

Windows 10 IoT



Windows 10 IoT 是 Windows 10 的專用版本，專為嵌入式設備和工業環境設計。它提供了更長的支援週期、更高的安全性和可靠性，以及針對特定行業的優化功能。Windows 10 IoT 允許開發人員在各種硬體平台上部署 Windows 10，從而簡化開發過程並提高生產力。


Windows 10 IoT 的安裝過程相對簡單，但需要一些技術知識。開發人員需要選擇合適的硬體平台，並確保其符合 Windows 10 IoT 的要求。安裝過程包括下載 Windows 10 IoT 安裝映像、將其燒錄到 USB 驅動器或 SD 卡上，然後將映像安裝到目標設備上。安裝後，開發人員可以根據需要進行配置和定制，以滿足特定應用程序的需求。

Windows 10 IoT 安裝指南

Windows 10 IoT 安裝指南

- .10 10
- .2
- .8
- .HDMI
- .
-
- Microsoft IoT

.10 10 500 10 <http://go.microsoft.com/fwlink/?LinkId=616847> IOT Core Rpi Windows_10_IoT_Core-Rpi2

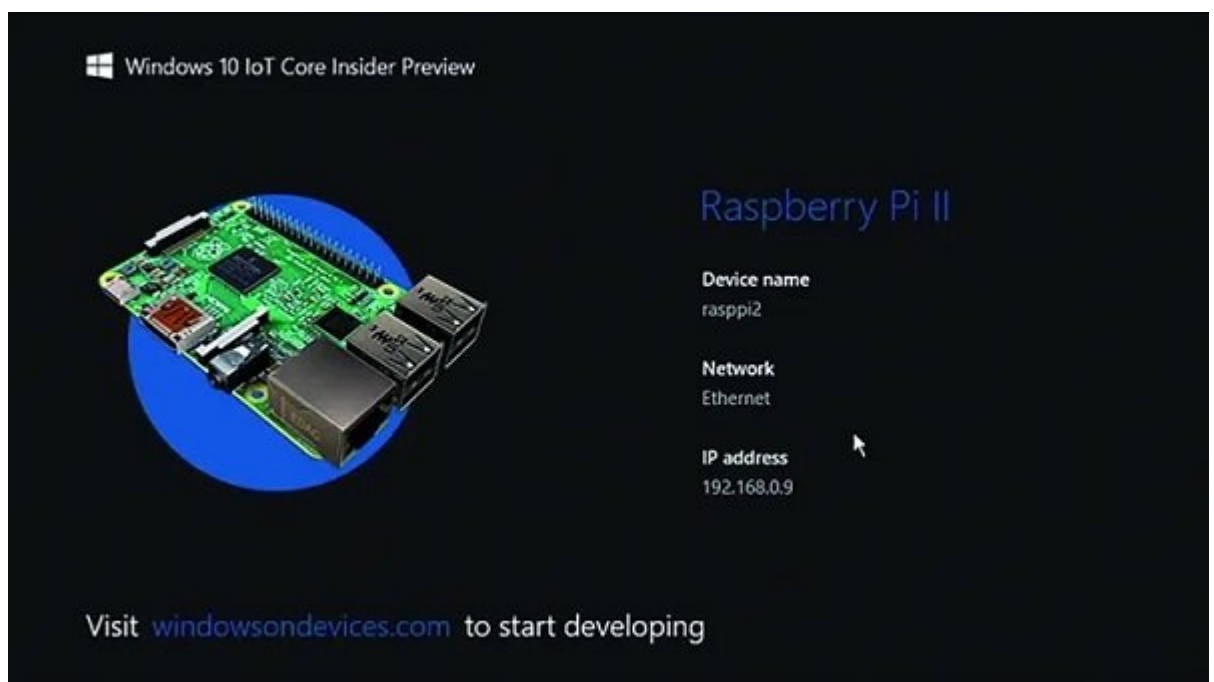
This PC > DVD Drive (H:) J_ARPI_A32FRE0_EN-US_DV5			
Name	Date modified	Type	Size
 Windows_10_IoT_Core_RPi2	6/17/2015 2:04 PM	Windows Installer ...	472,600 KB

C:\Program Files (x86)\Microsoft FLASH.FFU IoT\FFU\RaspberryPi2 WindowsIoTImageHelper FFU FLAH



