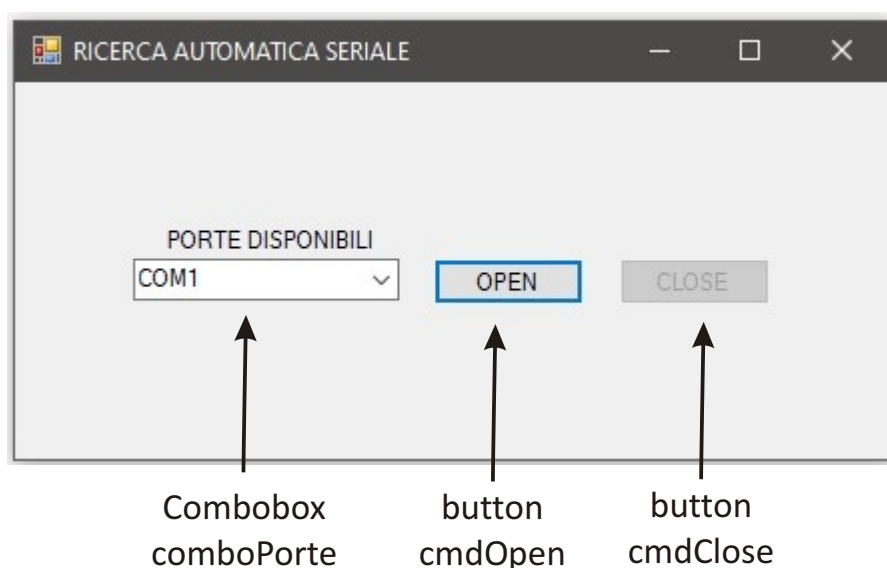


# L@BOROBOTICA

## SCHEDE DI APPROFONDIMENTO

### VISUAL C#: RICERCA AUTOMATICA DELLE PORTE SERIALI

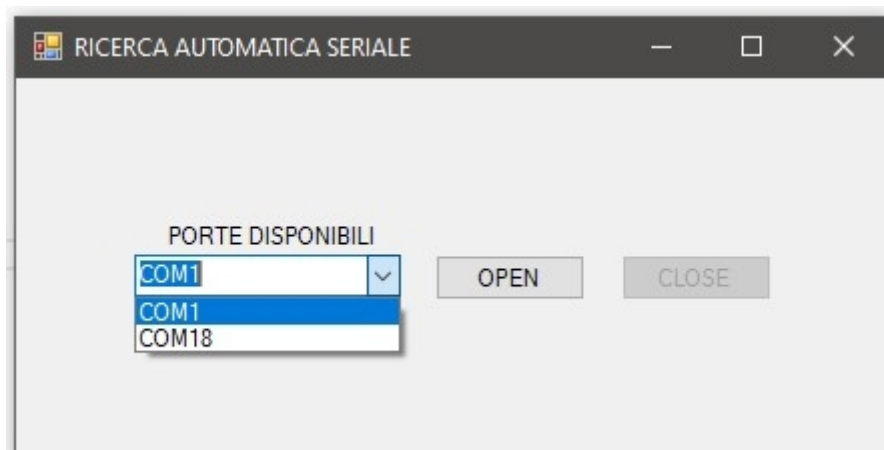
Quando si utilizza Visual C# collegato via seriale con Arduino, ci si trova sempre a non dover conoscere la porta di connessione del microcontrollore. In questa lezione approfondiremo alcune righe di codice che, messe all'inizio di un qualsiasi nostro programma ci faciliteranno molto la vita, individuando tutte le porte seriali connesse e risparmiandoci inutili manovre attacca-stacca con Arduino.



Per fare questo, ci serviranno una ComboBox e due buttons. Ricordiamoci anche di inserire lo strumento SerialPort1 e di settare il baud rate a 9600.

Il software che creeremo, andrà a verificare la presenza di porte seriali, le inserirà in un array e le renderà disponibili nella combobox.

Ecco un esempio del funzionamento; il software ha individuato la COM1 (relativa al desktop) e la COM18 (relativa ad un Arduino collegato ad una USB del computer). A noi basterà selezionare la porta che ci interessa e con il button OPEN aprirla per la comunicazione.



## ED ECCO IL PROGRAMMA LATO VISUAL C#

```
using System;
using System.Windows.Forms;
using System.IO.Ports; //necessario per disporre della classe SerialPort

namespace ricerca_porte
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }
        string[] porte;
        int numPorte = 0; //numero delle porte (per adesso 0)

        private void Form1_Load(object sender, EventArgs e) //per ottenere questo void doppio click sul form
        {
            porte = SerialPort.GetPortNames(); //cerca le porte disponibili e le carica nell'array
            foreach(string s in porte) //per ogni porta trovata (s)
            {
                comboPorte.Items.Add(s); //caricala nel combobox
                numPorte = numPorte + 1; //incrementa il contatore delle porte
            }

            if (numPorte != 0) //se ? stata trovata almeno una porta
            {
                comboPorte.SelectedIndex = 0; //seleziona la prima porta
                cmdOpen.Enabled = true; //abilita il button di apertura porta
            }
            else
            {
                cmdOpen.Enabled = false; //disabilita il button
            }
        }

        private void cmdOpen_Click(object sender, EventArgs e)
        {
            serialPort1.PortName = comboPorte.Text; //seleziona la porta dal combo box
            serialPort1.Open(); //apre la porta seriale
            cmdOpen.Enabled = false; //disabilita il pulsante open
            cmd_Close.Enabled = true; //abilita il pulsante close
            comboPorte.Enabled = false; //disabilita il comboporte
        }

        private void cmd_Close_Click(object sender, EventArgs e)
        {
            serialPort1.Close(); //apre la porta seriale
            cmdOpen.Enabled = true;
            cmd_Close.Enabled = false;
            comboPorte.Enabled = true;
        }
    }
}
```