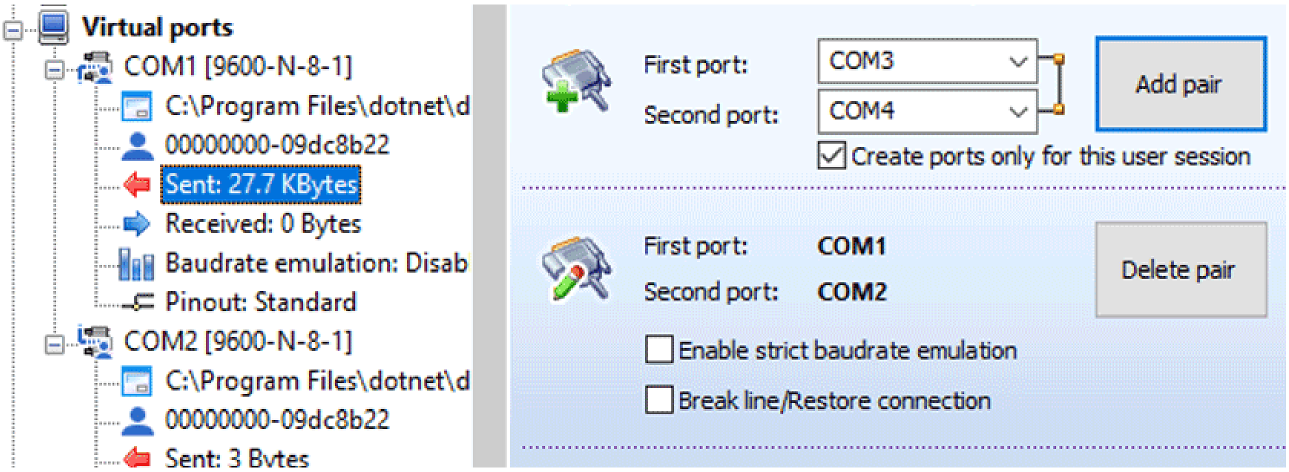
Maytham Fahmi

A Promise of Excellence, Software Developer

How to Create virtual COM port pairs in Windows for development (IoT)



Virtual ports
- COM1 [9600-N-8-1]
  - C:\Program Files\dotnet\d
  - 00000000-09dc8b22
  - Sent: 27.7 KBytes
  - Received: 0 Bytes
  - Baudrate emulation: Disab
  - Pinout: Standard
- COM2 [9600-N-8-1]
  - C:\Program Files\dotnet\d
  - 00000000-09dc8b22
  - Sent: 3 Bytes

First port: COM3
Second port: COM4
Add pair
☑ Create ports only for this user session

First port: COM1
Second port: COM2
Delete pair
☐ Enable strict baudrate emulation
☐ Break line/Restore connection

Just imagine you are developing embedded device or IoT device solution where you need to interact your software with a hardware device over Com port.

Normally you need to have your real device turn on and connect it to your development PC before you can interact with it. The challenge is that you might developing your software and the embedded prototype is not ready yet, you are waiting for it, you need to testing it locally before connecting to a real device or just to emulate the data communication before developing embedded prototype. Then your only way is to create a virtual Com port.

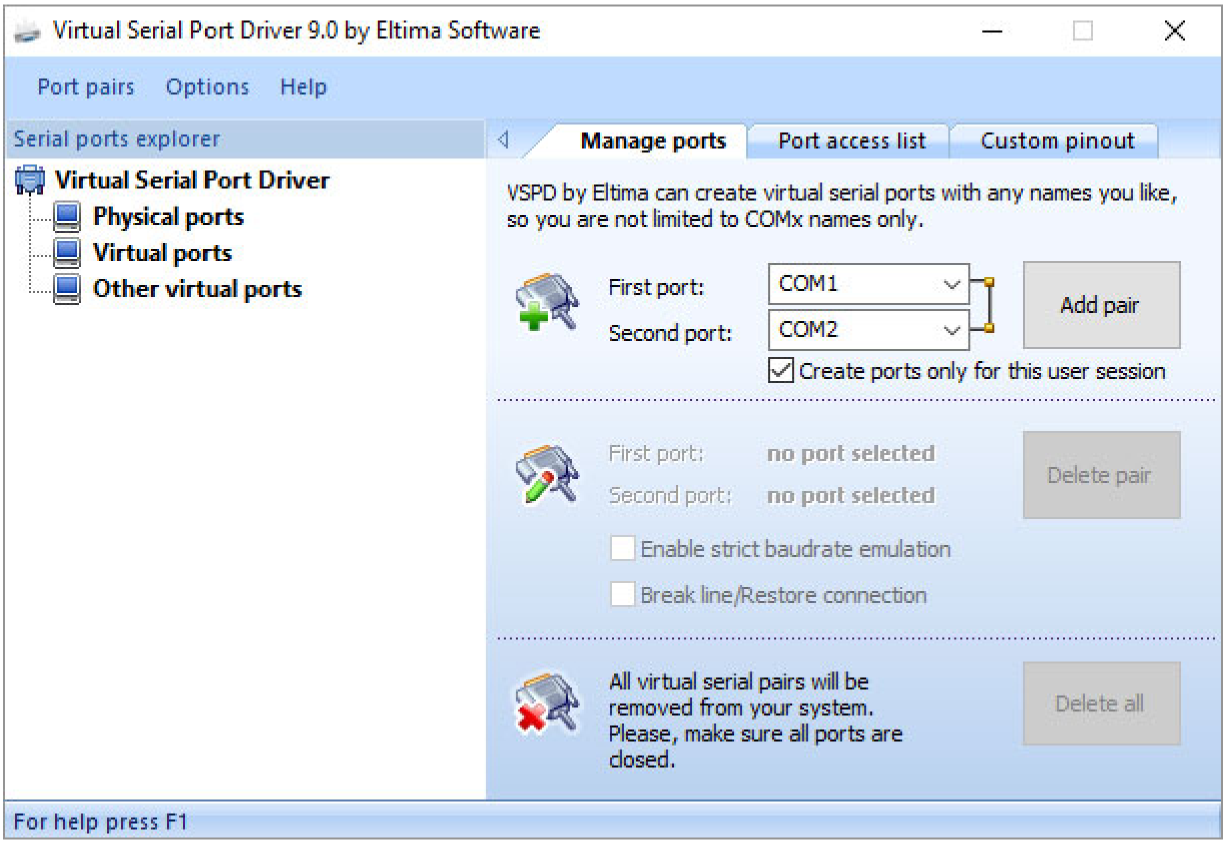
In this article I am going to use Virtual Serial Port Driver 9.0 from Eltima's software.

