KiZAN IoT Workshop

Lab Exercises

# Lab 5 – Connecting to IoT Hub

## Goal

The goal of this exercise is to connect to an Azure IoT Hub and stream telemetry data using a Windows IoT Core device (Raspberry Pi 2, in our case) and an app built and debugged with Visual Studio 2015.

## Requirements

Prior to starting this exercise, please ensure that you meet the following requirements. This exercise assumes that you have previously completed Lab 4.

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

No additional hardware is necessary for this lab.

### 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 – 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*.

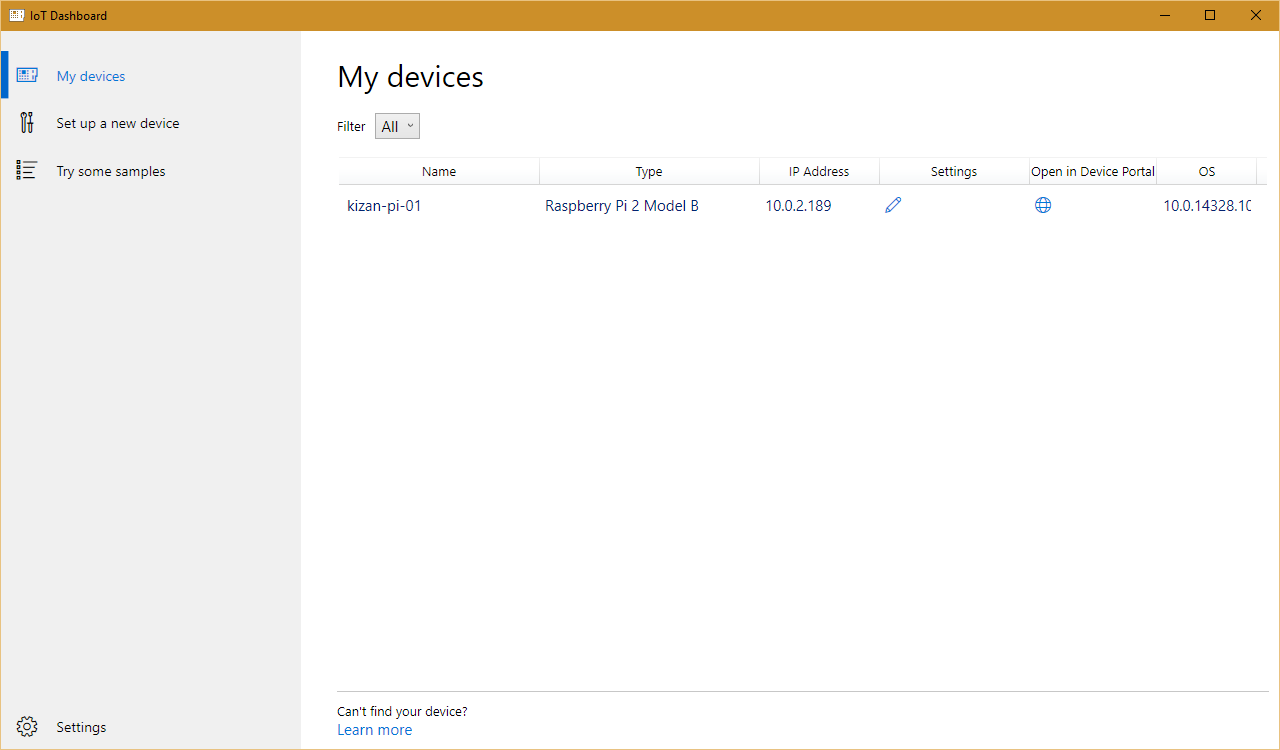


Figure 15 - Windows 10 IoT Core Dashboard

1. Using Visual Studio 2015, open the Lab05.sln solution from the downloaded source package. Lab05.sln is located in the *KiZAN-IoT-Workshop\src\Lab05* folder.
2. Inside of StartupTask.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.

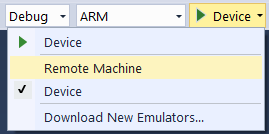


Figure 17 - Change Run from Device to Remote Machine

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.

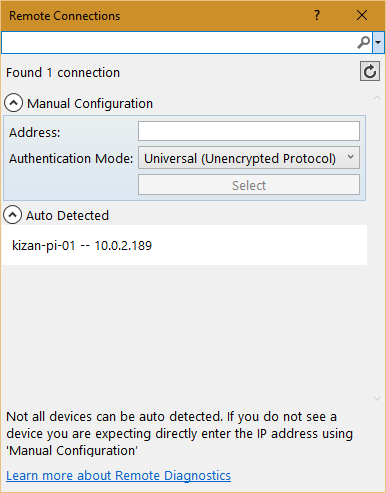


Figure 18 - Remote Connections

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.

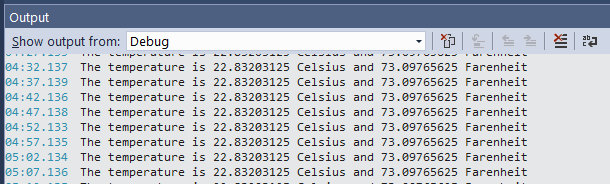


Figure 19 - Output Window

1. We will use the data collected by the IoT Hub in future demonstrations.
2. Stop debugging.
3. Using Windows 10 IoT Core Dashboard, click the globe icon in the *Open in Device Portal* column for your device.
4. 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.

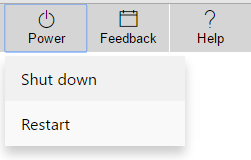


Figure 20 - Power Menu

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”.