

Source: C# Corner (www.c-sharpcorner.com)

PRINT

Article



Experiment On IoT: Simple LED Blink Example

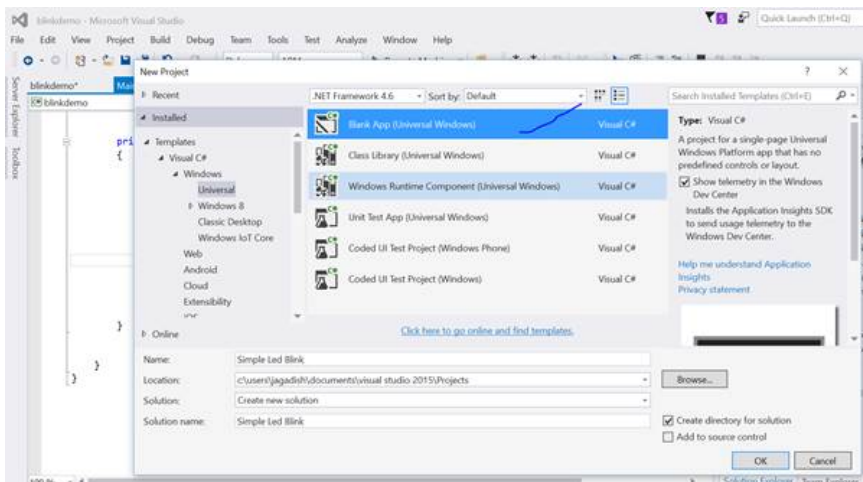
By **Abdul Rasheed Feroz Khan** on **Apr 21 2016**

Requirements:

1. Raspberry Pi 2/3 with Windows 10 IoT Core Operating System connected to internet.
2. Laptop or PC with Windows 10 OS connected to Internet.
3. Visual Studio 2015 installed on your laptop or PC.
4. Bread Board
5. LED light
6. Resistor

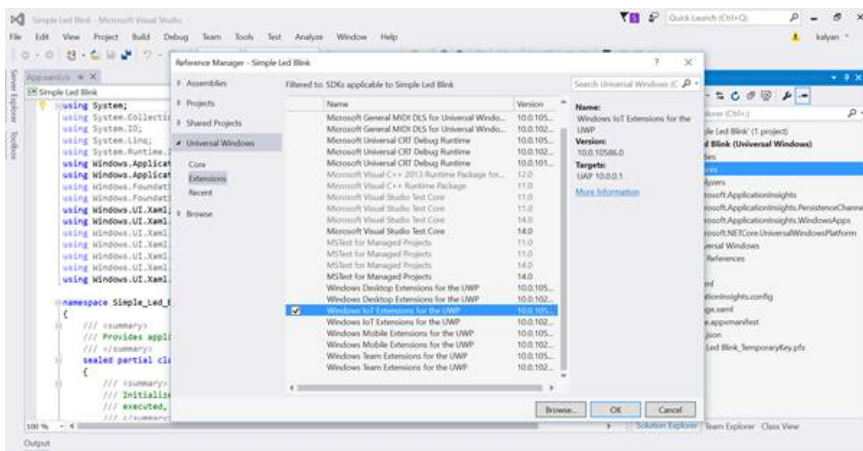
Follow the below steps:

Step 1: Click *File*, then *New Project* like the following screenshot,



Name it *Simple Led Blink*.

Step 2: Right click on the selection and add reference,



Select Windows IoT Extension for the UWP and click OK.

Step 3: Open MainPage.xaml and add the following for UI,

1. <Grid Background="Wheat">
2. <StackPanel HorizontalAlignment="Center" VerticalAlignment="Center">
3. <Ellipse x:Name="LED" Fill="LightGray" Stroke="White" Width="100" Height="100" Margin="10"/>
4. <TextBlock x:Name="DelayText" Text="500ms" Margin="10" TextAlignment="Center" FontSize="26.667" />
5. <TextBlock x:Name="GpioStatus" Text="Waiting to initialize GPIO..." Margin="10,50,10,10" TextAlignment="Center" FontSize="26.667" />

```

6.     </StackPanel>
7. </Grid>

