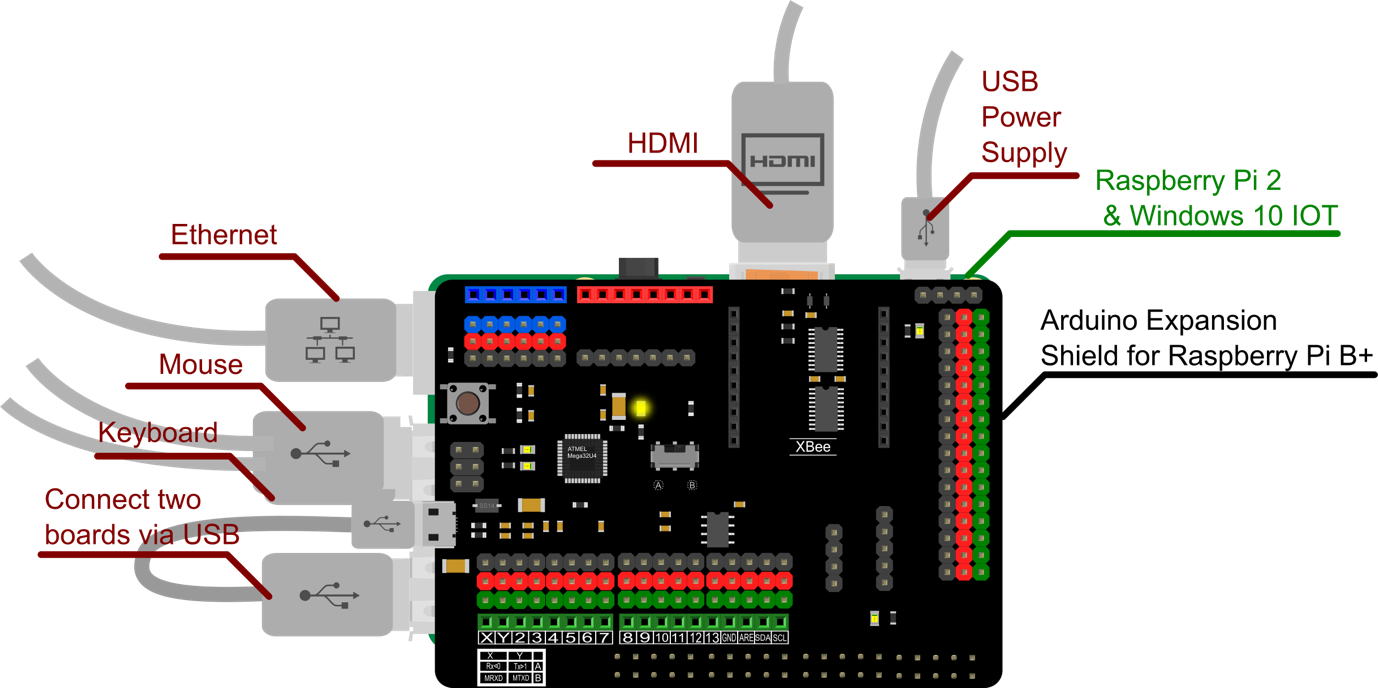
Window 10 IOT Get Started

In this tutorial, you can get stared with the window 10 IOT through the Basic Demo, blinking the onboard LED.

Hardware List:

* Raspberry Pi 2
* Arduino Expansion Shield for Raspberry Pi B+
* Magnet Micro USB Cable
* CAT 5 Ethernet Cable
* HDMI Cable
* Micro SD Memory Card 16GB Class10
* USB Keyboard
* USB Mouse

Hardware Connection:

