KiZAN IoT Workshop

Lab Exercises

# Lab 3 – Temperature Sensor & Connecting to IoT Hub

## Goal

The goal of this exercise is to build a simple circuit to read from a temperature sensor using a Windows IoT Core device (Raspberry Pi 2, in our case) and an app built and debugged with Visual Studio 2015. We will then connect to an Azure IoT Hub and stream telemetry data for aggregation and reporting. Lastly, the average temperature from all of our temperature sensors will be broadcasted, using an Azure Service Bus, to all of our devices.

## Requirements

Prior to starting this exercise, please ensure that you meet the following requirements. This exercise assumes that you have previously completed Labs 1 & 2.

This lab assumes that you have previously created the Azure IoT Hub and registered your device with this IoT Hub. You will need both the IoT Hub Hostname and the Device Key for your registered device.

### Hardware

Please verify that you have the following components:

1. Raspberry Pi 2 Model B w/ EDIMAX Wifi Adapter
2. Wall Adapter Power Supply
3. Raspberry Pi GPIO Ribbon Cable
4. SparkFun Pi Wedge Breakout Board
5. Breadboard
6. (1) TMP36 Temperature Sensor
7. (1) MCP3008 Analog to Digital Converter
8. (1) 0.1uF ceramic capacitor
9. Jumper Wires M/M (as required)

### Software

Please verify that you have the following software installed on your Windows 10 laptop:

1. Visual Studio 2015 Community Edition (or greater)

<https://www.visualstudio.com/vs-2015-product-editions>

1. Windows 10 IoT Core Dashboard

<https://developer.microsoft.com/en-us/windows/iot/downloads>

### Networking

To communicate with the Raspberry Pi 2, you must be connected to the following Wifi network:

|  |  |
| --- | --- |
| SSID | Password |
| Iotlab | p@ssw0rd |

## Part 1 – Circuit Assembly

1. Carefully attach the SparkFun Pi Wedge Breakout Board to the Breadboard, ensuring that the pins on the Pi Wedge align with the pin holes on the Breadboard.

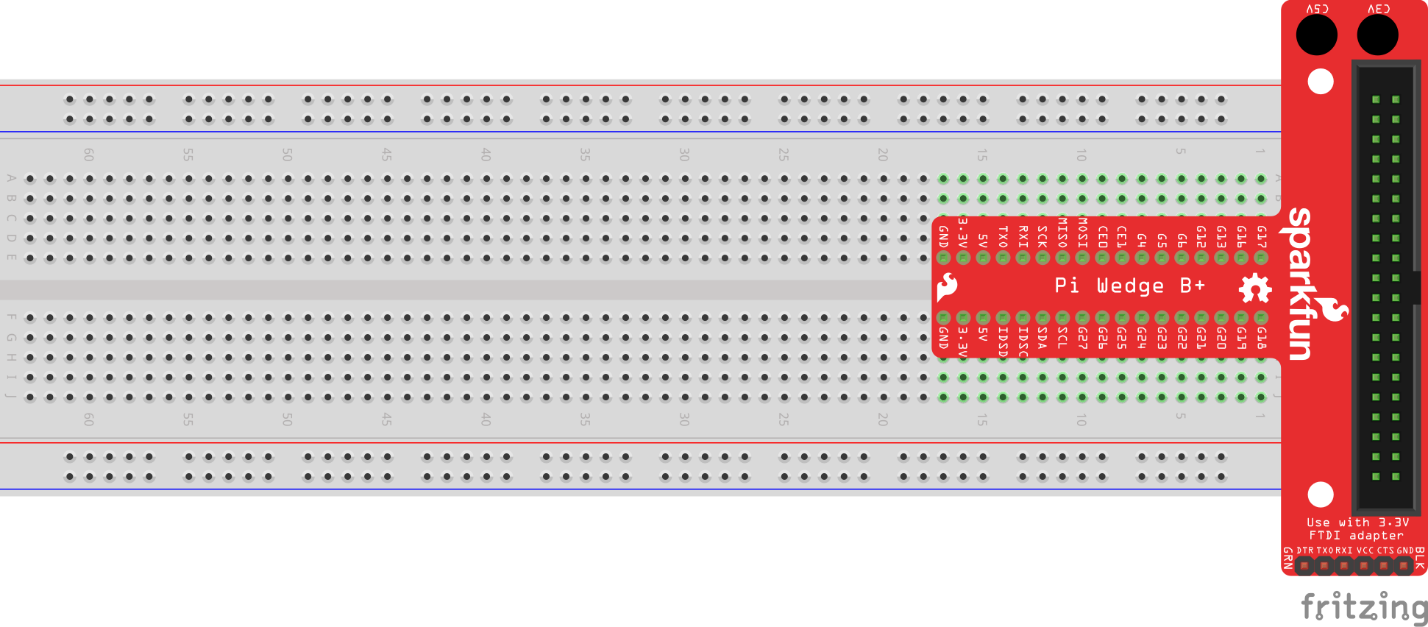


Figure 1 - SparkFun Pi Wedge connected to the Breadboard

1. Using (2) Black Jumper Wires, connect the 3.3V and GND pins on the SparkFun Pi Wedge to the positive (\*) and negative (-) rails on the side of the breadboard, respectively.

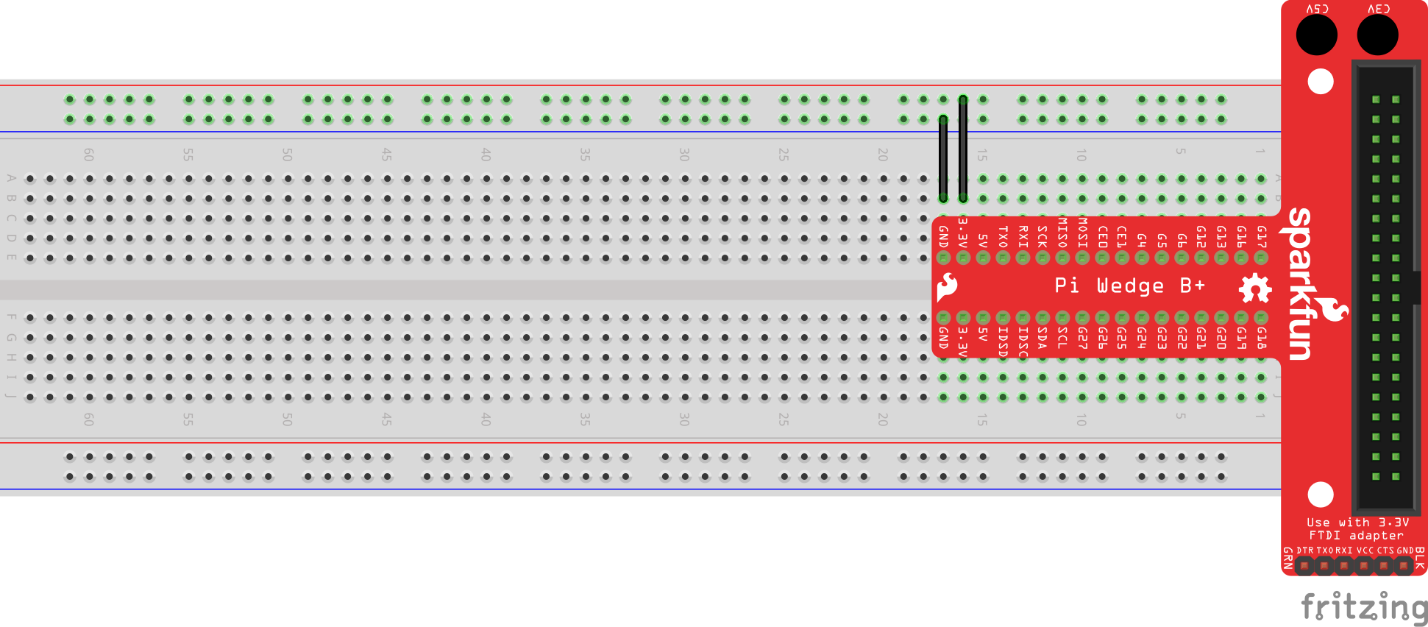


Figure 2 - 3.3V and GND connected to the Breadboard

1. Carefully place the MCP3008 Analog to Digital Converter on the breadboard. The dimple on the top of the chip should be located on the bottom-left after it is placed on the breadboard.

