

[www.vskom.de](http://www.vskom.de)

# User Manual VScom PCI Cards VScom H-Series Card

Edition: February 2010



Tel: +49 40 528 401 0

Fax: +49 40 528 401 99

Web: [www.visionsystems.de](http://www.visionsystems.de)

Support: [service@visionsystems.de](mailto:service@visionsystems.de)

---

The software described in this manual is furnished under a license agreement and may be used only in accordance with the terms of that agreement.

## **Copyright Notice**

Copyright © 2009 Vision Systems. All rights reserved. Reproduction without permission is prohibited.

## **Trademarks**

VScom is a trademark of Vision Systems GmbH. All other trademarks and brands are property of their rightful owners.

## **Disclaimer**

Vision Systems reserves the right to make changes and improvements to its product without providing notice.

Vision Systems provides this document “as is”, without warranty of any kind, either expressed or implied, including, but not limited to, its particular purpose. Vision Systems reserves the right to make improvements and/or changes to this manual, or to the products and/or the programs described in this manual, at any time.

Information provided in this manual is intended to be accurate and reliable. However, Vision Systems assumes no responsibility for its use, or for any infringements on the rights of third parties that may result from its use.

This product might include unintentional technical or typographical errors. Changes are periodically made to the information herein to correct such errors, and these changes are incorporated into new editions of the publication.

## Contents

<b>1</b>	<b>Overview</b>	<b>6</b>
<b>2</b>	<b>Introduction</b>	<b>6</b>
2.1	Features	6
2.2	Product Specifications	7
2.2.1	VScom 200H UPCI	7
2.2.2	VScom 400H UPCI	8
2.2.3	VScom 800H UPCI	9
2.2.4	VScom 011H UPCI	10
2.2.5	VScom 110H PCI	11
2.2.6	VScom 210H PCI	12
2.2.7	VScom 410H PCI	13
2.2.8	VScom 400H RJ45 PCI	14
2.2.9	VScom 021H PCI	15
2.3	Packing List	15
2.4	About this Manual	16
<b>3</b>	<b>Hardware Configuration</b>	<b>16</b>
3.1	VScom 200H UPCI	16
3.2	VScom 400H UPCI	17
3.3	VScom 800H UPCI	17
3.4	VScom 210H PCI	17
3.5	VScom 410H PCI	17
3.6	VScom 400H RJ45 PCI	17
3.7	VScom 021H PCI	18
<b>4</b>	<b>Windows Driver</b>	<b>19</b>
4.1	Installing the Drivers	19
4.1.1	Install in Windows 9x, 2000, XP and Vista	19
4.1.2	Install in Windows 7	20
4.1.3	Verify the Installation	22
	About assigned port numbers	22
4.1.4	Removing the Software in Windows 9x, 2000, XP and Vista	23
4.1.5	Removing the Software in Windows 7	23
4.2	Card and Port Configuration	25
4.2.1	Card and Ports in Device Manager	25
4.2.2	Card Advanced Settings	27
4.2.3	Ports Configuration	28
	VScom Standard Port Settings	29
	VScom Advanced Port Settings	30
	Enhanced Software Settings	34
	Automatic Software Flow Control	35
	Automatic Hardware Flow Control by RTS/CTS	37
	Automatic Hardware Flow Control by DTR/DSR	38
4.3	Windows NT 4.0	39
4.3.1	Installing in Windows NT 4.0	40
	Automatic Detection of PCI Cards	40

Configuration via Control Panel	41
System Configuration	41
Card and Port Configuration	41
Uninstall Drivers from Windows NT	42
Checking Installation in Windows NT	42
<b>5 Linux Driver</b>	<b>43</b>
5.1 Manual Installation	43
5.2 Automatic Installation	43
5.2.1 Known Cards	43
5.2.2 New Models	44
5.3 Problems since Kernel 2.6	44
5.3.1 Compile a new Kernel	44
5.3.2 Kernel Start Parameters	44
5.4 Card Hardware Parameters	45
<b>6 Connector Definitions</b>	<b>47</b>
6.1 DB62 female	47
6.2 DB37 female	48
6.3 DB9 male	49
6.4 DB25 male RS 232	49
6.5 DB25 female parallel (LPT)	50
6.6 DB25 female RS 232	50
6.7 RJ45 for RS 232	51
<b>7 History</b>	<b>52</b>