I will demonstrate how to use Virtual Serial Port Driver 9.0 using two different examples:

1. Chat example based on C#, using Microsoft System.IO.Ports example.
2. Using C# to emulate embedded device that sending sensor data using and receive the data.

Before we dig into examples, let download Virtual Serial Port 9.0 standard edition from https://www.eltima.com/products/vspdxp.

Install the software and start it.



VSPD by Eltima can create virtual serial ports with any names you like, so you are not limited to COMx names only.

First port: COM1

Second port: COM2

☑ Create ports only for this user session

First port: no port selected

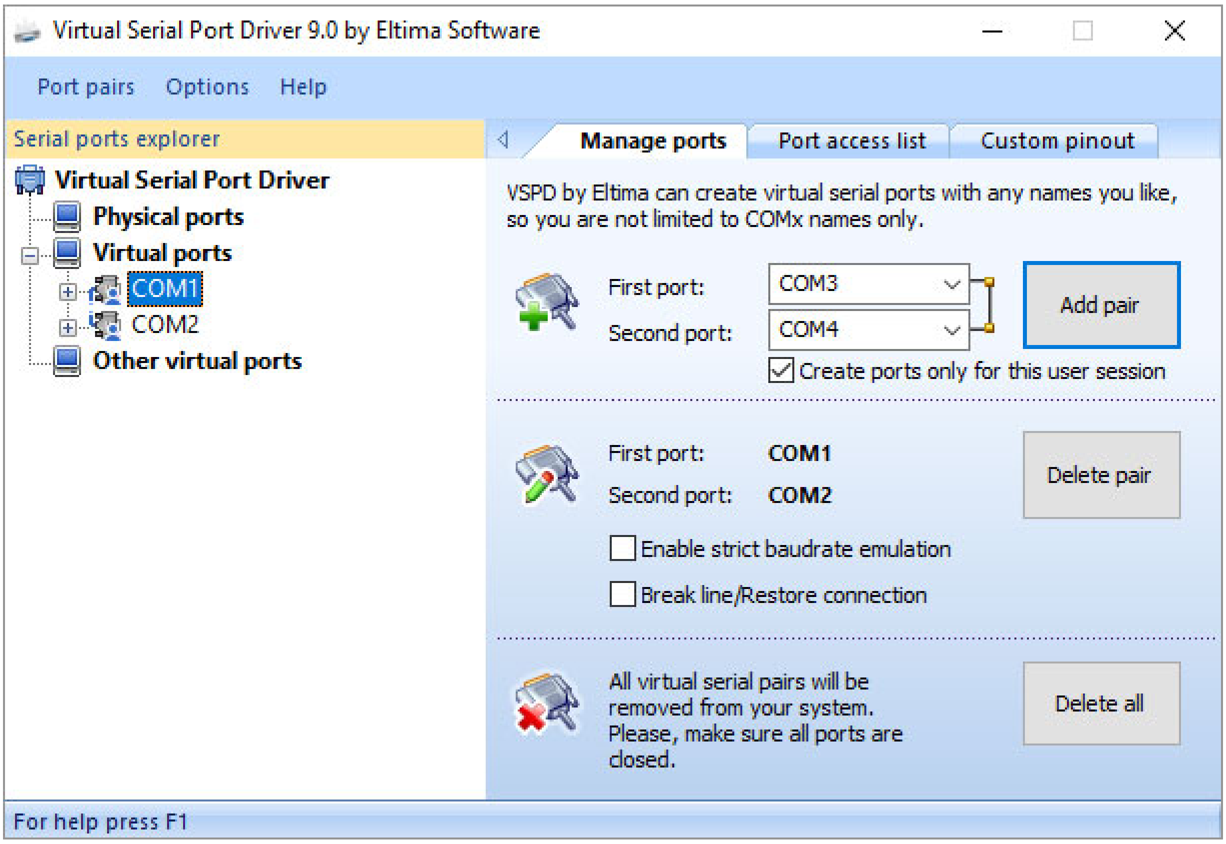
Second port: no port selected

□ Enable strict baudrate emulation

□ Break line/Restore connection

All virtual serial pairs will be removed from your system. Please, make sure all ports are closed.

Click on **Add Pair**



VSPD by Eltima can create virtual serial ports with any names you like, so you are not limited to COMx names only.

First port: COM3
Second port: COM4
☑ Create ports only for this user session

First port: COM1
Second port: COM2
□ Enable strict baudrate emulation
□ Break line/Restore connection

All virtual serial pairs will be removed from your system. Please, make sure all ports are closed.