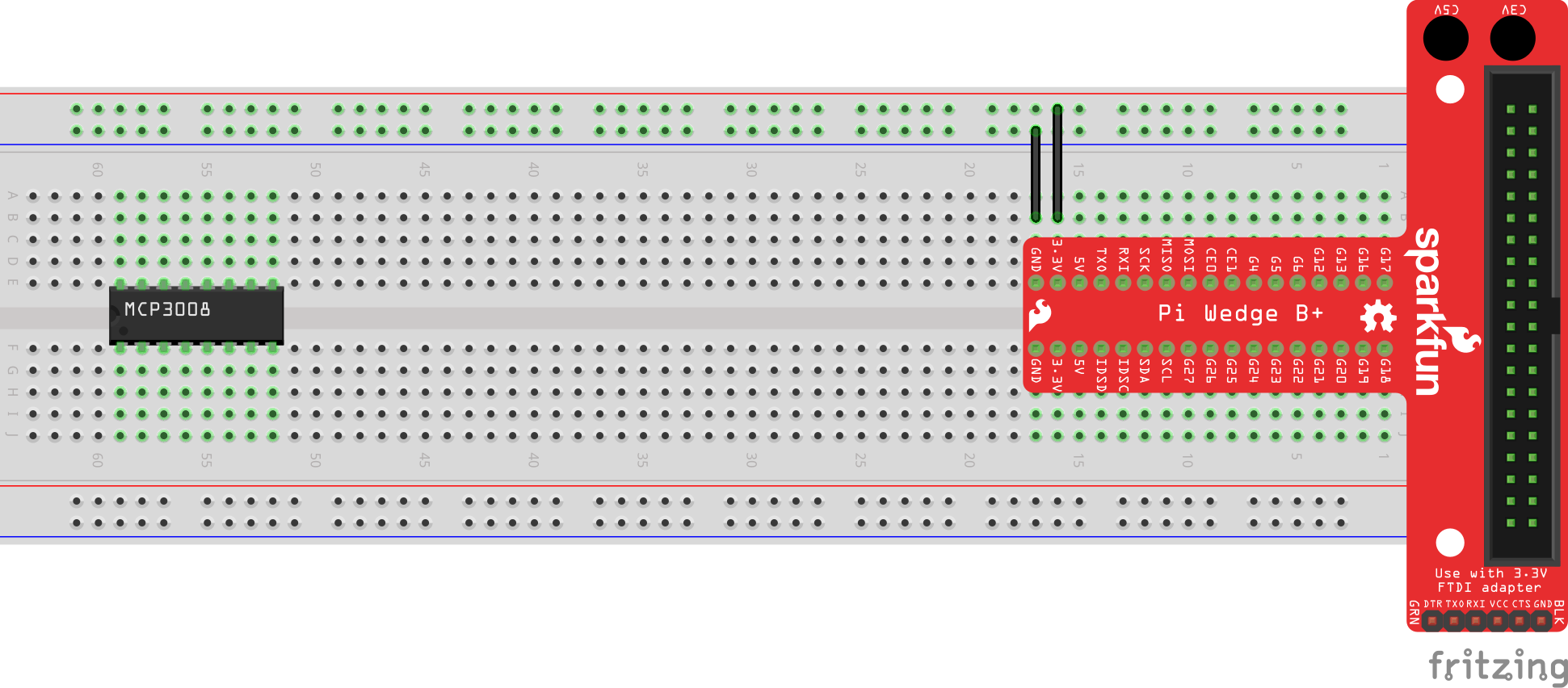
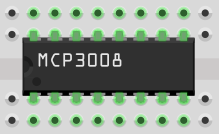


Figure 3 - MCP3008 Analog to Digital Converter

1. Using (2) Red Jumper Wires, connect both pin 16 (VDD) and pin 15 (VREF) of the MCP3008 to the positive (+) rail on the breadboard.

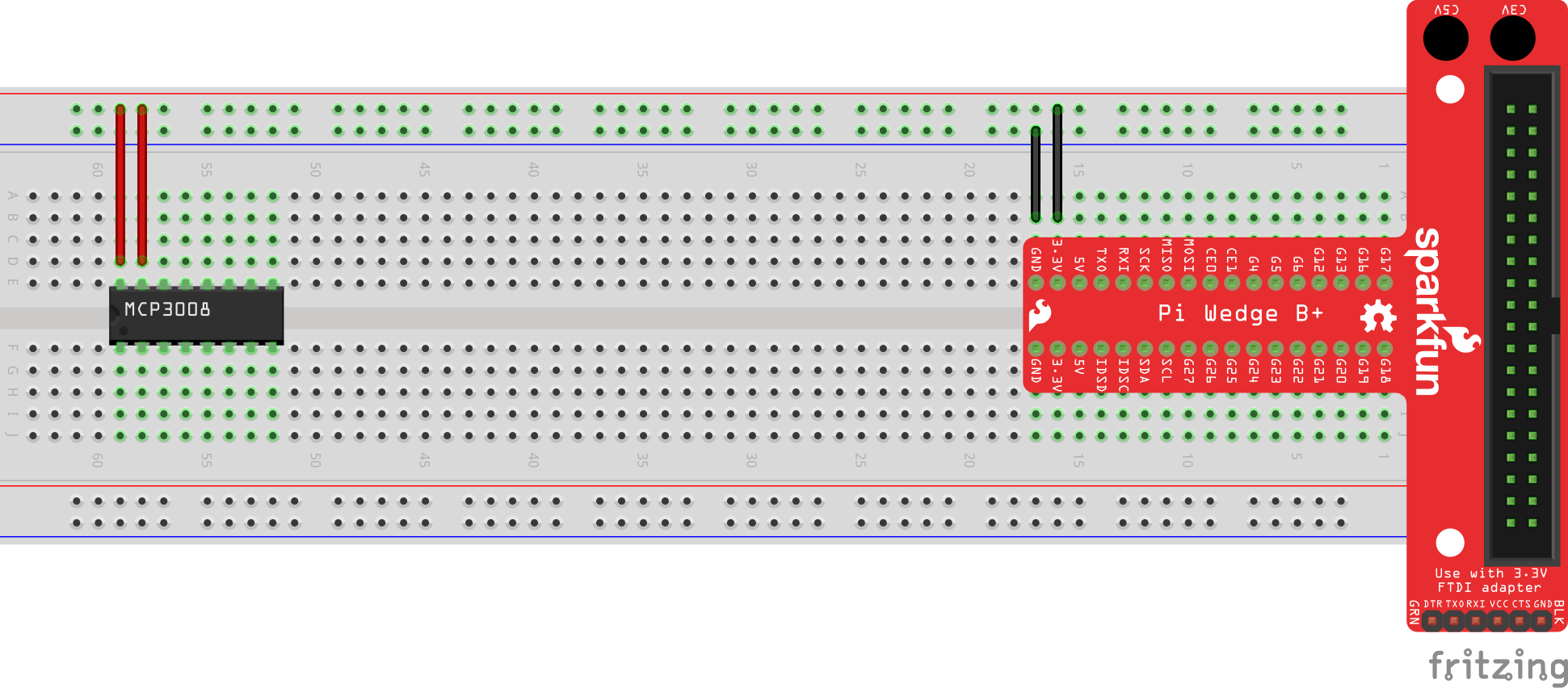


Figure 4 – MCP3008 pins 15 and 16 connected to 3.3V

1. Using (2) Black Jumper Wires, connect both pin 14 (AGND) and pin 9 (DGND) of the MCP3008 to the negative (-) rail on the breadboard.