## List of Figures

1 VScom 200H UPCI	7
2 VScom 400H UPCI	8
3 VScom 800H UPCI	9
4 VScom 011H UPCI	10
5 VScom 110H PCI	11
6 VScom 210H PCI	12
7 VScom 410H PCI	13
8 VScom 400H RJ45 PCI	14
9 VScom 021H PCI	15
10 Windows 7 Installation	21
11 Driver Installation Target	21
12 Windows 7 Installing Drivers	22
13 Windows 7 Driver Removal	24
14 Windows 7 Drivers removed	25
15 Cards and Ports in Device Manager	26
16 Card Advanced Properties	27
17 Port Renaming	28
18 Standard Settings	29
19 Advanced Settings 16C950	30
20 Advanced Settings 16C550A	32

21	UART Types . . . . .	32
22	Software Settings . . . . .	34
23	Auto XON/XOFF . . . . .	35
24	Auto RTS/CTS . . . . .	37
25	Auto DTR/DSR . . . . .	38
26	Auto-DTR Modes . . . . .	39
27	NT Plug & Play . . . . .	40
28	NT Device list . . . . .	41
29	DB62 female Connector . . . . .	47
30	DB37 female Connector . . . . .	48
31	DB9 male Connector . . . . .	49
32	DB25 male RS 232 . . . . .	49
33	DB25 female LPT . . . . .	50
34	RJ45 for RS 232 . . . . .	51

## List of Tables

1	Features of VScOm UPCI models . . . . .	7
2	Features of VScOm 200H UPCI . . . . .	7
3	Features of VScOm 400H UPCI . . . . .	8
4	Features of VScOm 800H UPCI . . . . .	9
5	Features of VScOm 011H UPCI . . . . .	10
6	Features of VScOm 110H PCI . . . . .	11
7	Features of VScOm 210H PCI . . . . .	12
8	Features of VScOm 410H PCI . . . . .	13
9	Features of VScOm 400H RJ45 PCI . . . . .	14
10	Features of VScOm 021H PCI . . . . .	15
11	JP1 for two Speeds . . . . .	16
12	JP1 for four Speeds . . . . .	17
13	Kernel Compile Options . . . . .	44
14	Hardware Parameters . . . . .	46
15	DB62 female Connector . . . . .	47
16	DB37 female Connector . . . . .	48
17	DB9 male Connector . . . . .	49
18	DB25 male RS 232 . . . . .	49
19	DB25 parallel Connector . . . . .	50
20	DB25 female RS 232 . . . . .	50
21	RJ45 for RS 232 . . . . .	51

## 1 Overview

VScom provides several models of serial port cards for the classic PCI bus. This bus will be replaced by PCIe in the future, but it is still sufficient for many cards with low system demand. Serial and parallel ports are examples of such devices. VScom has developed PCI serial and parallel cards since the PCI bus entered the market, and now offers models ranging from one to eight serial ports. The cards are designed to control industrial hardware via asynchronous serial ports, as many software requires today. There are also models with one and two parallel ports.

The VScom UPCI cards are designed universal, they operate in the classic 5V signaling environment, as well as 3.3V PCI slots. Even the slightly newer PCI-X bus slots may be used to install and operate the card and the ports. Some models are restricted to 5V PCI slots for electrical reasons.

The supplied drivers for Windows operating system install the ports into the Windows API, so they appear as the usual Com and LPT ports software requires. A rich set of driver options is configured via the Device Manager in Windows. The driver allows to configure the serial ports for a bitrate of up to 3.6 Mbps. Of course RS 232 can not reliably transport very high bitrates, even under best conditions 1 Mbit/s is the limit. As the result of the very high maximum bitrate, there is no bitrate below 1 Mbit/s which the cards can not generate. No matter which configuration the hardware requires, the VScom UPCI H-series cards deliver it.

## 2 Introduction

This manual describes the hardware of VScom UPCI H-series cards. UPCI is the name used in VScom products for Universal PCI products. Also the Windows driver for these cards is described in detail.

### 2.1 Features

- Universal PCI 33 MHz
- Serial port interface RS 232
- Max 3.6 Mbps
- Huge FIFO buffers 128 Byte
- Drivers for Windows™ operating system
- Supported by Linux

## 2.2 Product Specifications

The VScom UPCI cards share some common properties, the most important difference is the number of serial and parallel ports. The maximum speed is 3.6 Mbit/s, but this is theoretical because the electrical interface RS 232 is not capable of such transmission rates.