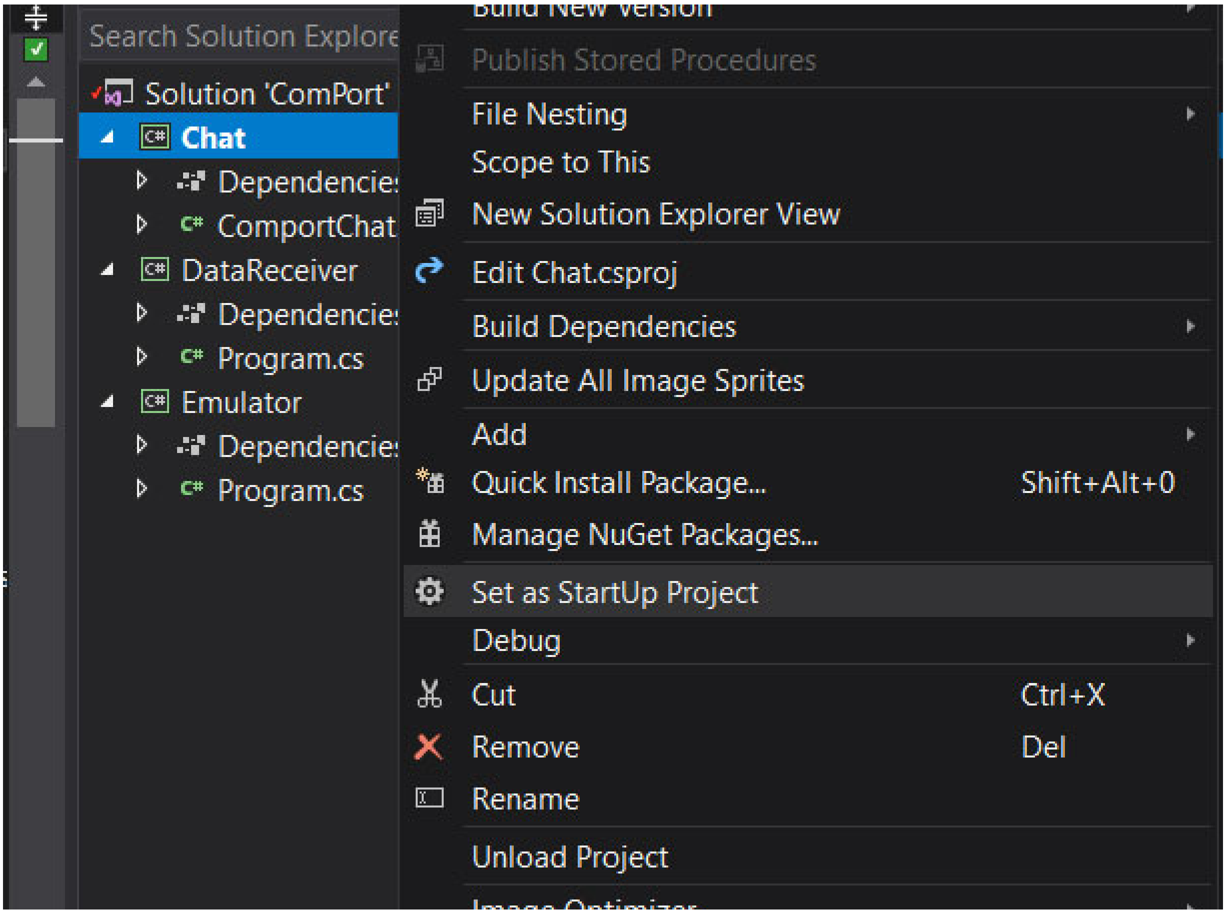
And Ya, That’s it, now you have 2 com virtual ports **COM1** and **COM2**.

> *I found Virtual Serial Port Driver 9.0 one of the easiest and most stable virtual serial port software on market. It allows me to develop and test my code before spending hours on a real device testing.*

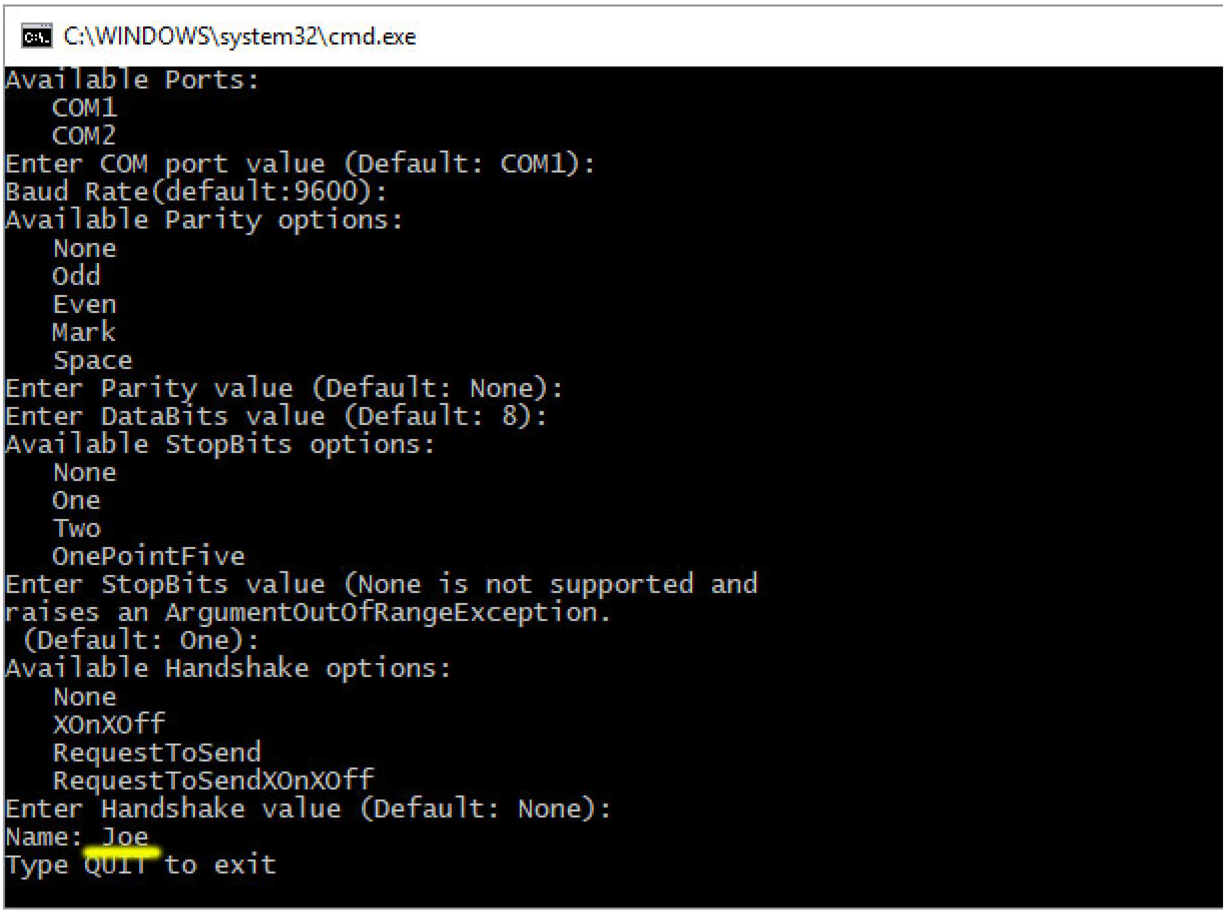
Example 1

I have visited Microsoft SerialPort Class (Sytem.IO.Ports) page and found a chat example. I have created visual studio solution that contains all example of this article on my git ComPort. Clone it and fire up your Visual Studio with the solution twice (I do used VS2017) and yes, I mean start Visual Studio two times because we need to have 2 chats to connect to each other.