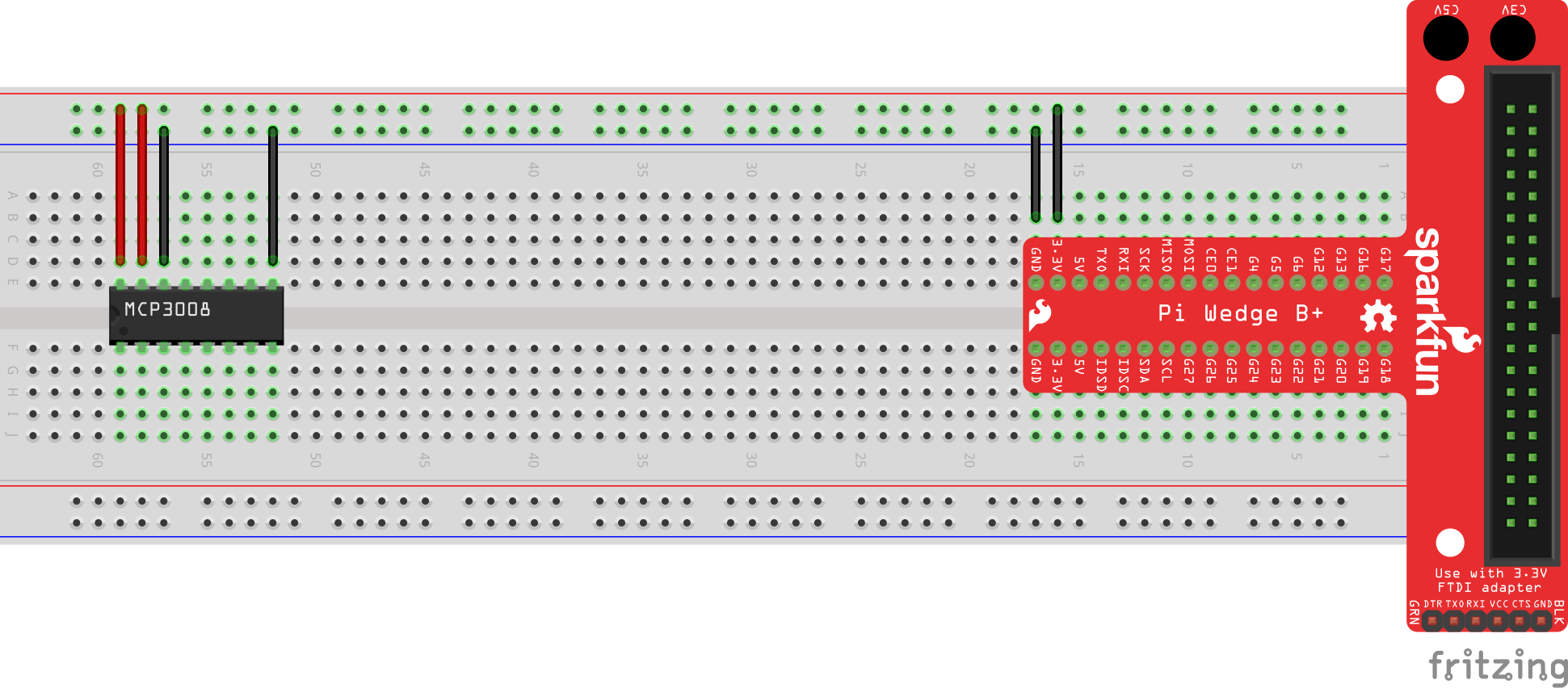


Figure 5 - MCP3008 pins 9 and 14 connected to GND.

1. Using a Yellow Jumper Wire, connect pin 13 (CLK) to the SCK pin of the SparkFun Pi Wedge.

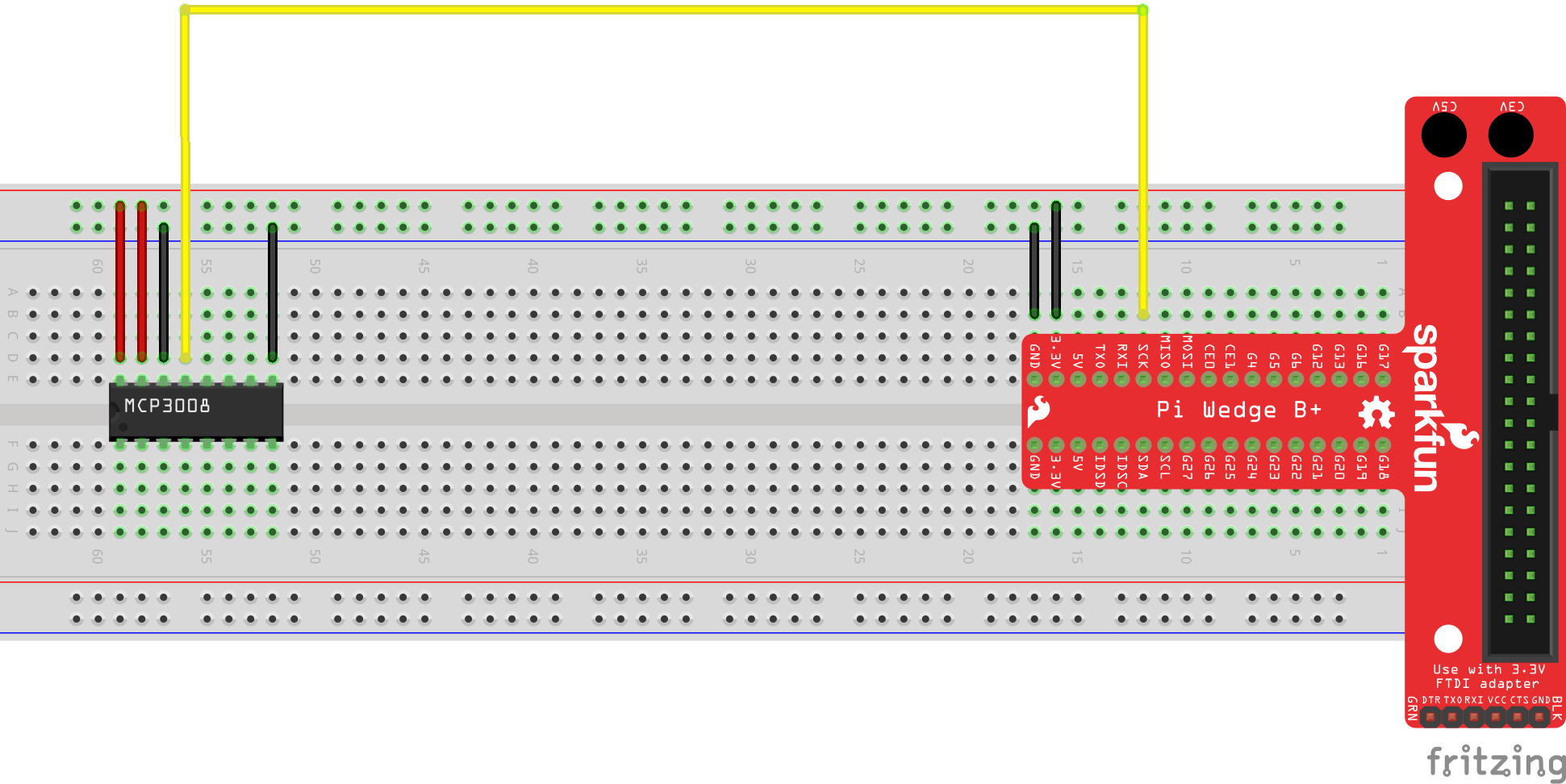


Figure 6 - MCP3008 pin 13 (CLK) connected to SparkFun Pi Wedge pin SCK.