Bus interface	Universal PCI	5V and 3.3V signals, 33MHz
Serial Ports	0 to 8	RS 232
Signals	RS 232	RxD, TxD, RTS, CTS, DTR, DSR, DCD, RI, GND
Max. Bitrates	Theoretical	3.6 Mbit/s
	Effective	1 Mbit/s
Serial configurations	Data bits	5, 6, 7, 8
	Parity	None, Odd, Even, Mark, Space
	Stop bits	1, 1.5, 2
FIFO size	128 Bytes	For transmit and receive each
Parallel Ports	0 to 2	SPP, PS/2, EPP, ECP

Table 1: Features of VScom UPCI models

### 2.2.1 VScom 200H UPCI

The VScom 200H UPCI card features two serial ports, available via a universal PCI slot.



Figure 1: VScom 200H UPCI

Serial Ports	2	RS 232
Connectors	2×DB9 male	Standard (6.3)
Signals	RS 232	RxD, TxD, RTS, CTS, DTR, DSR, DCD, RI, GND

Table 2: Features of VScom 200H UPCI

### 2.2.2 VScom 400H UPCI

The VScom 400H UPCI card features four serial ports, available via a universal PCI slot.

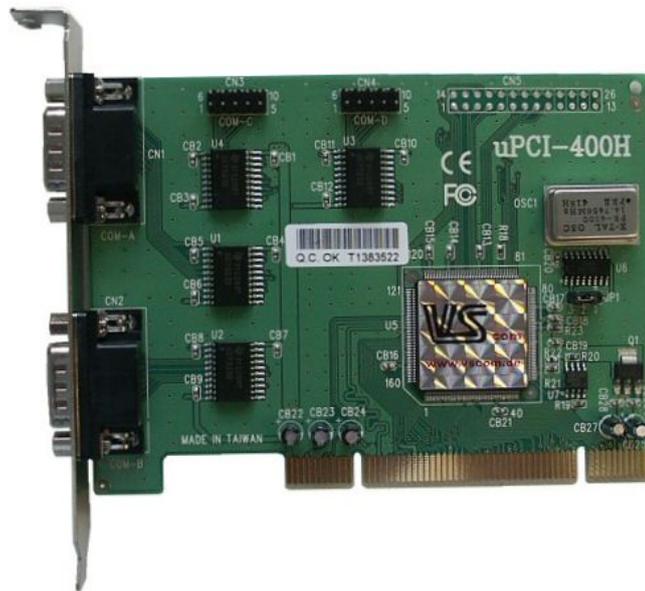


Figure 2: VScom 400H UPCI

A model for Low Profile cases is available.

Serial Ports	4	RS 232
Connectors	4×DB9 male	2 on the card, 2 on extra bracket (6.3)
Signals	RS 232	RxD, TxD, RTS, CTS, DTR, DSR, DCD, RI, GND

Table 3: Features of VScom 400H UPCI

### 2.2.3 VScom 800H UPCI

The VScom 800H UPCI card features eight serial ports, available via a universal PCI slot. All eight serial ports share one common connector DB62-female. For signal assignment see [6.1](#).

An adapter cable to eight times DB9-male is not provided, customers can choose from several connection options. There are Octopus-type cables to DB9 and DB25, male and female. There are also connection boxes for DB9 male, or DB25 male and female. Also converter boxes to RS 422 and RS 485 are possible.

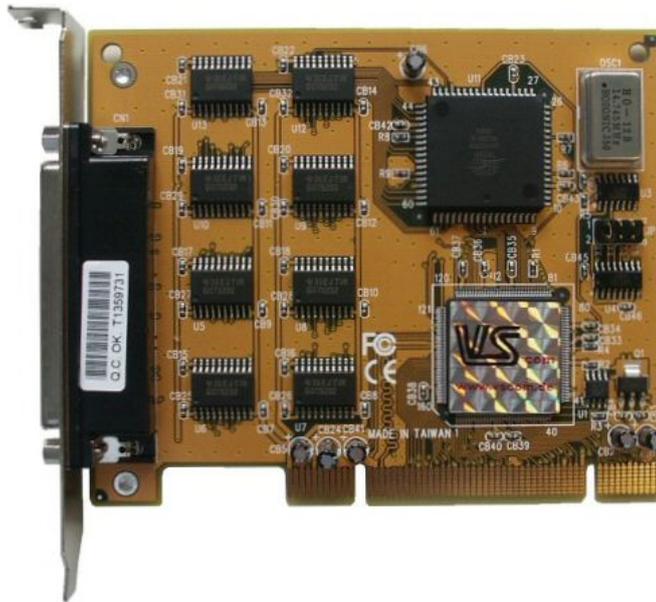


Figure 3: VScom 800H UPCI

A model for Low Profile cases is available.

