**Utilizzare la porta Seriale in Windows 10**

Con Windows 10 il namespace **System.IO.Ports** che conteneva la classe **SerialPort**, non è più disponibile.

Bisogna usare **Windows.Devices.SerialCommunication** e la relativa classe **SerialDevice**

Inoltre, su Raspberry Pi2 l’unica seriale presente, viene utilizzata dal kernel debugger quindi non è a disposizione dell’utente.

Per creare app che utilizzano la seriale, è necessario montare un adattarore USB->232   
(ad esempio il classico FT232-R della FTDI)

Ricordarsi di abilitare le porte seriali nel manifest della nostra App.

Notare che è possibile anche utilizzare il driver FTDI in modalità D2XX (vedi qui <https://github.com/Jark/FTDISample> ) e vederlo come un device più evoluto, non solo una normale serial port.

Ad esempio è possibile accedere direttamente alle porte CBUS del dispositivo (una sorta di piccola porta parallela bidirezionale da 5 bit)

La documentazione FTDI per Windows 8.1 è questa, ma è valida anche per Win10  
<http://www.ftdichip.com/Support/Documents/InstallGuides/AN_271%20D2xx%20WinRT%20Guide.pdf>

Questo è un esempio di package.appmanifest per utilizzare la seriale

<Capabilities>

<Capability Name="internetClient" />

<DeviceCapability Name="serialcommunication">

<Device Id="any">

<Function Type="name:serialPort"/>

</Device>

</DeviceCapability>

<DeviceCapability Name="usb">

<!--FT232AM, FT232BM, FT232R and FT245R Devices-->

<Device Id="vidpid:0403 6001">

<Function Type="name:vendorSpecific" />

</Device>

<!--FT2232D and FT2232H Devices-->

<Device Id="vidpid:0403 6010">

<Function Type="name:vendorSpecific" />

</Device>

<!--FT4232H Device-->

<Device Id="vidpid:0403 6011">

<Function Type="name:vendorSpecific" />

</Device>

<!--FT232H Device-->

<Device Id="vidpid:0403 6014">

<Function Type="name:vendorSpecific" />

</Device>

<!--FT-X-Series Devices-->

<Device Id="vidpid:0403 6015">

<Function Type="name:vendorSpecific" />

</Device>

<!--My Custom Device-->

<!--<Device Id="vidpid:1234 4321">

<Function Type="name:vendorSpecific" />

</Device>-->

</DeviceCapability>

</Capabilities>

**Sviluppare una UWP IoT App per Raspberry Pi2**

Dopo aver lanciato Visual Studio 2015, e creato una **Universal Blank Application**, ricordarsi di includere i riferimenti alle **IoT Extensions**