1. Using a White Jumper Wire, connect pin 12 (DOUT) to the MISO pin of the SparkFun Pi Wedge.

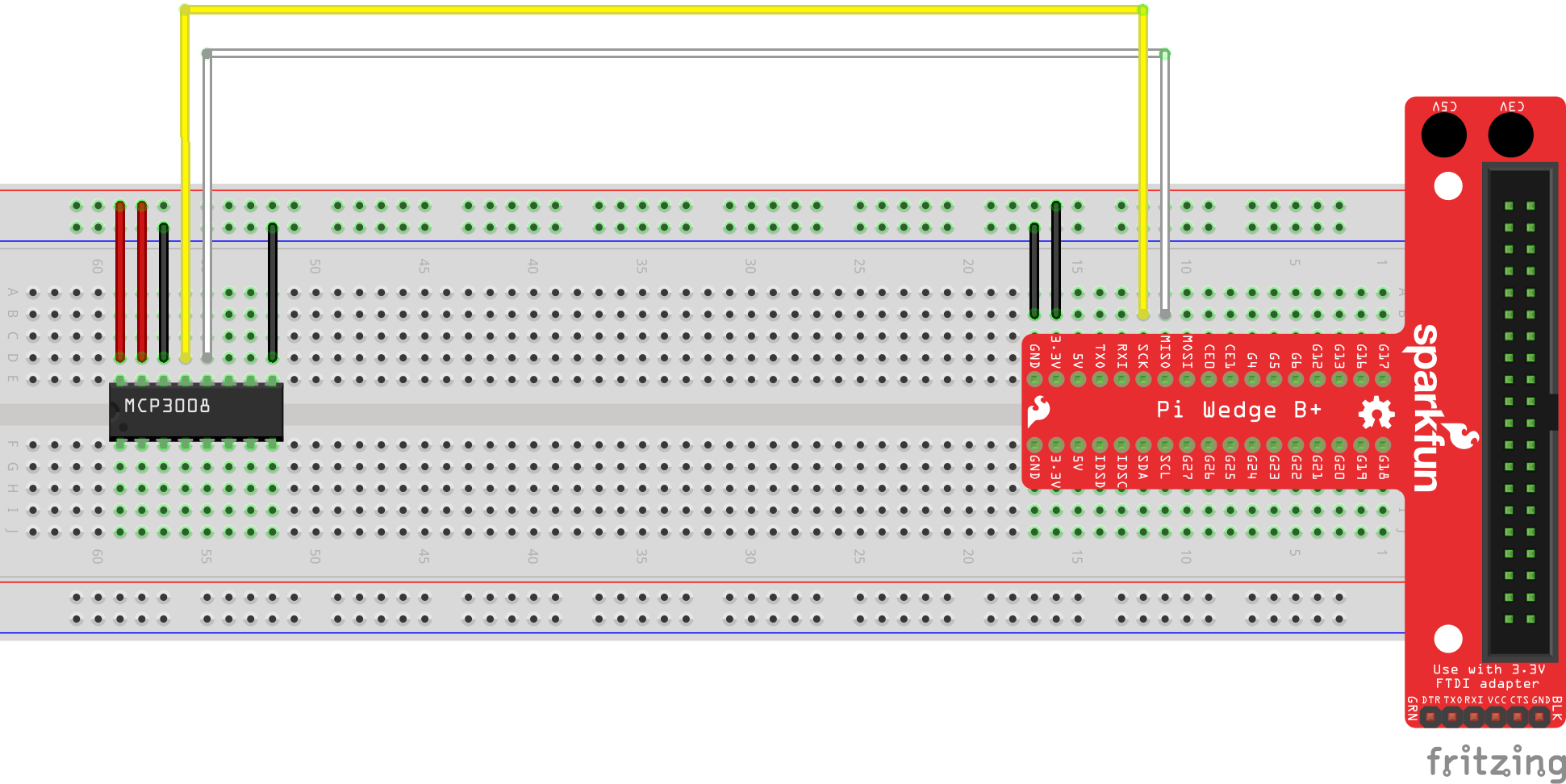


Figure 7 – MCP3008 pin 12 (DOUT) connected to SparkFun Pi Wedge pin MISO.

1. Using a Blue Jumper Wire, connect pin 11 (DIN) to the MOSI pin of the SparkFun Pi Wedge.

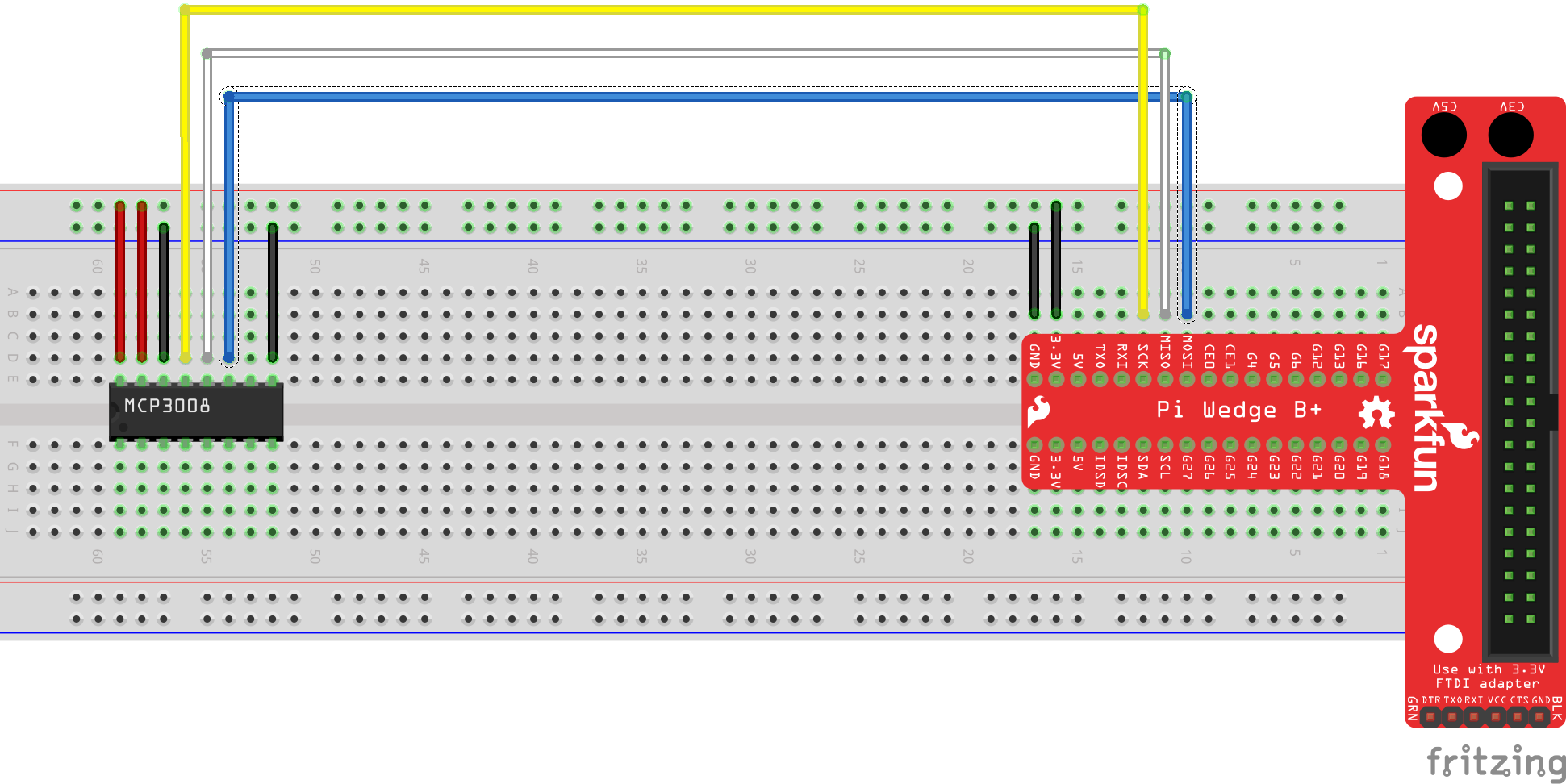


Figure 8 - MCP3008 pin 11 (DIN) connected to SparkFun Pi Wedge pin MOSI.

