer}")

API KEY:

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

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

sk-proj-Fi45CVeZjhwu\_oW0eVFsR\_6luSBuzMEGsC3XE5ewgunIPCDHmaH1KxP7OM5lLafsvS2leODABqT3BlbkFJLqex8d1YhNmUCybn2bXVvIavJ2PEcKGyH8N4e2m\_D2\_M8zVWvEnXwmzyc79oceDr-eN400TIoA

hf\_kmILdxGICuNEowxIStkuomDukTSQQWgIGc

export HUGGINGFACEHUB\_API\_TOKEN=hf\_kmILdxGICuNEowxIStkuomDukTSQQWgIGc

export HUGGINGFACEHUB\_API\_TOKEN=hf\_your\_token\_here