Make sure your Chat project is set as Start-up project on both VS.

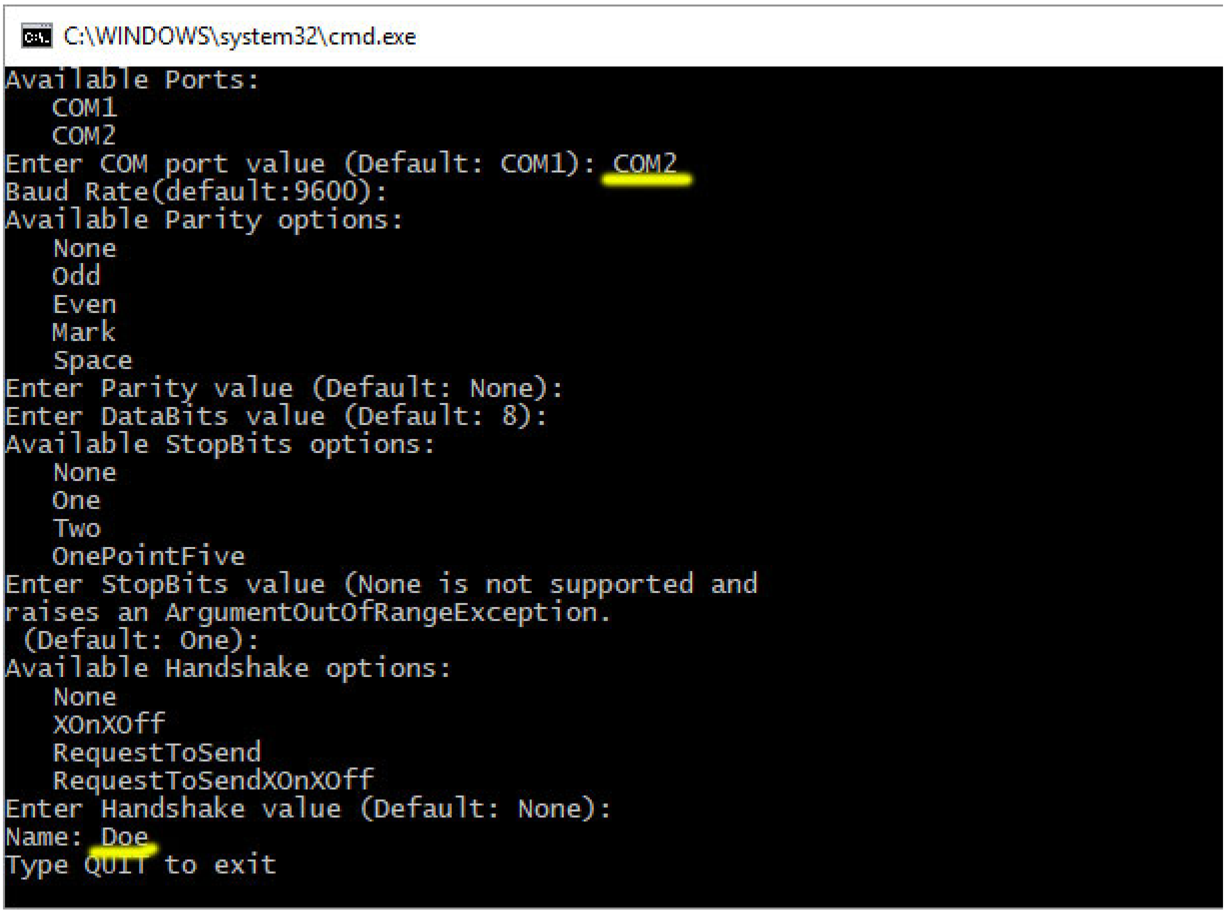


Start the program (ctrl + F5) on your first VS and your console will start asking you questions, just press ENTER (leave empty) until you reach the name, just give any name, I call my first chatter Joe as you can see in the image below.