1. Using a Green Jumper Wire, connect pin 10 (CS) to the CE0 pin of the SparkFun Pi Wedge.

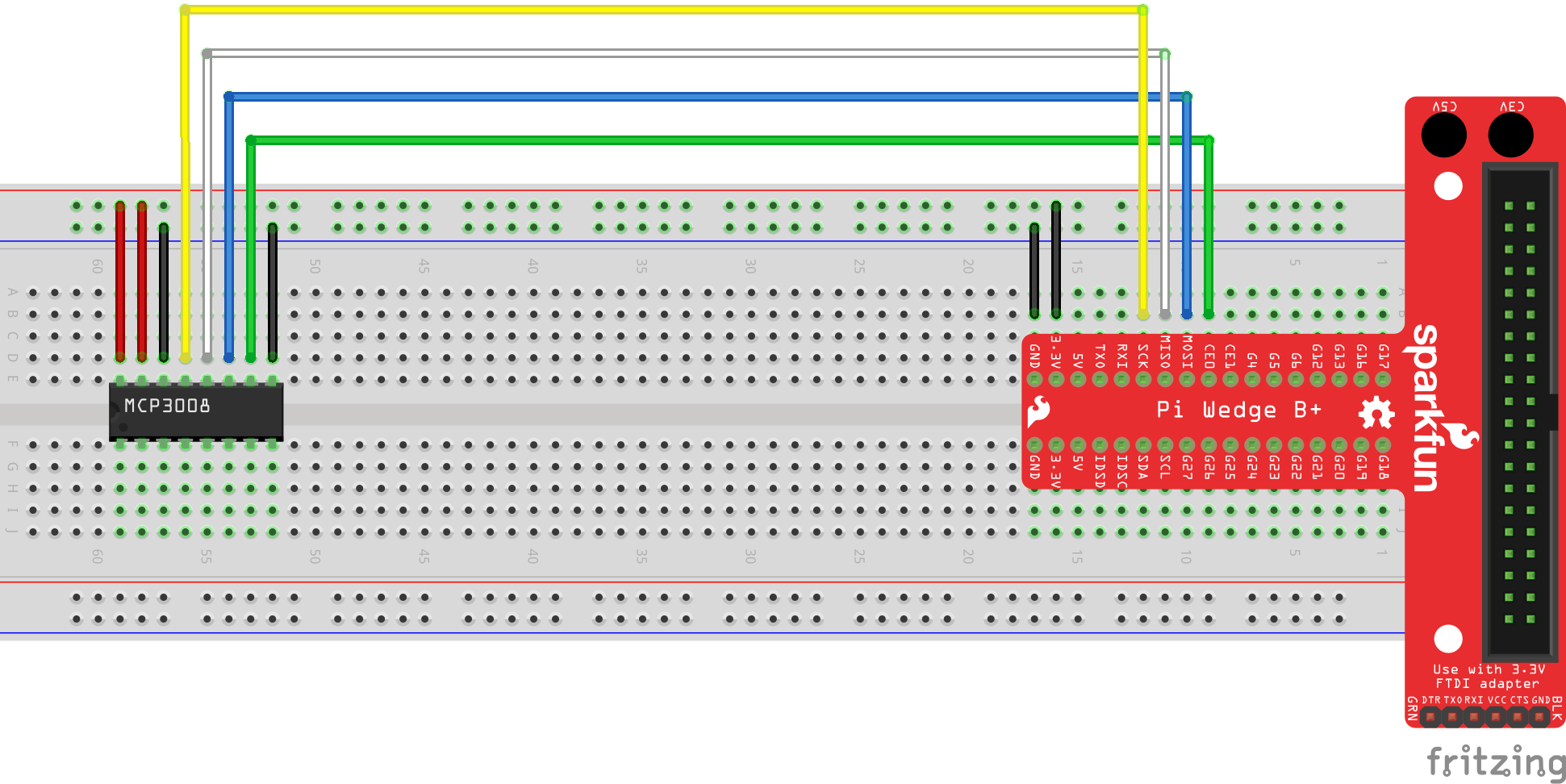


Figure 9 - MCP3008 pin 10 (CS) connected to SparkFun Pi Wedge pin CE0.

1. Connect the TMP36 Temperature Sensor to the breadboard as shown. Be sure to orient the temperature sensor appropriately.

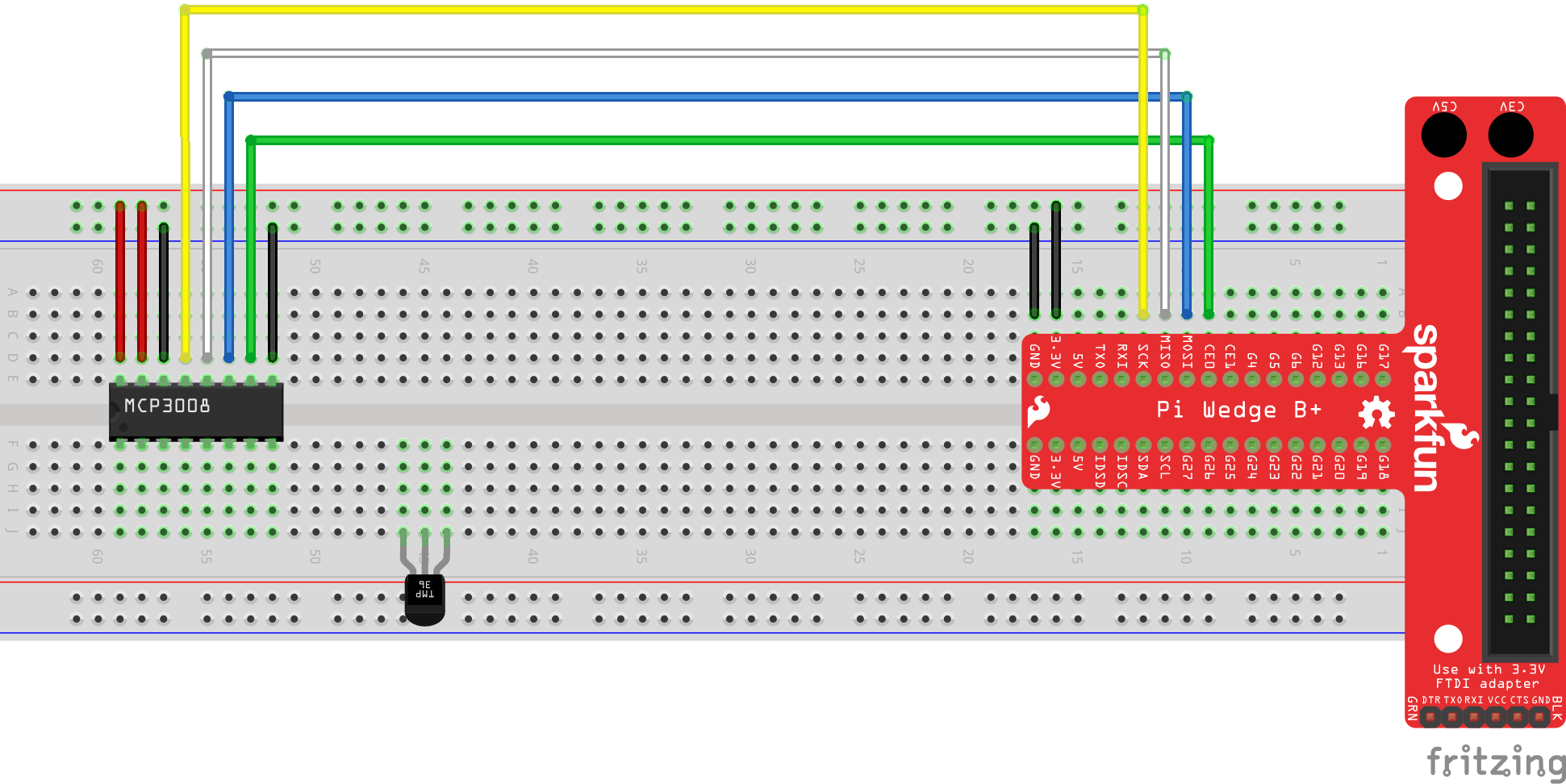


Figure 10 - TMP36 Temperature Sensor connected to the breadboard.