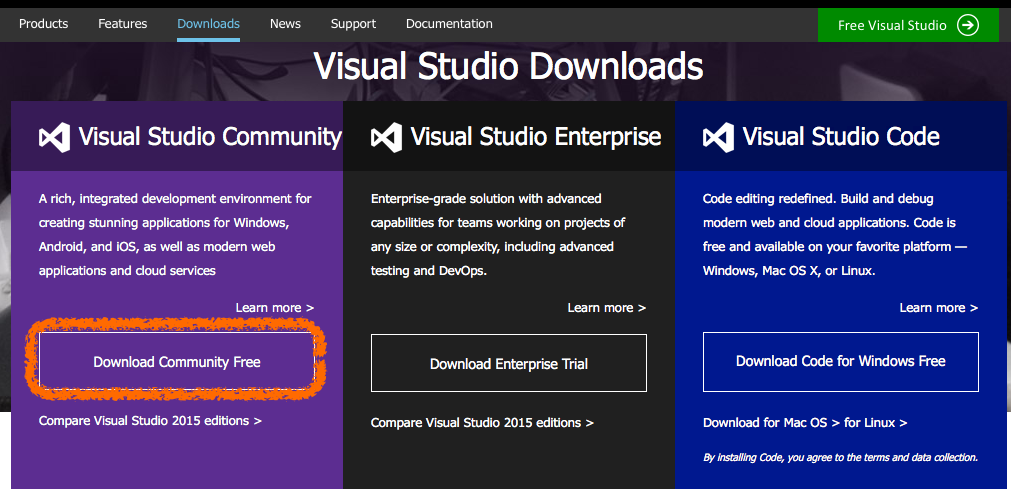


1. Plug the “Arduino Expansion Shield for Raspberry Pi B+” onto the Raspberry Pi 2
2. Connect Ethernet, mouse, keyboard, HDMI.
3. Connect Two boards via USB.
4. Insert the Micro SD card into Raspberry Pi.
5. Connect USB Power Supply.
6. After windows 10 IOT is successfully booted. Some basic device information is there and you can use your mouse to control it.

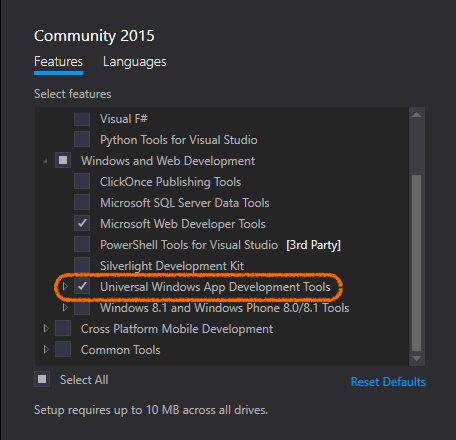
Set up your PC

1. Make sure you are running Windows 10 (version 10.0.10240) or better.
2. Download Visual Studio 2015 Here:

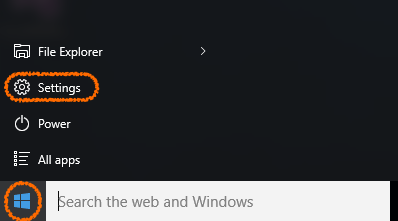
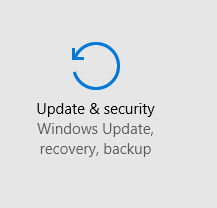
<https://www.visualstudio.com/downloads/download-visual-studio-vs>

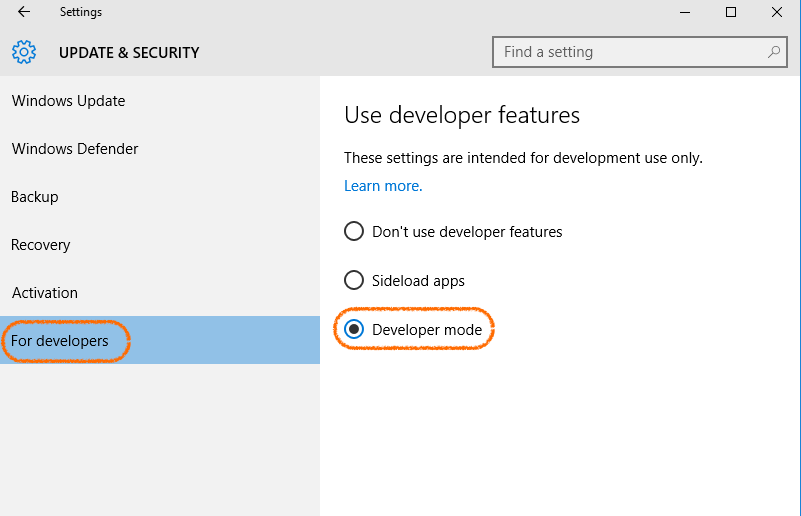


1. Install It, remember to select the “Universal Windows App Development Tools”:



1. Enable developer mode on your windows 10



Blink Demo:

