Designing a Temperature Sensor Device

Instructions Guide

1 Introduction

In this document, we will provide step-by-step instructions on how to construct a temperature sensor device. The device will be able to accurately measure temperature and humidity in a variety of environments and provide data that can be used for a range of applications.

2 Materials

The following materials will be required to design the temperature sensor device:

- DHT20 (Temperature & Humidity Sensor)
- Raspberry Pi Pico W (Microcontroller)
- Solid Wire
- Wire cutter & Wire stripper
- Hot Glue Gun (Along with sticks)
- Micro USB cable
- Popsicle Stick
- Electrical Tape
- Soldering Iron (Along with accessories)

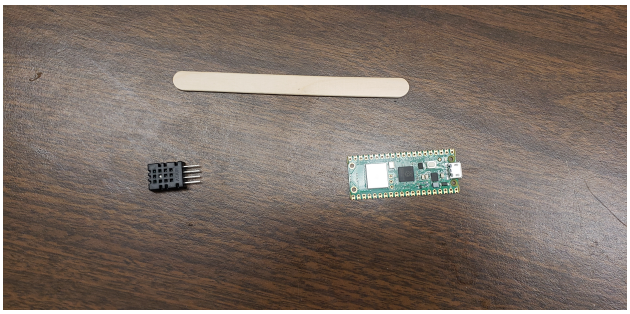

Figure 1: these are the main three components

3 Instructions

3.1 Step 1: Connect the Components

1. Begin by gluing the Raspberry Pi Pico W and the DHT20 to opposite ends of a popsicle stick using a hot glue gun as shown in figure 2, Make sure the pins of the DHT20 are aligned towards the Pico and that the micro USB header of the pico is aligned sigh that it is placed on the far end of the popsicle stick. The vent holes must be facing up such that they are not obstructed by any surface, such as the popsicle stick.

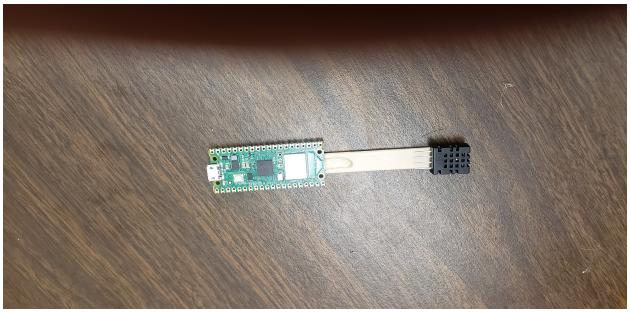

Figure 2: This is the device should look after gluing the pico and DHT20 to the popsicle stick

2. Then cut several solid wire from the wire spool box (I recommended following the color code scheme shown in 3). Make sure they are long enough to connect from the appropriate pin hole on the pico to the pin on the DHT20, as shown in figure 3. You will need to strip the ends of the wires. Now align the wires through to proper pin holes of the pico and to the correct pins on the sensor. You will have to cut off the excess of wire so that the wire does not hang loose. You will then strip that end once excess is cut, as shown in **??**. Make sure excess wire through pin hole is also cut.

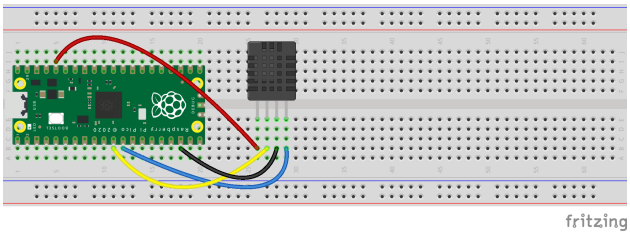

Figure 3: This is how the pin holes on the pico sould be connected to the DHT20 pins. The illustration uses a breadboard.