```

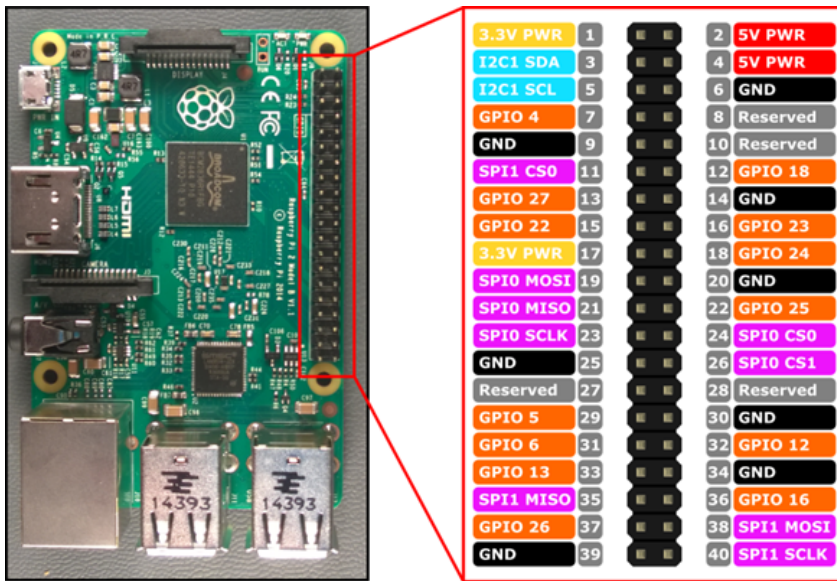
Step 4: Open the MainPage.xaml.cs add the following in the cs file.

```

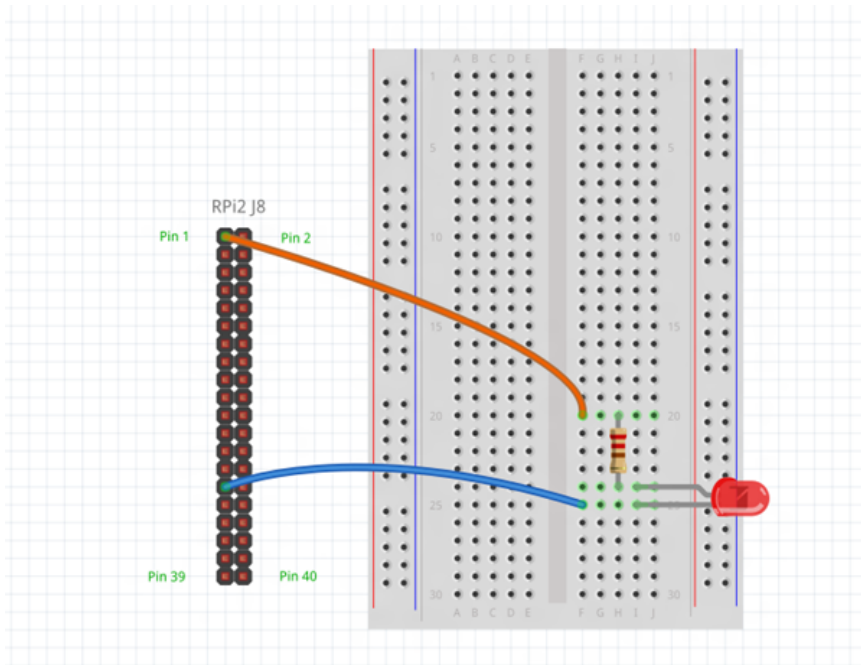
1. private const int LED_PIN =5;
2.     private GpioPin pin;
3.     private GpioPinValue pinValue;
4.     private DispatcherTimer timer;
5.     private SolidColorBrush redBrush = new SolidColorBrush(Windows.UI.Colors.Red);
6.     private SolidColorBrush grayBrush = new SolidColorBrush(Windows.UI.Colors.LightGray);
7.
8.     public MainPage()
9.     {
10.         InitializeComponent();
11.
12.         timer = new DispatcherTimer();
13.         timer.Interval = TimeSpan.FromMilliseconds(500);
14.         timer.Tick += Timer_Tick;
15.         InitGPIO();
16.         if (pin != null)
17.         {
18.             timer.Start();
19.         }
20.     }
21.
22.     private void InitGPIO()
23.     {
24.         var gpio = GpioController.GetDefault();
25.
26.         // Show an error if there is no GPIO controller
27.         if (gpio == null)
28.         {
29.             pin = null;
30.             GpioStatus.Text = "There is no GPIO controller on this device.";
31.             return;
32.         }
33.
34.         pin = gpio.OpenPin(LED_PIN);
35.         pinValue = GpioPinValue.High;
36.         pin.Write(pinValue);
37.         pin.SetDriveMode(GpioPinDriveMode.Output);
38.
39.         GpioStatus.Text = "GPIO pin initialized correctly.";
40.
41.     }
42.     private void Timer_Tick(object sender, object e)
43.     {
44.         if (pinValue == GpioPinValue.High)
45.         {
46.             pinValue = GpioPinValue.Low;
47.             pin.Write(pinValue);
48.             LED.Fill = redBrush;
49.         }
50.         else
51.         {
52.             pinValue = GpioPinValue.High;
53.             pin.Write(pinValue);
54.             LED.Fill = grayBrush;
55.         }
56.     }
57.
58. }
59.