C:\WINDOWS\system32\cmd.exe

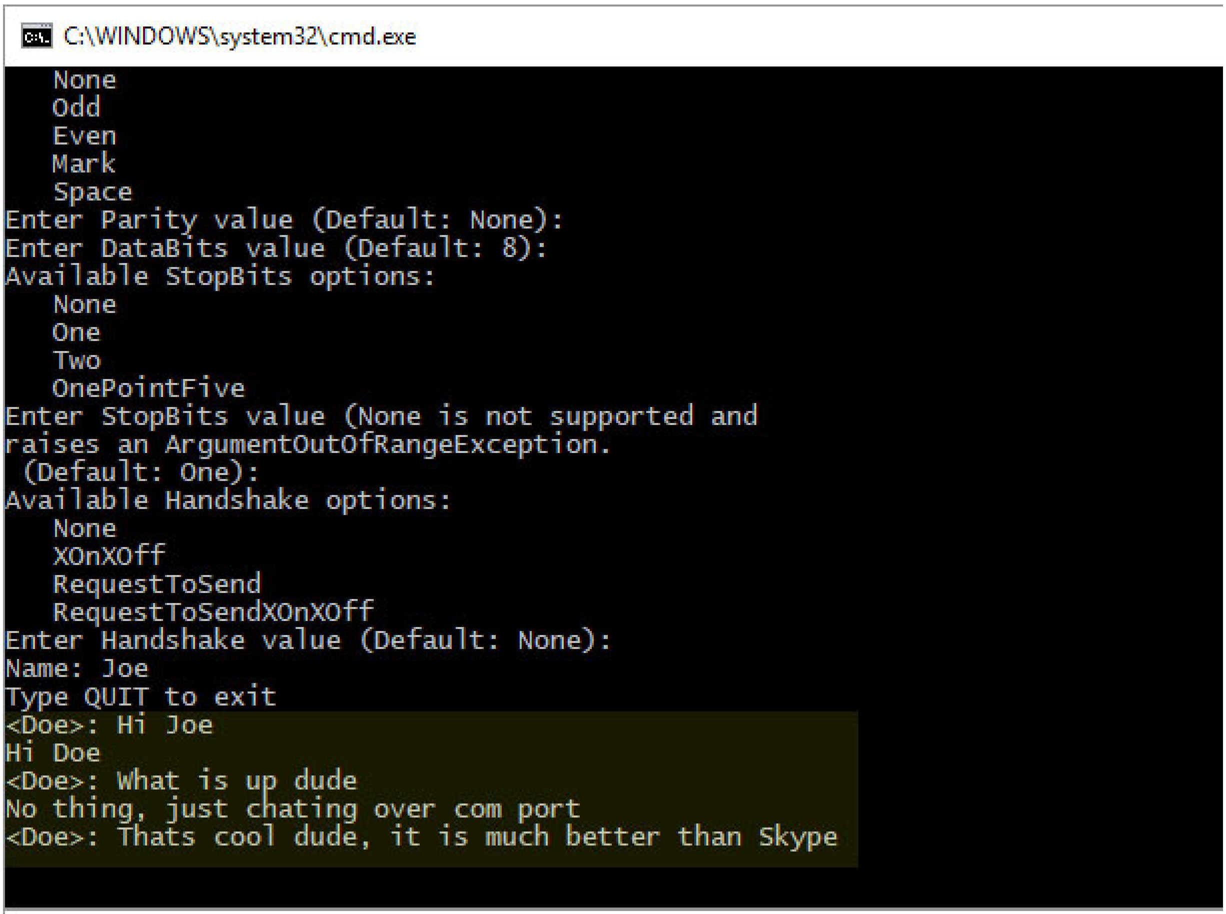
```
Available Ports:
   COM1
   COM2
Enter COM port value (Default: COM1):
Baud Rate(default:9600):
Available Parity options:
   None
   Odd
   Even
   Mark
   Space
Enter Parity value (Default: None):
Enter DataBits value (Default: 8):
Available StopBits options:
   None
   One
   Two
   OnePointFive
Enter StopBits value (None is not supported and
raises an ArgumentOutOfRangeException.
 (Default: One):
Available Handshake options:
   None
   XOnXOff
   RequestToSend
   RequestToSendXOnXOff
Enter Handshake value (Default: None):
Name: Joe
Type QUIT to exit
```

Start your second console program, this time you need to tell the program that you are using **COM2** as shown in the image, and tap ENTER until the name and just give it any name, I call my second chatter Doe as you can see in the image below.

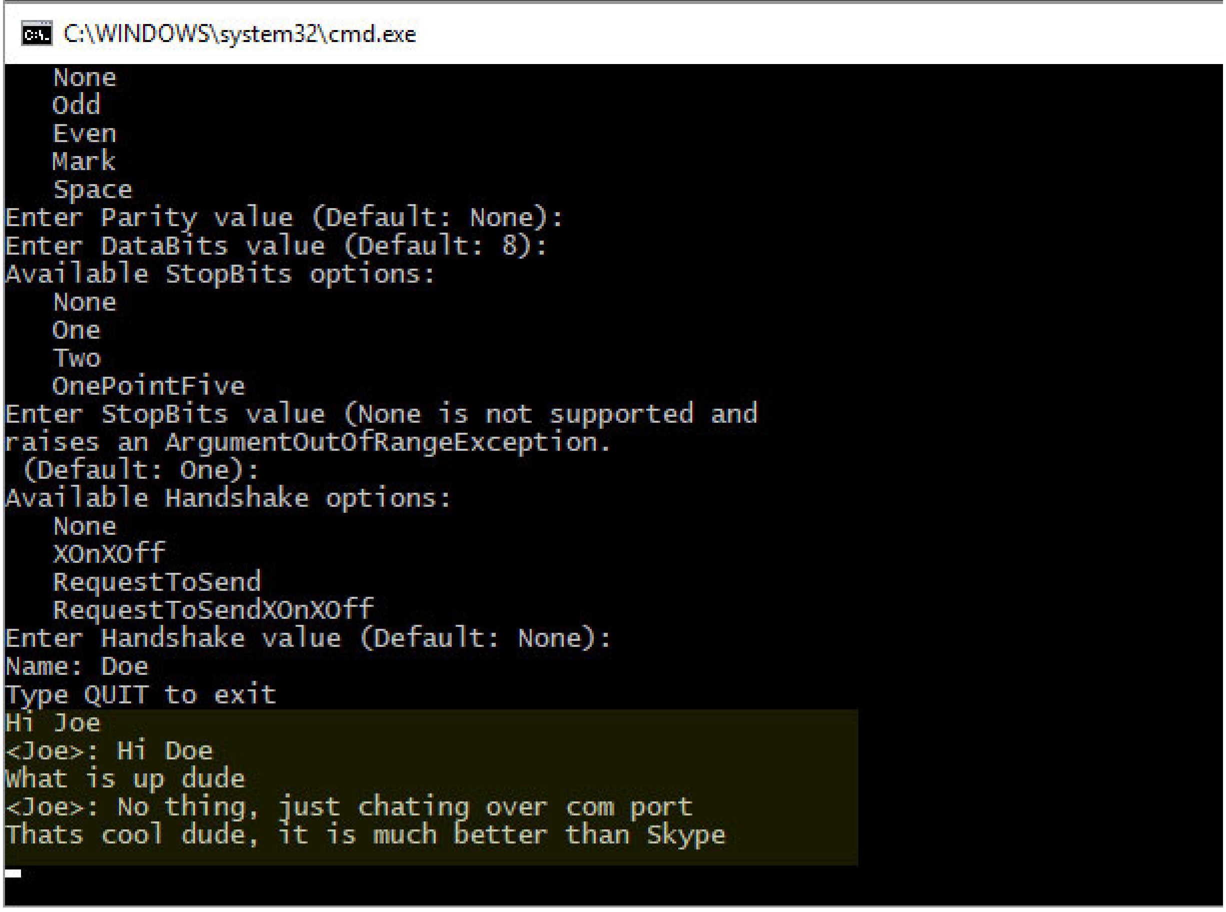


```
Available Ports:
   COM1
   COM2
Enter COM port value (Default: COM1): COM2
Baud Rate(default:9600):
Available Parity options:
   None
   Odd
   Even
   Mark
   Space
Enter Parity value (Default: None):
Enter DataBits value (Default: 8):
Available StopBits options:
   None
   One
   Two
   OnePointFive
Enter StopBits value (None is not supported and
raises an ArgumentOutOfRangeException.
 (Default: One):
Available Handshake options:
   None
   XOnXOff
   RequestToSend
   RequestToSendXOnXOff
Enter Handshake value (Default: None):
Name: Doe
Type QUIT to exit
```