Windows 10 IoT Core 安裝與測試

這篇 文章 介紹了 Windows 10 IoT Core 的安裝與測試。首先，我們需要準備一個 SD 卡，並將其插入電腦。然後，我們需要下載 Windows 10 IoT Core 的映像檔，並使用 Windows IoT Core Image Helper 工具將其燒錄到 SD 卡上。最後，我們將 SD 卡插入 Raspberry Pi 2，並啟動系統。測試部分包括檢查網路設定、IP 地址以及系統的基本功能。



3 圖

10 圖展示了 Windows 10 IoT Core 的安裝與測試過程。圖中顯示了 Windows IoT Core Image Helper 的安裝界面，以及 Raspberry Pi 2 的系統信息，包括 IP 地址 192.168.0.9。

10. **අනුරාධපුර පුරාණ නගරයේ පිහිටි පුරාණ ස්ථානවලට පිවිසීමේදී ප්‍රතිරෝධයක් දැක්වූයේ නැත.**

☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐

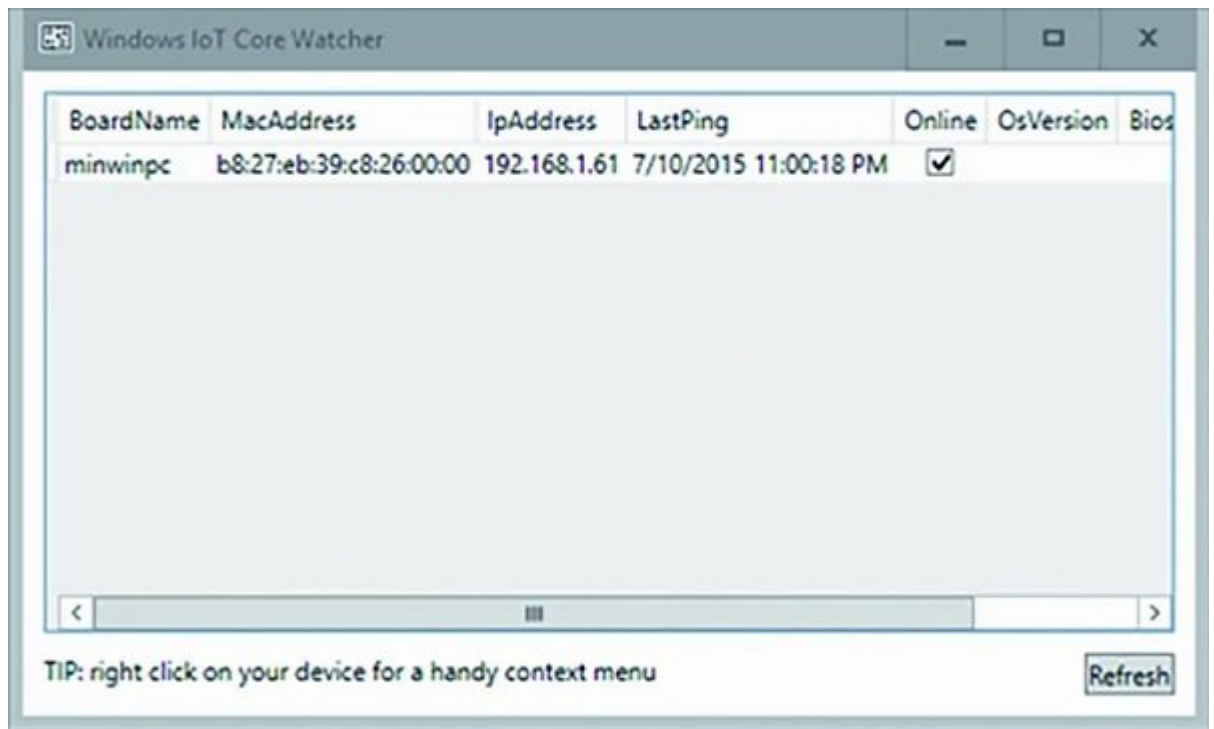
۱. در ابتدا باید مطمئن شوید که سیستم شما از Windows 10 IoT Core پشتیبانی می‌کند. برای این منظور، به وبسایت <https://www.microsoft.com/en-us/windows/iot-core> مراجعه کنید.