Serial Ports	8	RS 232
Connectors	1×DB62 female	Adapter not included ( <a href="#">6.1</a> )
Signals	RS 232	RxD, TxD, RTS, CTS, DTR, DSR, DCD, GND

Table 4: Features of VScom 800H UPCI

Note: There is no RI signal, because no pins are available for this on the DB62 connector.

### 2.2.4 VScom 011H UPCI

The VScom 011H UPCI card features one parallel port, available via a universal PCI slot.

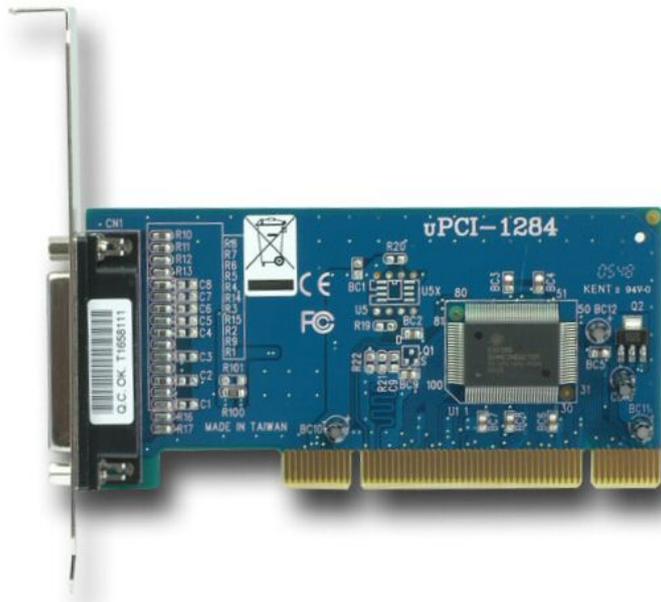


Figure 4: VScom 011H UPCI

A model for Low Profile cases is available.

Parallel Ports	1	SPP, PS/2, EPP, ECP
Connectors	1×DB25 female	
Signals	LPT	Standard for LPT (6.5)

Table 5: Features of VScom 011H UPCI

### 2.2.5 VScom 110H PCI

The VScom 110H PCI card features one serial and one parallel port, available via a classic 5V PCI slot.

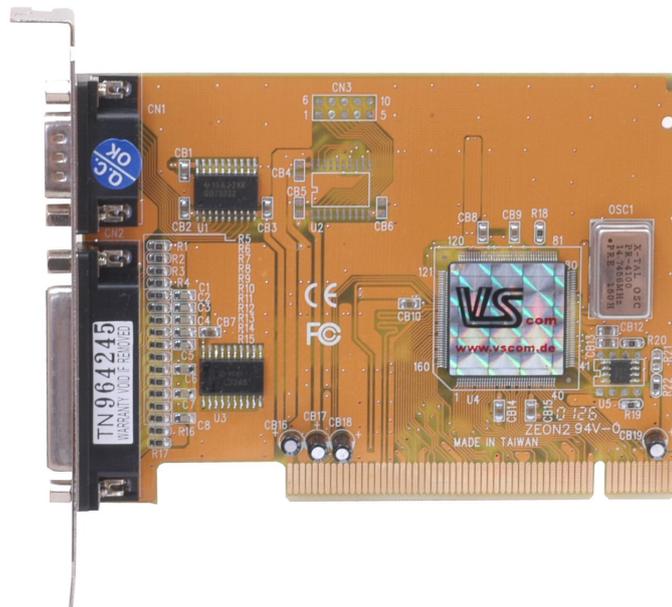


Figure 5: VScom 110H PCI

Bus	5V PCI	Classic slots only
Serial Ports	1	RS 232
Connectors	1×DB9 male	Standard (6.3)
Signals	RS 232	RxD, TxD, RTS, CTS, DTR, DSR, DCD, RI, GND
Parallel Ports	1	SPP, PS/2, EPP
Connectors	1×DB25 female	
Signals	LPT	Standard for LPT (6.5)

Table 6: Features of VScom 110H PCI

### 2.2.6 VScom 210H PCI

The VScom 210H PCI card features two serial ports and one parallel port, available via a classic 5V PCI slot.