Now Joe and Doe can chat between each other.



```
   None
   Odd
   Even
   Mark
   Space
Enter Parity value (Default: None):
Enter DataBits value (Default: 8):
Available StopBits options:
   None
   One
   Two
   OnePointFive
Enter StopBits value (None is not supported and
raises an ArgumentOutOfRangeException.
 (Default: One):
Available Handshake options:
   None
   XOnXOff
   RequestToSend
   RequestToSendXOnXOff
Enter Handshake value (Default: None):
Name: Joe
Type QUIT to exit
<Doe>: Hi Joe
Hi Doe
<Doe>: What is up dude
No thing, just chating over com port
<Doe>: Thats cool dude, it is much better than Skype
```



```
   None
   Odd
   Even
   Mark
   Space
Enter Parity value (Default: None):
Enter DataBits value (Default: 8):
Available StopBits options:
   None
   One
   Two
   OnePointFive
Enter StopBits value (None is not supported and
raises an ArgumentOutOfRangeException.
 (Default: One):
Available Handshake options:
   None
   XOnXOff
   RequestToSend
   RequestToSendXOnXOff
Enter Handshake value (Default: None):
Name: Doe
Type QUIT to exit
Hi Joe
<Joe>: Hi Doe
What is up dude
<Joe>: No thing, just chating over com port
Thats cool dude, it is much better than Skype
```

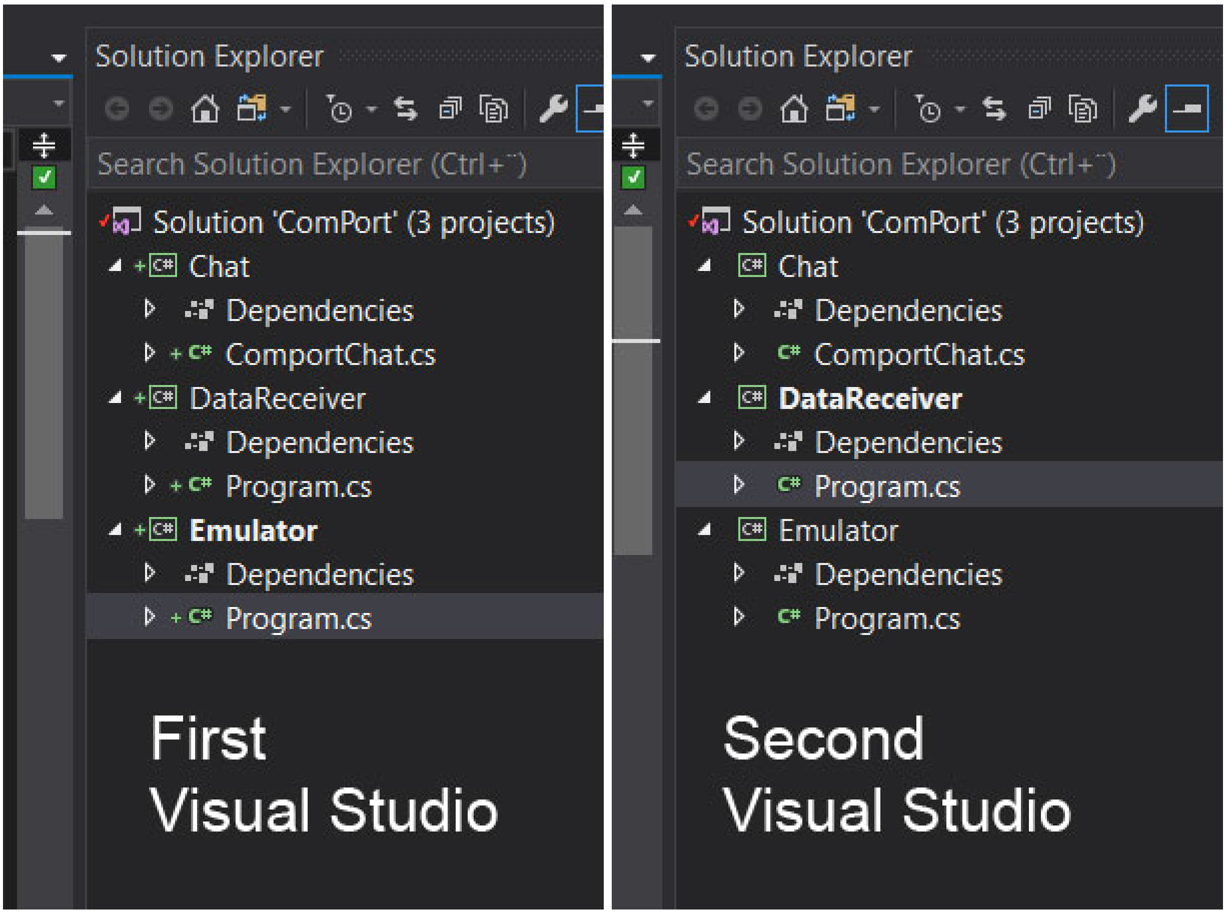
Just imagine if this virtual communication was made with a real device that can receive messages from your system or the other way around, as we gone demonstrate on the next example.