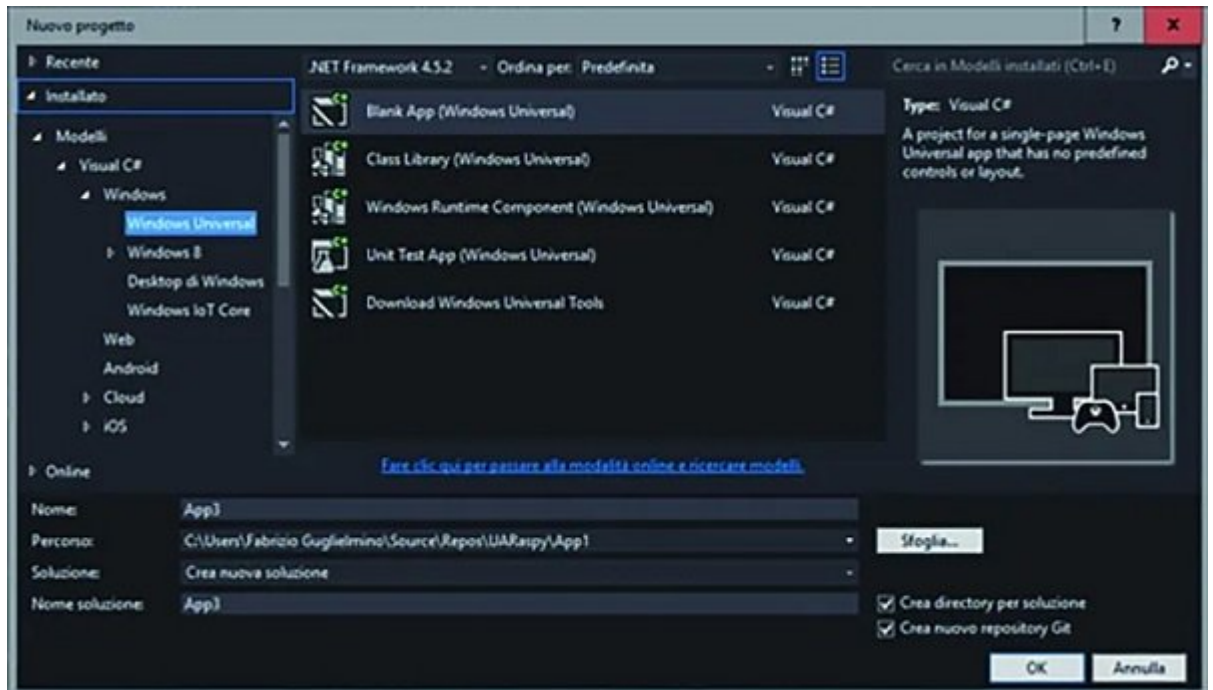
۲. پس از اطمینان از پشتیبانی، باید یک دستگاه IoT Core را تهیه کنید. این دستگاه‌ها معمولاً به صورت بردهای توسعه یا دستگاه‌های کوچک عرضه می‌شوند.

۳. برای نصب و راه‌اندازی Windows 10 IoT Core، باید یک سیستم عامل IoT Core را روی دستگاه خود نصب کنید. این فرآیند شامل دانلود فایل‌های نصب و اجرای آن‌هاست.

۴. پس از نصب، باید دستگاه خود را به اینترنت متصل کنید و یک حساب کاربری Microsoft ایجاد کنید.

۵. در نهایت، می‌توانید از ویژگی‌های مختلف Windows 10 IoT Core استفاده کنید، از جمله برنامه‌های سفارشی، ابزارهای توسعه و ...

[illegible]



Il primo passo è creare un nuovo progetto. In Visual Studio, selezionare "Nuovo progetto" e scegliere "Windows Universal" come modello. Il progetto sarà creato in una cartella specifica. Il file principale, MainPage.xaml, conterrà il contenuto dell'applicazione. In questo caso, il contenuto sarà "Hello World".