1. Connect the 0.1uF ceramic capacitor across the left two pins of the temperature sensor (GND and VOUT, respectively).



Figure 11 - 0.1 uF Capacitor connected to the breadboard.

1. Using a Black Jumper Wire, connect the left-most pin (GND) of the TMP36 Temperature Sensor to the negative (-) rail of the breadboard.



Figure 12 - TMP36 GND pin connected to GND.

1. Using a Red Jumper Wire, connect the right-most pin (+VS) of the TMP36 Temperature Sensor to the positive (+) rail of the breadboard.

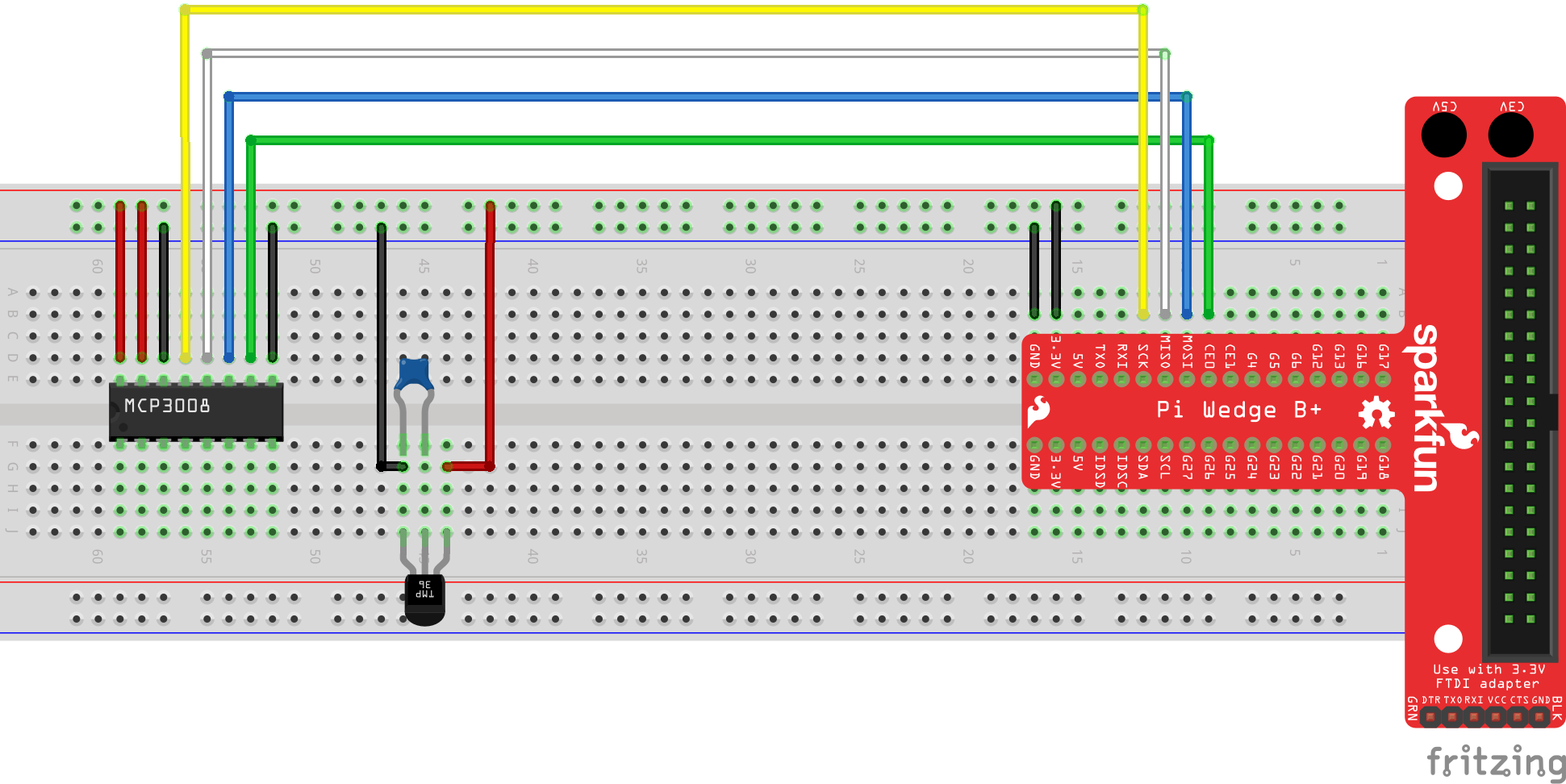


Figure 13 - TMP36 +VS pin connected to 3.3V.

1. Using a Yellow Jumper Wire, connect the middle pin (VOUT) of the TMP36 Temperature Sensor to pin 1 (CH0) of the MCP3008 Analog to Digital Converter.