Example 2

In this example, I am going to use one port to sending some random sensor data, and I will fetch the data on the other port.

This example is based on real scenario that I have used previous in one of my research and development projects. I was developing a hand gesture solution and my only prototype was not available, So I designed a software that emulate the data and send it as it was the hand gesture, but over my virtual com port.

Now in stead of have Chat as start-up project, set your first Visual Studio start-up project to Emulator and the second one to DataReceiver.



Solution Explorer

Search Solution Explorer (Ctrl+¨)

```
Solution 'ComPort' (3 projects)
  Chat
    Dependencies
    ComportChat.cs
  DataReceiver
    Dependencies
    Program.cs
  Emulator
    Dependencies
    Program.cs
```

First Visual Studio

Solution Explorer

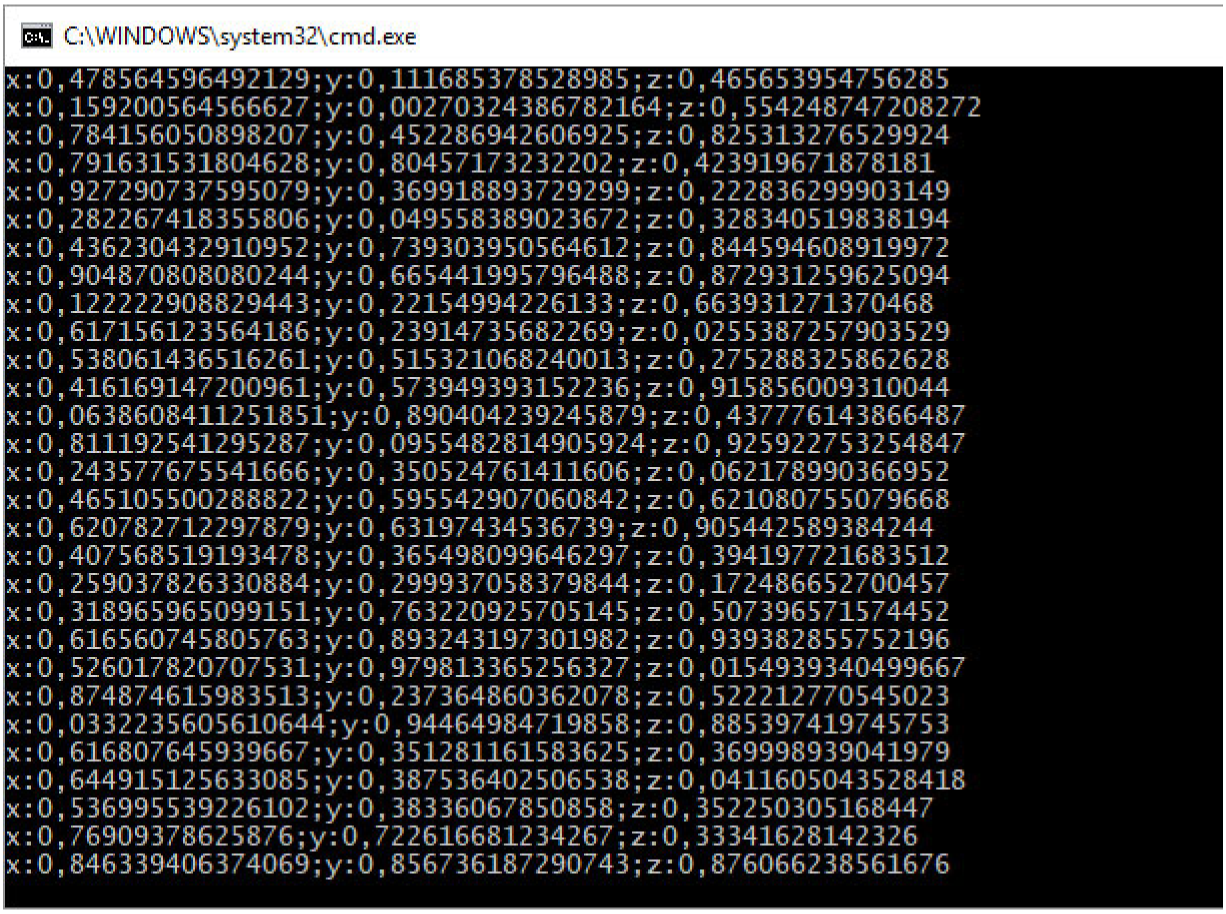
Search Solution Explorer (Ctrl+¨)

```
Solution 'ComPort' (3 projects)
  Chat
    Dependencies
    ComportChat.cs
  DataReceiver
    Dependencies
    Program.cs
  Emulator
    Dependencies
    Program.cs
```

Second Visual Studio

Now start-up your Emulator program (ctrl + F5) nothing should happen.