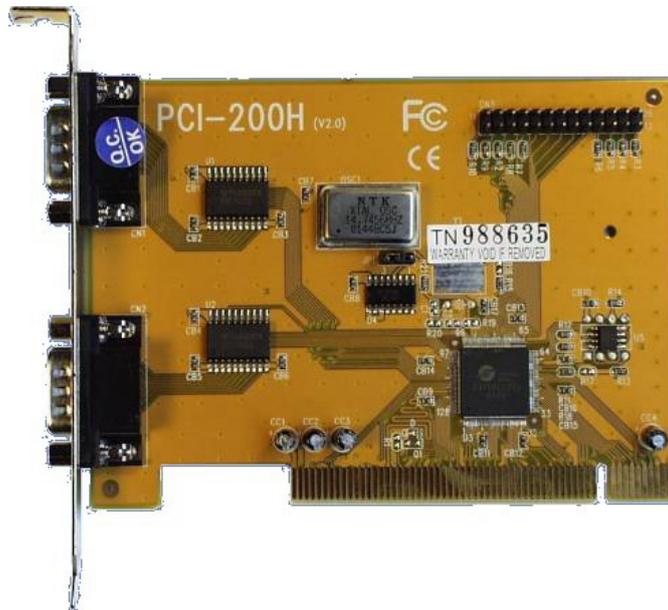


Figure 6: VScom 210H PCI

Bus	5V PCI	Classic slots only
Serial Ports	2	RS 232
Connectors	2×DB9 male	Standard (6.3)
Signals	RS 232	RxD, TxD, RTS, CTS, DTR, DSR, DCD, RI, GND
Parallel Ports	1	SPP, PS/2, EPP, ECP
Connectors	1×DB25 female	Extra bracket
Signals	LPT	Standard for LPT (6.5)

Table 7: Features of VScom 210H PCI

### 2.2.7 VScom 410H PCI

The VScom 410H PCI card features four serial ports and one parallel port, available via a classic 5V PCI slot. All four serial ports share one common connector DB37-female. For signal assignment see [6.2](#).

An adapter cable to four times DB9-male is provided, customers can order several other connection options.



Figure 7: VScom 410H PCI

Bus	5V PCI	Classic slots only
Serial Ports	4	RS 232
Connectors	1×DB37 female	Adapter to 4×DB9 male included ( <a href="#">6.2</a> )
Signals	RS 232	RxD, TxD, RTS, CTS, DTR, DSR, DCD, RI, GND
Parallel Ports	1	SPP, PS/2, EPP
Connectors	1×DB25 female	Extra bracket
Signals	LPT	Standard for LPT

Table 8: Features of VScom 410H PCI

### 2.2.8 VScom 400H RJ45 PCI

The VScom 400H RJ45 PCI card features four serial ports, available via a classic 5V PCI slot. To place all four serial ports to the bracket only, each port is provided by an RJ45 connector. Optional available are ready made adapter cables to DB9 male, but usually the cables are configured by the customer himself.

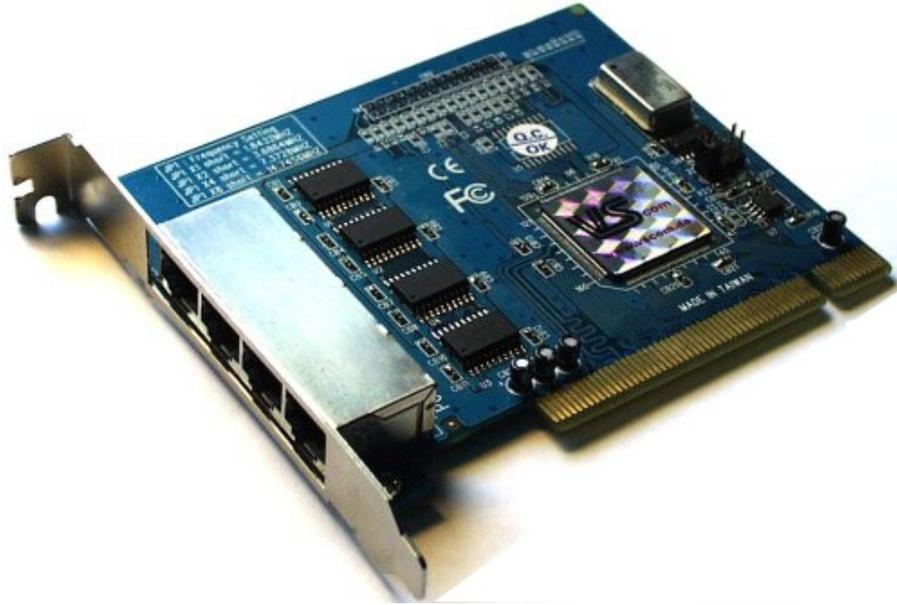


Figure 8: VScom 400H RJ45 PCI

Serial Ports	4	RS 232
Connectors	4×RJ45	4 on the card ( <a href="#">21</a> )
Signals	RS 232	RxD, TxD, RTS, CTS, DTR, DSR, DCD, GND