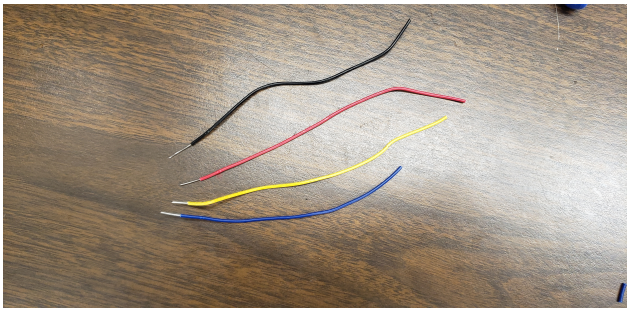

Figure 4: Prepared wires

3. Now we can start the soldering process. First, turn on both the soldering station and the fume fan. Make sure that fan is oriented such that is ventilates fumes away from and not towards you. You will want to want take the sponge and wet it with water. Once wet, place it back in its tray. Then you can turn the temperature knob of soldering station to $350°C$. When operating soldering iron, be sure to practice **EXTREME CUATION** as to where you place your hands- it is very easy to burn yoursself. When you are planning on soldering two components together, you place the soldering iron on the components first to heat them up, and then you apply solder. You do **NOT** heat the solder on the components with soldering iron (see 5 & 6).

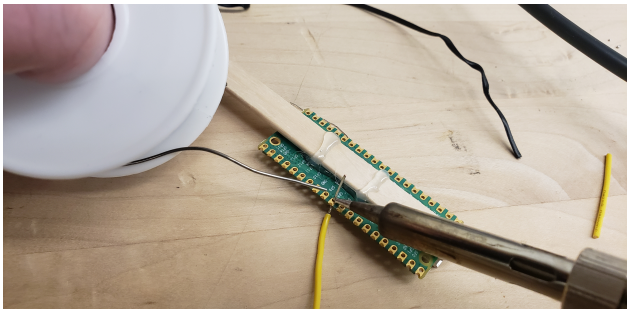

Figure 5: This is how you want to solder

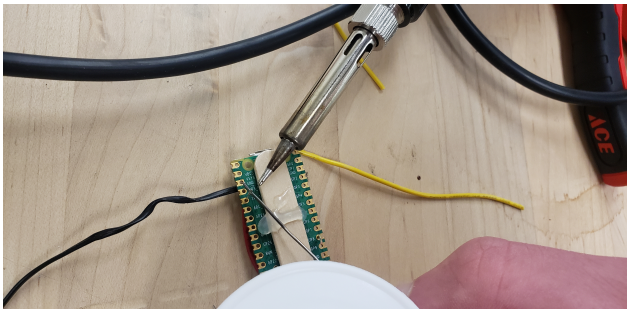

Figure 6: This is incorrect

With all this in mind, you can begin soldering the wire connections between the pico and the DHT20. *Note: it is good pratice to cover the soldering iron tip with solder when done soldering or if you expect there to be a long pause between solderings. This will prevent oxidation of the tip.*

3.2 Step 2: Upload the Code

Next, upload the code to the Temp sensor device. 1. You will have to upload the .UF2 find in the github repository to the pico. You can do this by holding down the "BOOTSEL" button the pico while connecting it to a PC. You can release after it is connected. You can then drag and drop the .UF2 file into the pico.

2. You will have to clone the code from the github repo if not already available. You will also need an IDE (I recommend Thonny) to interface with the device using a computer. One you have Thonny installed, you will have to unplug and replug the pico to the PC. At the top of the Thonny, you will click the tools tab, and then the options... tab in the dropdown menu. Select the interpreter tab in the Thonny options dialog box. In the interpeter box, you will search for and select the "MicroPython (Raspberry Pi Pico)" option from the dropdown menu. For the Port box, you will select the "try to detect port automatically" if it is not already selected. Click OK. Now click the view tab at the top of the Thonny window. Select files in the dropdown menu. You will see two additional panes appear to the left of the window. You will locate the directory of containing the main code to upload on the pico, highlight all the code, right click it, and select the "upload to /" option to upload the files to the pico.

3. Once code is upload, select and run the main code in Thonny. It should return the MAC address of the device. You will need this to register the device to the UTK netowrk. You will go to https://netreg.utk.edu to register the device.

4. Once the device is registered, you can now run the code by runnning it in thonny or replugging it to powered usb port to see if it functional. Then you are done.

3.3 Step 3: Test the Device

Once the device is registered, you can now run the code by runnning it in thonny or replugging it into a powered USB port to see if it functional. Then you are done.