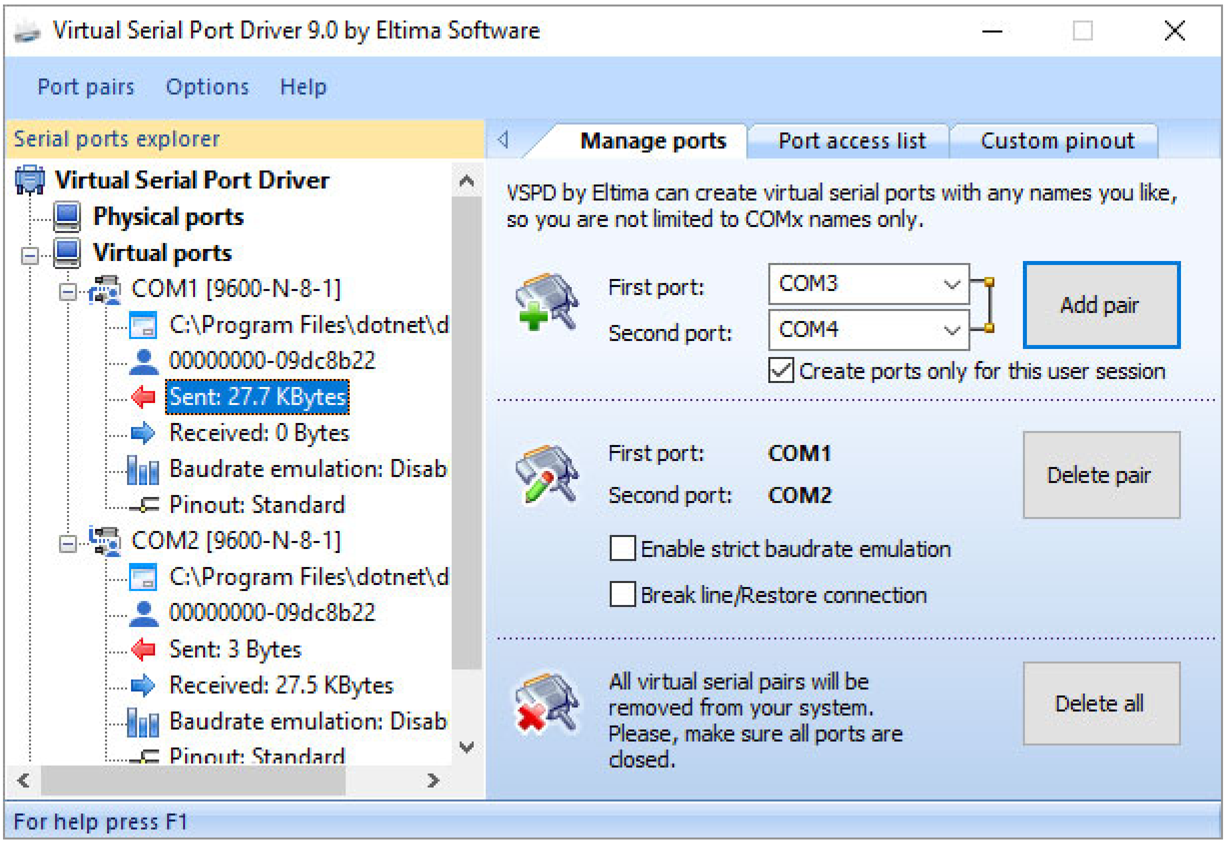
On your second Visual Studio start DataReceiver program and you should see that Emulator program is sending x, y, z data and it being collected by DataReceiver program.



C:\WINDOWS\system32\cmd.exe

```
x:0,478564596492129;y:0,111685378528985;z:0,465653954756285
x:0,159200564566627;y:0,00270324386782164;z:0,554248747208272
x:0,784156050898207;y:0,452286942606925;z:0,825313276529924
x:0,791631531804628;y:0,80457173232202;z:0,423919671878181
x:0,927290737595079;y:0,369918893729299;z:0,222836299903149
x:0,282267418355806;y:0,049558389023672;z:0,328340519838194
x:0,436230432910952;y:0,739303950564612;z:0,844594608919972
x:0,904870808080244;y:0,665441995796488;z:0,872931259625094
x:0,122229088294443;y:0,22154994226133;z:0,663931271370468
x:0,617156123564186;y:0,23914735682269;z:0,0255387257903529
x:0,538061436516261;y:0,515321068240013;z:0,275288325862628
x:0,416169147200961;y:0,573949393152236;z:0,915856009310044
x:0,0638608411251851;y:0,890404239245879;z:0,437776143866487
x:0,811192541295287;y:0,0955482814905924;z:0,925922753254847
x:0,243577675541666;y:0,350524761411606;z:0,062178990366952
x:0,465105500288822;y:0,595542907060842;z:0,621080755079668
x:0,620782712297879;y:0,63197434536739;z:0,90544258938424
x:0,407568519193478;y:0,365498099646297;z:0,394197721683512
x:0,259037826330884;y:0,299937058379844;z:0,172486652700457
x:0,318965965099151;y:0,763220925705145;z:0,507396571574452
x:0,616560745805763;y:0,893243197301982;z:0,939382855752196
x:0,526017820707531;y:0,979813365256327;z:0,0154939340499667
x:0,874874615983513;y:0,237364860362078;z:0,522212770545023
x:0,0332235605610644;y:0,944649847198858;z:0,885397419745753
x:0,616807645939667;y:0,351281161583625;z:0,369998939041979
x:0,644915125633085;y:0,387536402506538;z:0,0411605043528418
x:0,536995539226102;y:0,38336067850858;z:0,352250305168447
x:0,76909378625876;y:0,722616681234267;z:0,33341628142326
x:0,846339406374069;y:0,856736187290743;z:0,876066238561676
```

You can also see the amount of data being sent and recevied by ports.



Virtual Serial Port Driver 9.0 by Eltima Software

Virtual Serial Port Driver
Physical ports
Virtual ports
COM1 [9600-N-8-1]
C:\Program Files\dotnet\d
00000000-09dc8b22
Sent: 27.7 KBytes
Received: 0 Bytes
Baudrate emulation: Disab
Pinout: Standard
COM2 [9600-N-8-1]
C:\Program Files\dotnet\d
00000000-09dc8b22
Sent: 3 Bytes
Received: 27.5 KBytes
Baudrate emulation: Disab
Pinout: Standard

VSPD by Eltima can create virtual serial ports with any names you like, so you are not limited to COMx names only.

First port: COM3
Second port: COM4
☑ Create ports only for this user session

First port: COM1
Second port: COM2
☐ Enable strict baudrate emulation
☐ Break line/Restore connection

All virtual serial pairs will be removed from your system. Please, make sure all ports are closed.

Now you should be able to parse your received data and make your programming logic with out having a real device.

Links:

- https://github.com/maythamfahmi/ComPort
- Virtual Serial Port Driver 9.0

Maytham Fahmi / 07-07-2018 / Article, Internet of Things (IoT) / Create virtual COM port pairs in Windows, eltima, emulator, IoT, Virtual Serial Port Driver

One thought on “How to Create virtual COM port pairs in Windows for development (IoT)”

Peter Martenson

22-01-2019 at 22:31

Thanks, this was useful to kick start my project.

Comments are closed.

Maytham Fahmi / Disclaimer / Proudly powered by WordPress / Customized by BoldThemes