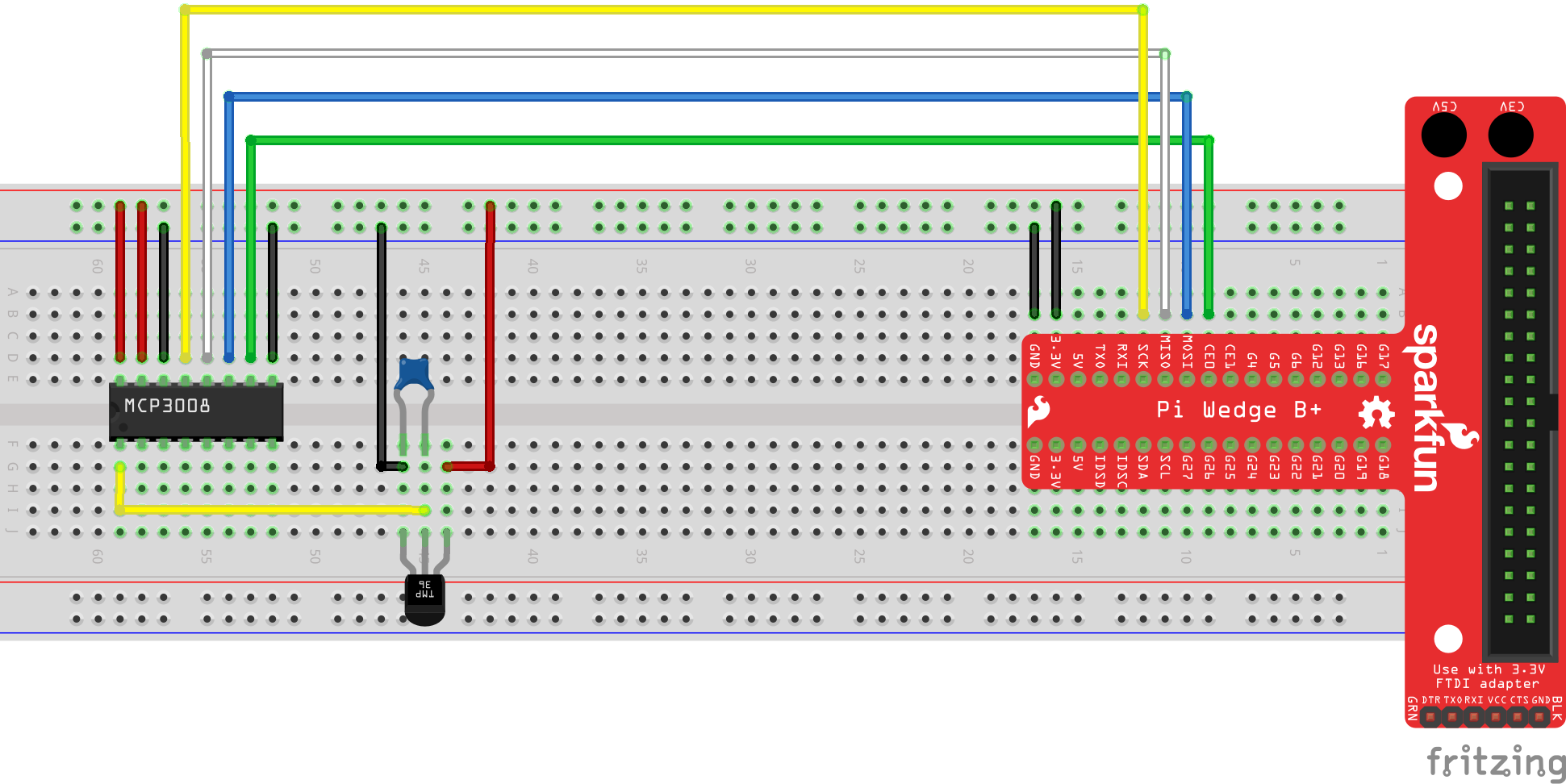


Figure 14 - TMP36 VOUT connected to MCP3008 pin 1 (CH0).

## Part 2 – Code

1. Connect the Wall Adapter Power Supply to the micro USB port on the Raspberry Pi. Next, plug in the Wall Adapter Power Supply.
2. Open the Windows 10 IoT Core Dashboard application. Once your device has completed its power up sequence, you should see your device in the list of *My devices*.

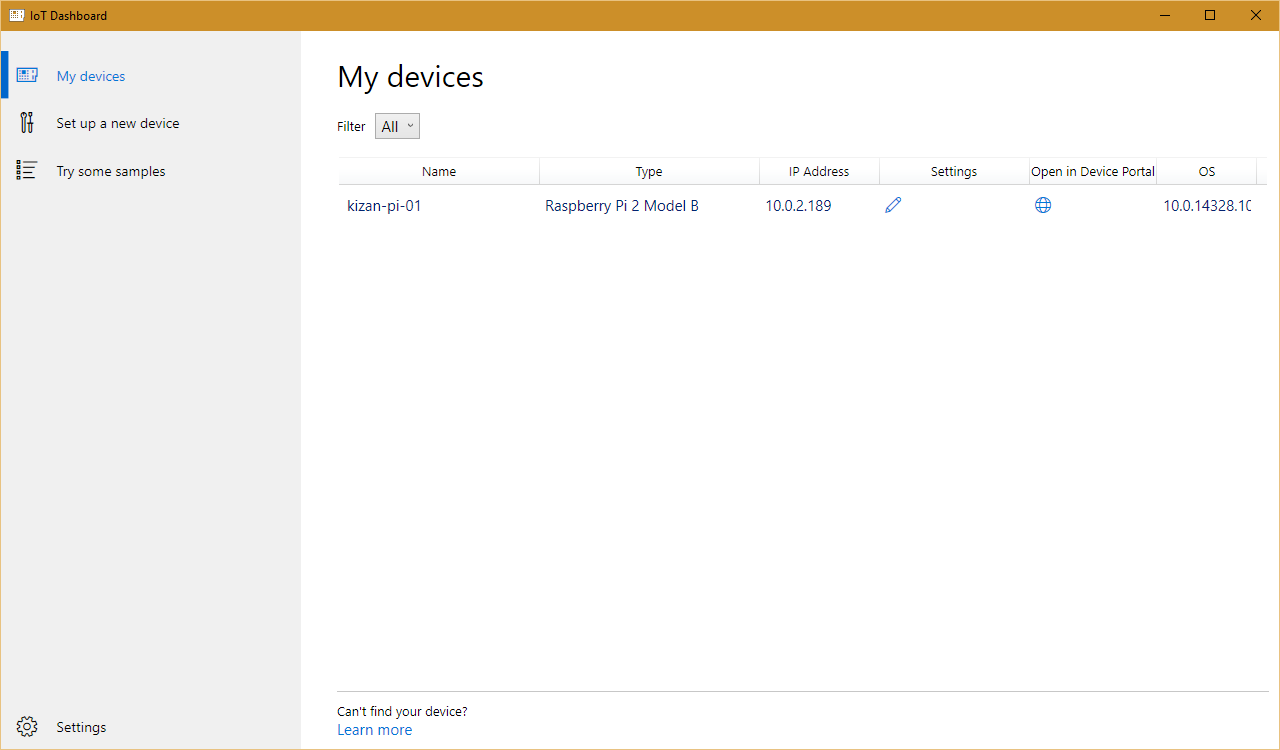


Figure 15 - Windows 10 IoT Core Dashboard

1. Using Visual Studio 2015, open the Lab03.sln solution from the downloaded source package. Lab03.sln is located in the *KiZAN-IoT-Workshop\src\Lab03* folder.
2. Inside of MainPage.xaml.cs, update the value of the *IotHubHostName* constant, replacing the *{iot hub hostname}* token with the proper host name given during the workshop.
3. Update the value of the *DeviceId* constant, replacing the *{device id}* token with the proper device id from your hardware kit. Note: the device id will resemble the name *kizan-pi-01*.
4. Update the value of the *DeviceKey* constant, replacing the *{device key}* token with the proper device key given during the workshop.
5. Verify that the Debug configuration and ARM platform are selected.



Figure 16 - Build Configuration and Platform

1. Next, change Run from Device to Remote Machine.

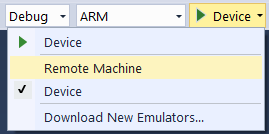


Figure 17 - Change Run from Device to Remote Machine

1. In the Remote Connections dialog box, find your device in the list of Auto Detected devices. Select your device, and click the Select button.

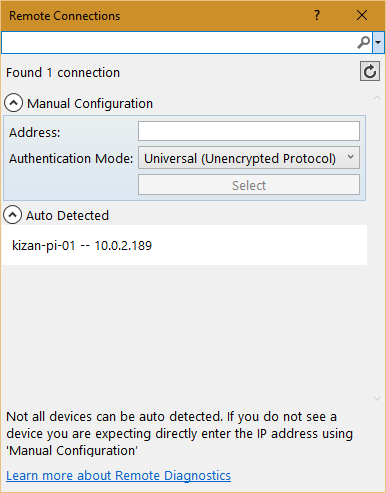


Figure 18 - Remote Connections

1. Press F5 to run the application. Visual Studio will restore any missing NuGet packages, build the application, and then deploy the application to the remote machine.
2. With the application running, open the Output window in Visual Studio (View \ Output).
3. Verify that the current temperature is written to the Output window.