Table 9: Features of VScom 400H RJ45 PCI

### 2.2.9 VScm 021H PCI

The VScm 021H PCI card features two parallel ports, available via a classic 5V PCI slot.

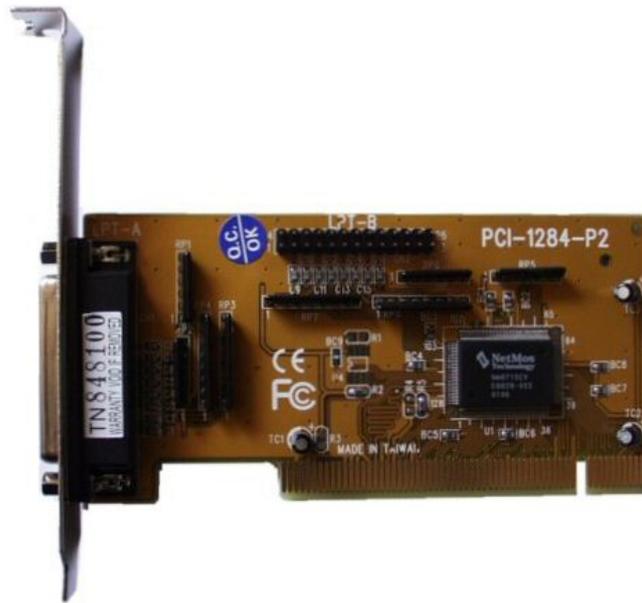


Figure 9: VScm 021H PCI

A model for Low Profile cases is available.

Bus	5V PCI	Classic slots only
Parallel Ports	2	SPP, PS/2, EPP, ECP
Connectors	1×DB25 female	1 on card, 1 on second bracket
Signals	LPT	Standard for LPT (6.5)

Table 10: Features of VScm 021H PCI

## 2.3 Packing List

All models are shipped with a driver CD-ROM, which includes the documentation and the driver software.

**VScm 400H UPCI** An extra bracket for third and fourth serial port is included.

**VScm 021H PCI** An extra bracket for the second parallel port is included.

**VScm 210H PCI** An extra bracket for the second serial port is included.

**VScm 410H PCI** An extra bracket for the parallel port is included.

An adapter cable from DB37 to four times DB9 is included.

**VScm 400H RJ45 PCI** Optional included are four adapter cables RJ45 to DB9.

## 2.4 About this Manual

This manual describes the hardware of VScOm H-Series Cards, as well as signal assignments. There is at most one configuration option for customers.

The part describing the driver is generic for all VScOm PCI cards, the same in all manuals. It will describe options not available on your VScOm H-Series Card. Just ignore such sections. Also Windows NT and even older versions of Windows operating system are mentioned.

The screen shots in this manual are made on an English language version of Windows XP. It is not difficult to find the appropriate part in any other language version of Windows.

The current version of the driver is 3.1.2.

## 3 Hardware Configuration

There are not much configuration options on the VScOm UPCI cards, since PCI is a Plug and Play system. System parameters like address or IRQ are not defined by hardware, they are assigned by software. Some models of VScOm UPCI cards allow to change the clock frequency of the serial ports. This is available as “compatibility speed mode”, configured by a separate jumper.

The H-series cards are based on a 16C950 type UART, which has many advantages over the more commonly used 16C550A and compatible devices. Given a certain crystal frequency the maximum bitrate is four times as high as on a 16C550A. Since the speed configuration is for software which is not aware of the added options in a 16C950, the maximum bitrate is given in 16C550A mode. This is 921,600 bps, when the top speed option is selected (which is by default).

Some models provide too many ports for placing them all on the bracket attached to the card. There are extra brackets with additional ports, connected via cable.

### 3.1 VScOm 200H UPCI

Right below the crystal is the compatibility speed header JP1. This is used to configure the maximum bitrate of the serial ports, please see table 11.