1. Download the “DFrobotWindowIOTTempelate”

<https://github.com/DFRobot/DFrobotWindowIOTTempelate>

1. Open DFrobotWindowIOTTempelate Project in the DFrobotWindowIOTTempelate folder.



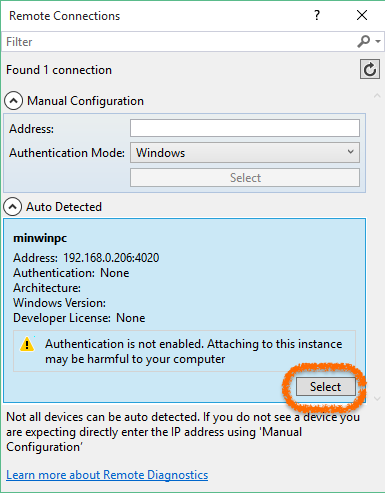
1. Change the Target to ARM:



1. Change the target device to Remote Machine:



1. Select your raspberry Pi on the network



1. Main Code is in “MainPage.xaml.cs”.



using Windows.UI.Xaml;

using Windows.UI.Xaml.Controls;

using Microsoft.Maker.RemoteWiring;

using Microsoft.Maker.Serial;

using System;

using System.Diagnostics;

namespace DFrobotWindowIOTTempelate

{

public sealed partial class MainPage : Page

{

UsbSerial usb; //Handle the USB connction

RemoteDevice arduino; //Handle the arduino

private DispatcherTimer blinkTimer; //Timer for the LED to blink every one second

private const int LED\_PIN = 13; //Pin number of the on board LED

private PinState ledState; //Pin state of the LED

public MainPage()

{

this.InitializeComponent();

//USB VID and PID of the "Arduino Expansion Shield for Raspberry Pi B+"

usb = new UsbSerial("VID\_2341", "PID\_8036");

//Arduino RemoteDevice Constractor via USB.

arduino = new RemoteDevice(usb);

//Add DeviceReady callback when connecting successfully

arduino.DeviceReady += onDeviceReady;

//Baudrate on 57600 and SerialConfig.8N1 is the default config for Arduino devices over USB

usb.begin(57600, SerialConfig.SERIAL\_8N1);

}

private void onDeviceReady()

{

//After device is ready this function will be evoked.

//Debug message "Device Ready" will be shown in the "Output" dialog.

Debug.WriteLine("Device Ready");

var action = Dispatcher.RunAsync(Windows.UI.Core.CoreDispatcherPriority.Normal, new Windows.UI.Core.DispatchedHandler(() =>

{

setup();

}));

}

private void setup()

{

//Set the initial state of the led.

ledState = PinState.LOW;

//Set the pin mode of the led.

arduino.pinMode(LED\_PIN, PinMode.OUTPUT);

//Set the timer to schedule blink() every one second.

blinkTimer = new DispatcherTimer();

blinkTimer.Interval = TimeSpan.FromMilliseconds(1000);

blinkTimer.Tick += blink;

blinkTimer.Start();

}

private void blink(object sender, object e)

{

if (ledState == PinState.HIGH) //LED state is HIGH.

{

//Turn off the LED.

arduino.digitalWrite(LED\_PIN, PinState.LOW);

//Show the message in the Output dialog.

Debug.WriteLine("OFF");

//Set local LED state to Low.

ledState = PinState.LOW;

}

else //LED state is LOW.

{

//Turn on the LED.

arduino.digitalWrite(LED\_PIN, PinState.HIGH);

//Show the message in the Output dialog.

Debug.WriteLine("ON");

//Set local LED state to Low.

ledState = PinState.HIGH;

}

}

}

}

1. The onboard LED start to blink. Congratulations!!

