

How to communicate with Mirrorbow USB25 I/O card using Access or Excel VBA

Firstly Microsoft Visual Basic for Applications which is bundled with MS Access, MExcel and MSWord does not have any built in comms handling commands.

So the first thing we have to do is download MSComm32.ocx which is available from here:

<http://www.yes-tele.com/mscomm.html>

Download to your System32 folder in the Windows or Winnt folder

TAKE CARE WHEN CHANGING THE REGISTRY USING REGEDIT.

Install the download by going to Run or the command prompt and typing

To do this Click **Start**, and then click **Run**. In the **Run** dialog box, type **regedit**, and then click **OK**.

Regsvr32 MSComm32.ocx

Navigate to the HKEY_CLASSES_ROOT\Licenses folder

This will register the file onto the windows operating system.

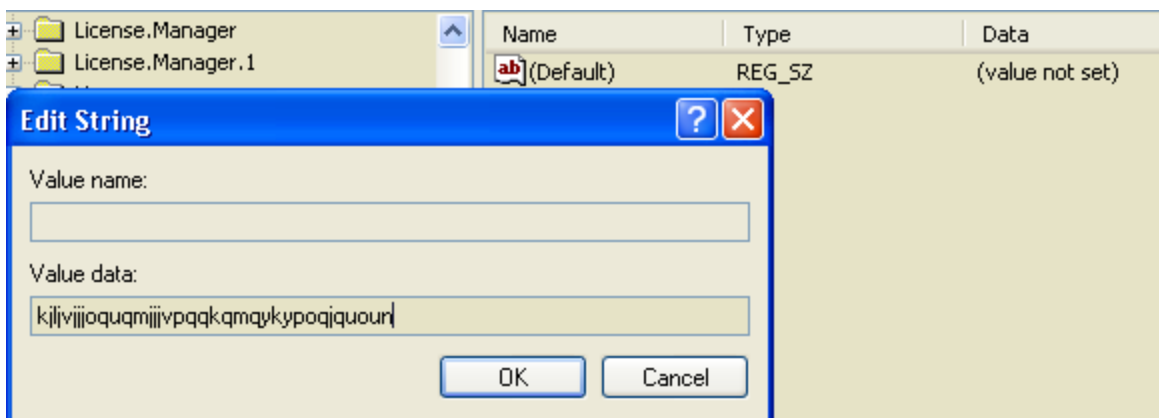
As you will see on the above site, you will need the ActiveX control liscensing for which they give the key.

Now choose Edit – New – Key and then rename the new key to

4250E830-6AC2-11cf-8ADB-00AA00C00905

Now Double Click on (Default) in the name column and under Value data type

kjljvjjoquqmjjjvpqqkqmkykypoqjquoun



Now close Regedit and your Control is licensed.

From Access or Excel Go into the Visual Basic Editor and choose Tools – References and then click Browse.

Navigate to your Sytem32 folder and change the Files of Type box to ActiveX Controls (*.ocx) Then double click on MSCOMM32.OCX. Then click OK. Now when you go into form design, if you click on the Toolbox and then More controls (another tool icon) you should see Microsoft Communications Control version 6.0, This must be placed on the form that contains the visual basic for the communication program.

The control will not be seen at run time. You should rename this control to MSComm1 or similar.

The commands to use in visual basic are all available on the above website. 112500 can be used as the Baud Rate.

Please Note: Whenever you send an output using MSComm1.Output = “OUT1 FF” for example, you must then immediately add a line below to accept the acknowledge back as an input using

```
I$ = MSComm1.Input (For example)
```

Here are some sample routines

```
Sub InitialiseIOBoard(PortNumber) ‘ PortNumber is the number of the Comm Port assigned to the I/O board
```

```
MDComm1.CommPort = PortNumber
```

```
MSComm1.DTREnable=False ‘Turns off DTR as it is not used in USB I/O
```

```
MSComm1.RTSEnable=False ‘Turns off RTS as it is not used in USB I/O
```

```
MSComm1.Settings=”112500,N,8,1” ‘Sets Baud to 112500, No Parity, 8 bits and 1 Stop bit
```

```
MSComm1.InputLen=0 ‘Causes any input statement to receive the entire buffer
```

```
End Sub
```

```
Sub SetAllPortsToOutput()
```

```
MSComm1.PortOpen = True
```

```
MSComm1.Output = “DIR1 00”
```

```
I$ = MSComm1.Input
```

```
MSComm1.Output = “DIR2 00”
```

```
I$ = MSComm1.Input
```

```
MSComm1.Output = “DIR3 00”
```

```
I$ = MSComm1.Input
```

```
MSComm1.PortOpen = False
```

```
End Sub
```

```
Sub OutputToPorts (Val1, Val2, Val3) ‘ Values must be a Hex String
```

```
MSComm1.PortOpen = True
```

```
MSComm1.Output = “OUT1 ” & Val1
```

```
I$ = MSComm1.Input
```

```
MSComm1.Output = “OUT 2 ” & Val2
```

```
I$ = MSComm1.Input
```

```
MSComm1.Output = “OUT 3 ” & Val3
```

```
I$ = MSComm1.Input
```

```
MSComm1.PortOpen = False
```

```
End Sub
```