```

In the above code we are using the GPIO(General purpose input output ports) for the interaction of the electronic circuit. We are using GPIO 5 over here,

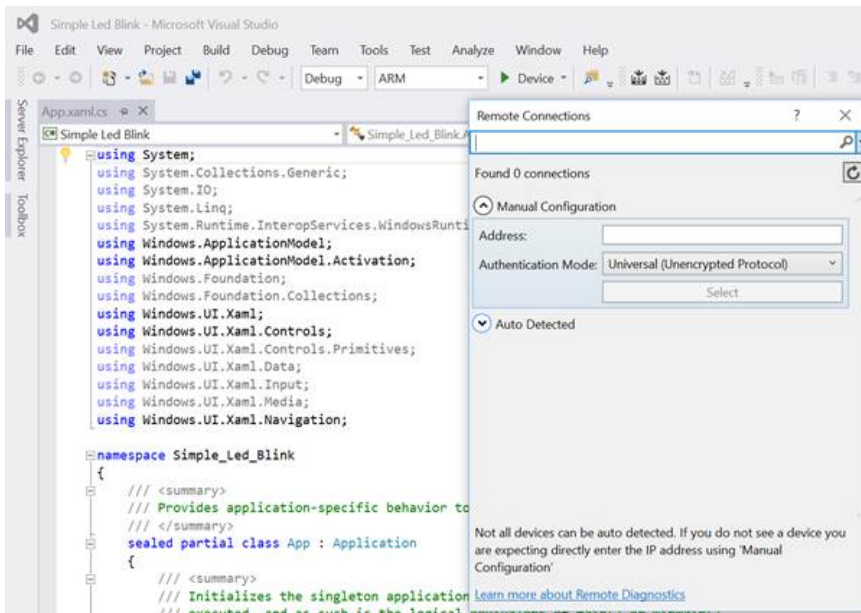


Connect the following circuit,

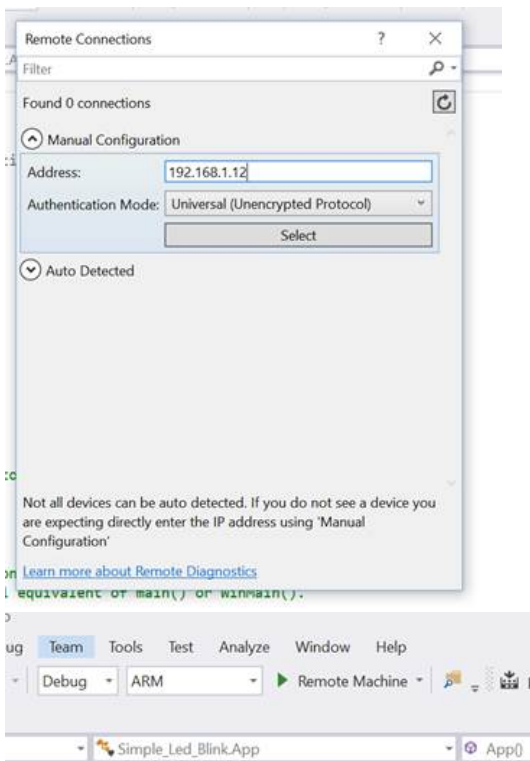


Once the circuit is connected, build the solution.

Step 5: Now build the solution and select ARM and select Remote Machine,



In the address column enter the ip address of the network connected with Raspberry Pi device and click ok.



Step 6:

Now once Remote Machine is selected build and run the application.

Once the application is built & run operation is completed successfully in the sense you will find the LED blinking according to the timer set.

Read more articles on **Internet of Things**:

- [Understanding IoT Analytics And Its Future Growth Prospects](#)
- [Finding The Location Status In Arduino Mega 2560](#)

Thank you for using C# Corner