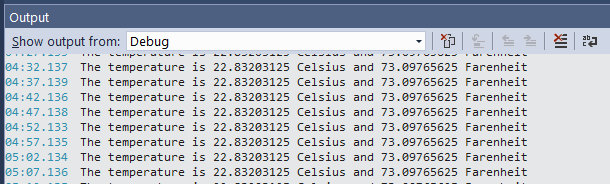


Figure 19 - Output Window

1. We will use the data collected by the IoT Hub in future demonstrations.
2. Open Windows IoT Remote Client.

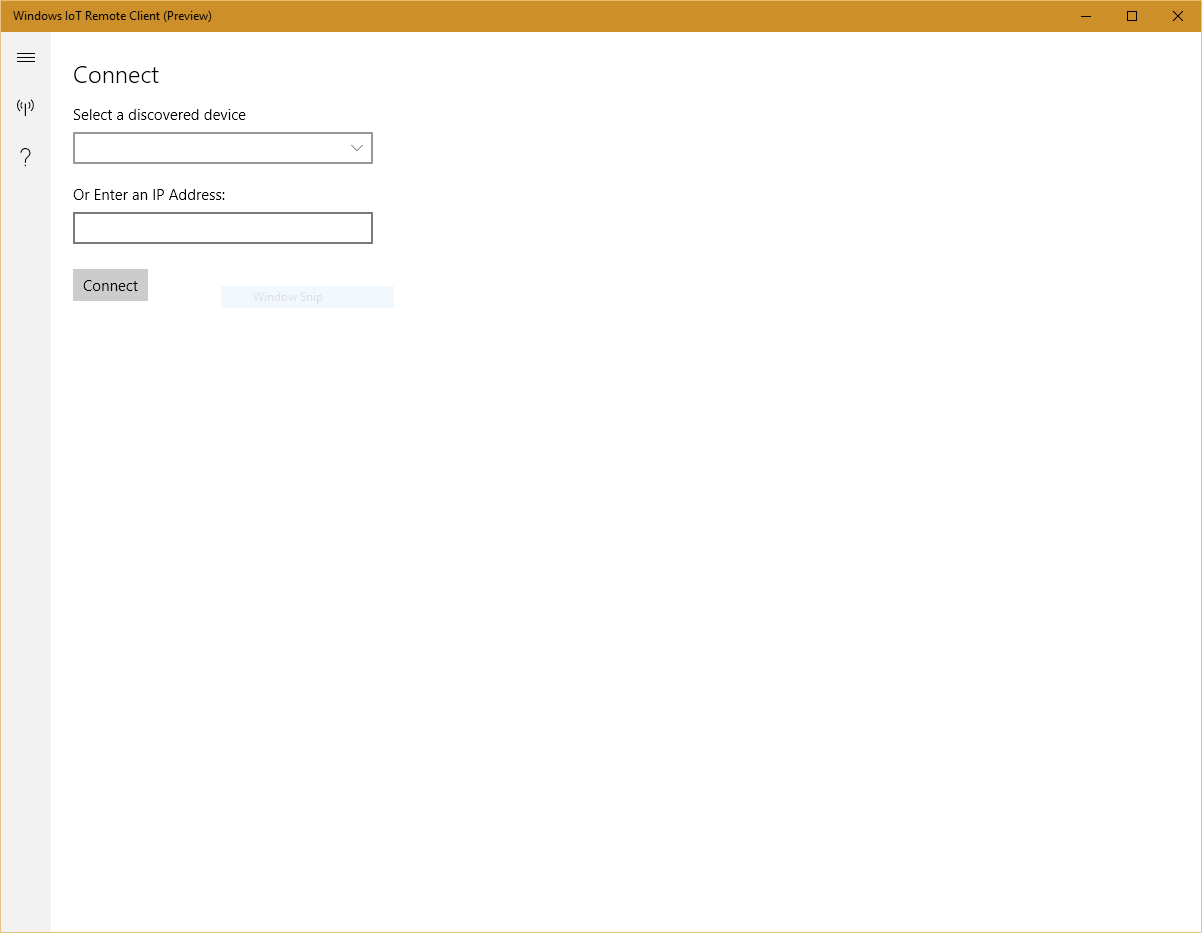


Figure 20 - Windows IoT Remote Client

1. Select your device from the list of discovered devices, or enter the IP Address of your device, and click Connect. You should see the application running on the Raspberry Pi.

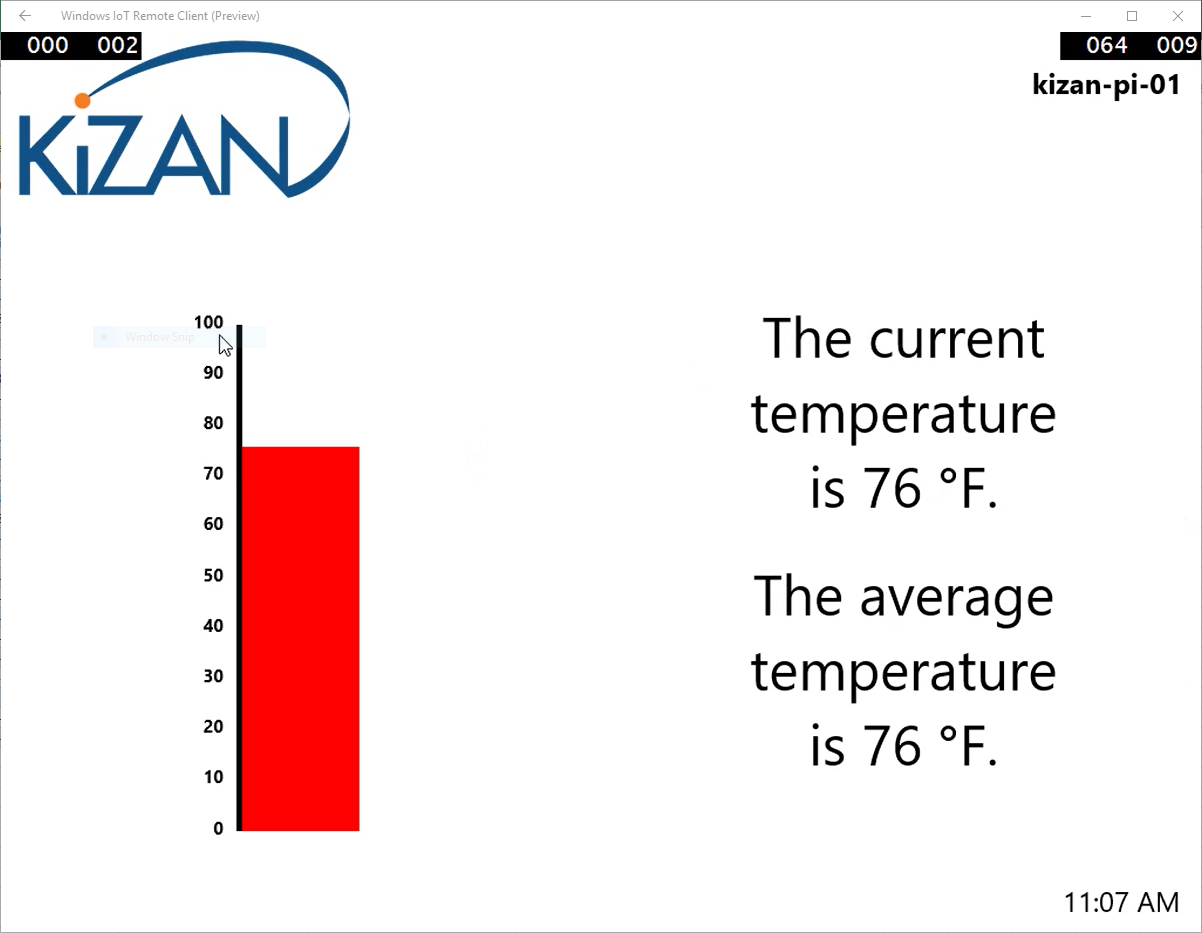


Figure 21 - Lab 03 running on the Raspberry Pi.

1. Stop debugging.
2. Using Windows 10 IoT Core Dashboard, click the globe icon in the *Open in Device Portal* column for your device.
3. When prompted enter the following User Name and Password.

|  |  |
| --- | --- |
| User Name | Password |
| Administrator | p@ssw0rd |

1. In the upper, right-hand corner, click Power, and then click Shut down.

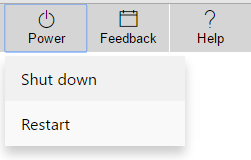


Figure 22 - Power Menu

1. When prompted to confirm, press OK.
2. It is safe to remove power when the Device Portal prompts to “Restart the device to reconnect”.