Pins	max. Bitrate	Remark
2-3	921,600	Default configuration
1-2	115,200	Compatibility option

Table 11: JP1 for two Speeds

The mentioned pins have to be connected by a cap to configure the operation mode. Make sure there is a cap placed in any of the two allowed positions, otherwise the ports will not operate. When the configuration has been changed, the driver for Windows has to learn this. In the Advanced panel click on the “Check” button (4.2.2).



### 3.2 VScom 400H UPCI

The third and fourth serial ports are connected via flat ribbon cable to the pin headers CN3 and CN4. Make sure the orientation of the connector is correct, the colored ribbon has to be aligned with pin 1.

The maximum bitrate of the VScom 400H UPCI is configured in exactly the same way as the 200H card, so please read in [3.1 above](#).

### 3.3 VScom 800H UPCI

Right below the crystal is the compatibility speed header JP1. This is used to configure the maximum bitrate of the serial ports. It is a block of eight pins, numbered top down and left to right. Please see table [12](#).

Pins	max. Bitrate	Remark
1–2	921,600	Default configuration, leftmost
3–4	460,800	
5–6	230,400	
7–8	115,200	Compatibility option, rightmost

Table 12: JP1 for four Speeds

The mentioned pins have to be connected by a cap to configure the operation mode. Make sure there is a cap placed in any of the four allowed positions, otherwise the ports will not operate. When the configuration has been changed, the driver for Windows has to learn this. In the Advanced panel click on the “Check” button ([4.2.2](#)).



### 3.4 VScom 210H PCI

The parallel port is connected via flat ribbon cable to the pin header. Make sure the orientation of the connector is correct, the colored ribbon has to be aligned with pin 1.

### 3.5 VScom 410H PCI

The parallel port is connected via flat ribbon cable to the pin header. Make sure the orientation of the connector is correct, the colored ribbon has to be aligned with pin 1.

### 3.6 VScom 400H RJ45 PCI

Right below the crystal is the compatibility speed header JP1. This is used to configure the maximum bitrate of the serial ports. It is identical with the configuration on VScom 800H UPC, so please check out section [3.3](#).

### **3.7 VScom 021H PCI**

The second parallel port is connected via flat ribbon cable to the pin header. Make sure the orientation of the connector is correct, the colored ribbon has to be aligned with pin 1.

## 4 Windows Driver

This chapter of the manual documents the drivers for Windows Operating Systems. The driver is basically the same for all the VScom PCI cards available, independent of the model. As one consequence there is only one driver for [download](#), linked to from all the products. It does not matter which link you use.

When Windows loads the driver, the type of card is identified. The driver adapts to the options available on the certain card, and disables or hides other options. Because of the great similarity of the products, users find the options always at the same positions, they do not have to learn each product individually.

The following text is common to all cards, important differences are mentioned when necessary. When a described option is not available on your product, this is not an error. This option is just not offered by your hardware.

The manual first describes the process of installing the driver software, followed by instructions for removing the drivers in the unlikely case you wish to do this. The drivers still support very old phased out products, as well as old Windows versions 95 to ME (9x strand) and NT. The old Windows systems are no longer supported by Microsoft, so there may be hidden problems on your computer. These may cause the VScom cards to malfunction in your system. Said that, we can not guarantee the old products operate reliably, and also modern products may have unexpected behaviour in old operating systems. The old operating systems are just mentioned in the following.

Many VScom cards may be installed simultaneously. In fact the driver does not limit the number of cards installed, Windows does. In Windows 9x the maximum number of serial or parallel ports installed is 127, in Windows NT or later the limit is 256 ports of serial or parallel kind.

The current version of the driver is 3.1.2.

### 4.1 Installing the Drivers

Windows Systems perform the hardware detection by Plug & Play, you do not have to configure the card in any way. Special DIP switches as described above of course are an exception.

Windows NT performs different, this is described later (section [4.3](#)).

#### 4.1.1 Install in Windows 9x<sup>1</sup>, 2000, XP and Vista

1. Install the card in your system and boot Windows.
2. Log on to an account with administrative privileges.  
In Windows 95/98/98SE/ME each account is appropriate,  
Windows NT is handled extra below.
3. Windows will automatically detect the VScom H-Series Card.
4. The “Add New Hardware Wizard” dialog box appears and searches for new drivers for the VScom H-Series Card.

