# C# Code Example for Serial Communication with Mark-10 Instruments

Below is example Microsoft C#.net code to write and read a serial port (including a USB Virtual COM Port) to communicate with Mark-10 instruments:

```csharp
// Example C#.net code for writing and reading a serial port
// Can wrap the writes and reads into separate, generalized functions,
// for example, "WriteCommand(string command)" and "string ReadResponse()".
// First open the serial port (baud rate, etc.) configured at design or run time.
// It is good practice to use try-catch blocks for exception handling.
serialPort1.Open();
if (serialPort1.IsOpen) // serialPort1 is the name of the serial port component
{
try // In case there is a read timeout or other exception occurs
{
// Write the command (refer to GCL2 commands in the instrument's User Guide)
// followed by a carriage return character (0xD)
serialPort1.Write("?\r");
// Read the response. ReadLine() will wait (for a timeout) until the termination
// character (LF (or "new line") = "\n" = 0xA by default) is received.
string response = serialPort1.ReadLine().Trim();
// Typical response from a Mark-10 gauge: "3.258 lbF\r\n" so need to parse
// the string to get the floating-point numeric value and the unit.
// For example:
// Can parse a string on a space as follows:
// string[] str = response.Split(' '); // str[0] is the load, str[1] is the unit
// Convert the numeric ASCII part of the string to a float or double:
//double d_number, load;
//if (double.TryParse(str[0], out d_number))
// load = d_number;
//else
// // Do whatever error handling is desired.
}
catch (Exception ex) // Could look for a TimeoutException explicitly
{
Debug.Print("Serial Port Exception: " + ex.Message); // For example
}
}
else
{
// Do whatever action, such as inform the user that the serial port is not open.
}
```

More information and examples may be found in Visual Studio's Help section and online resources.

Please contact our Technical Support or Engineering staff for any needed assistance.



Force and torque measurement engineered better