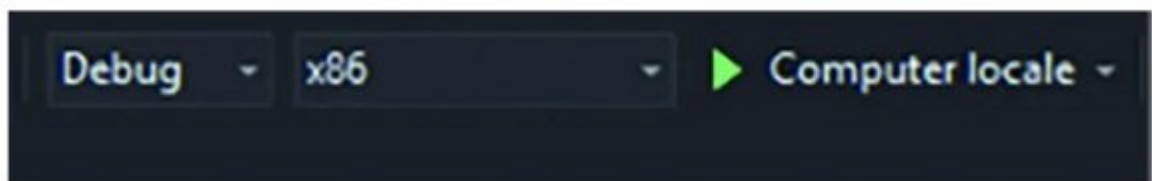
Listing 1

```

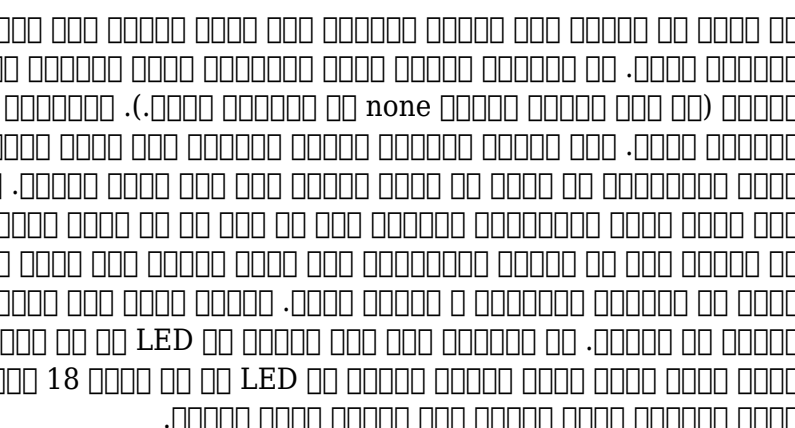

1 <Page
2 x:Class="App2.MainPage"
3 xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
4 xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
5 xmlns:local="using:App2"
6 xmlns:d="http://schemas.microsoft.com/expression/blend/2008"
7 xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"
8 mc:Ignorable="d">
9 <Grid Background="{ThemeResource ApplicationPageBackgroundThemeBrush}">
10 <TextBlock Foreground="Blue" HorizontalAlignment="Center"
11 VerticalAlignment="Center" FontSize="64">Hello World</TextBlock>
12 </Grid>
13 </Page>

```

Il codice XAML definisce una pagina con uno sfondo blu e un testo "Hello World" in bianco, centrato. Il font size è 64. Il testo è contenuto in un TextBlock all'interno di un Grid.



Il risultato della compilazione è l'applicazione "Hello World" che viene eseguita sul computer locale.



GPIOs
 MainPage.xaml
 11

```

1  <Page
2  x:Class="App2.MainPage"
3  xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
4  xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
5  xmlns:local="using:App2"
6  xmlns:d="http://schemas.microsoft.com/expression/blend/2008"
7  xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"
8  mc:Ignorable="d">
9  <Grid Background="{ThemeResource ApplicationPageBackgroundThemeBrush}" >
10 <StackPanel HorizontalAlignment="Center" VerticalAlignment="Center">
11 <TextBlock Foreground="Blue" HorizontalAlignment="Center"
12 FontSize="48">LED Control</TextBlock>
13 <Button Click="Button_Click" HorizontalAlignment="Center" Margin="
14 0,50,0,0" FontSize="64" >OFF</Button>
15 </StackPanel>
16 </Grid>
17 </Page>
    
```

11

StackPanel
 (TextBlock)
 Click="Button_Click
 Button_Click

```

1  bool state = false;
2  const int LED_PIN = 18;
3  GpioPin pin;
4  private void Button_Click(object sender, RoutedEventArgs e)
5  {
6  state = !state;
7  var pinValue = state ? GpioPinValue.High : GpioPinValue.Low;
8  pin.Write(pinValue);
9  Button btn = (Button)sender;
10 btn.Content = state ? "ON" : "OFF";
11 }
    
```

12

LED
 State
 pin
 GPIO
 LED
 13

```

1  private void InitGPIO()
2  {
3  var gpio = GpioController.Default;
4  pin = gpio.OpenPin(LED_PIN);
5  pin.SetDriveMode(GpioPinDriveMode.Output);
6  pin.Write(GpioPinValue.Low);
7  }
    
```

13

GpioController
 InitGPIO
 pin
 LED
 14
 InitGPIO



14 □□□

[illegible]

0000 000000 0000 .00000 00000 00 000 000000 0000 0000 0000000 00 00000 0 000000 0000 000000 000 00
 00 000 00000 00 Windows 10 IoT Core 00000 00000 0000 00 00 000000 000 000000 0000 0000 00000 000 000000
 0000 00 000 00000 .000 000000 00 <http://minwinpc.local:8080> 00000 00000 00 00 000000 000000 000000000
 00000 00 minwinpc.local 0000 00 00 000 000000 00 00000 0000000 000 00000 00000 00000 00000 00000 00 0000000
 00 .000 p@ssw0rd 0 Administrator 000000 00 0000000 000000000 0 0000000 0000 .000 000000000 0000 00000
 0000000 0000000000 0000 000000 00000 00000 00000000000 00 00000000 000000 000000 00000000 000 0000000 0000
 .0000 00000 00000 000000000000 0 00000 000000 00000