---

<sup>1</sup>Windows 95 up to Windows ME

5. Click on “Next”.
6. From the listed box, choose “Search for the best driver for your device”.
7. Click on “Next”.
8. If not done already, insert the driver CD into the CD-ROM drive.
9. From the generated choices, choose “CD-ROM drive”.
10. Windows searches for the driver, and detects the .inf-file on the CD. It is ready to install the driver for VScOm H-Series Card.  
Click on “Next”.
11. On some systems Windows may claim about a “Missing digital signature”. This is not an error, just “Continue”.
12. Click on Finish.
13. Windows now detects the COM ports on the VScOm H-Series Card.
14. If Windows requests the driver disk again, point to the directory named after your operating system, e.g. “Windows XP (32 bit)”.
15. Windows will copy and install the driver for the new PCI serial ports.
16. You might be asked to restart your computer in the process. We suggest to wait with until all components (cards and serial ports) are completely installed, and restart then.

### 4.1.2 Install in Windows 7

Windows 7 looks different when installing drivers. It seems it is very different from the other Plug & Play enabled versions. But this is a deception.

When detecting new hardware Windows first visits the website of Windows Update to find a suitable driver, which shall be installed. If none is found, Windows searches the local system for the required driver, and loads it. If also there is none, this would be the time to pop-up the “Add New Hardware Wizard”, so the user can supply a driver. And this step does not happen in Windows 7, the Wizard does not appear<sup>2</sup>.

Instead Microsoft decided to have the user run an installation program for the hardware. This software will install all the driver files to the system, and then initiates a hardware detection process. This installation program in version 3.3.0.15 is available for all cards.

The following assumes the card is already installed in a slot.

1. Start the installation program. On the CD-ROM this is named vscompci.exe, and vscomp-cix64.exe for the x64 Edition of Windows 7. For download from [download](#) it is available as a self-extracting file, which detects the system automatically. This is the program, click on “Install”.

---

<sup>2</sup>It is still available using some tricks



Figure 10: Windows 7 Installation

2. The software requests a location to install the required files.



Figure 11: Driver Installation Target

Confirm the location.

3. The installation proceeds.



Figure 12: Windows 7 Installing Drivers

When this is completed, the hardware is recognized by Windows 7, and the drivers are loaded. This may take some time, especially for many serial ports.

It is also possible to first run the Installation program. This will install the drivers in the same way, but they are not loaded. The system is prepared to detect a card. After shut down and installation of the hardware the serial ports are installed without further user intervention. The same happens when more cards are added to an existing system.

#### 4.1.3 Verify the Installation

You can now verify the installation by looking at the “Multi-port serial adapters” section of the Device Manager (Go there by Start - Settings - Control Panel - System - Hardware - Device Manager). There you will find the new device “VScom H-Series Card” listed.

Select this, and open the properties. In the “Advanced” tab, click the “Check” button to verify the speed setting from the hardware installation. You may also rename the ports here.

Check

**About assigned port numbers** In some systems the ports will receive numbers customers do not expect. For example a card with two serial ports may receive the names ‘Com5’ and ‘Com8’, but

the expected ports 'Com3' and 'Com4' do not appear in the Device Manager. This is neither an error of the card or the driver, it is just how Windows manages port numbers.

Windows maintains an internal data base<sup>3</sup> of port numbers which have been assigned to some hardware in the past. These numbers are skipped when assigning port names to new hardware. The numbers may be changed after installation of VScom hardware.

### 4.1.4 Removing the Software in Windows 9x<sup>4</sup>, 2000, XP and Vista

To remove installed files and Windows registry information

1. Go to the directory where you installed the driver from as described [above](#).
2. Double click the "VSCLEAN.EXE", and reboot your computer when asked to.

### 4.1.5 Removing the Software in Windows 7

To remove installed files and Windows registry information you may use the same way as in the past (above [4.1.4](#)). Or you may use the installation software.

1. Start the installation program, and use the "Remove" option.

---

<sup>3</sup>A Binary value in the registry: HKLM\SYSTEM\CurrentControlSet\Control\COM Name Arbiter

<sup>4</sup>Windows 95 up to Windows ME



Figure 13: Windows 7 Driver Removal

2. Short time later the removal is done.



Figure 14: Windows 7 Drivers removed

## 4.2 Card and Port Configuration

Each card is organized in two types of components in the system. One type is for the card itself, i.e. the basic PCI interface. The other type is for the ports. This is reflected in Windows and the driver software.

### 4.2.1 Card and Ports in Device Manager

The category where the VScom cards are installed is 'Multi-port serial adapters'. And as expected there is the category of 'Ports (COM & LPT)' for the serial ports. Both categories are standard on Windows.

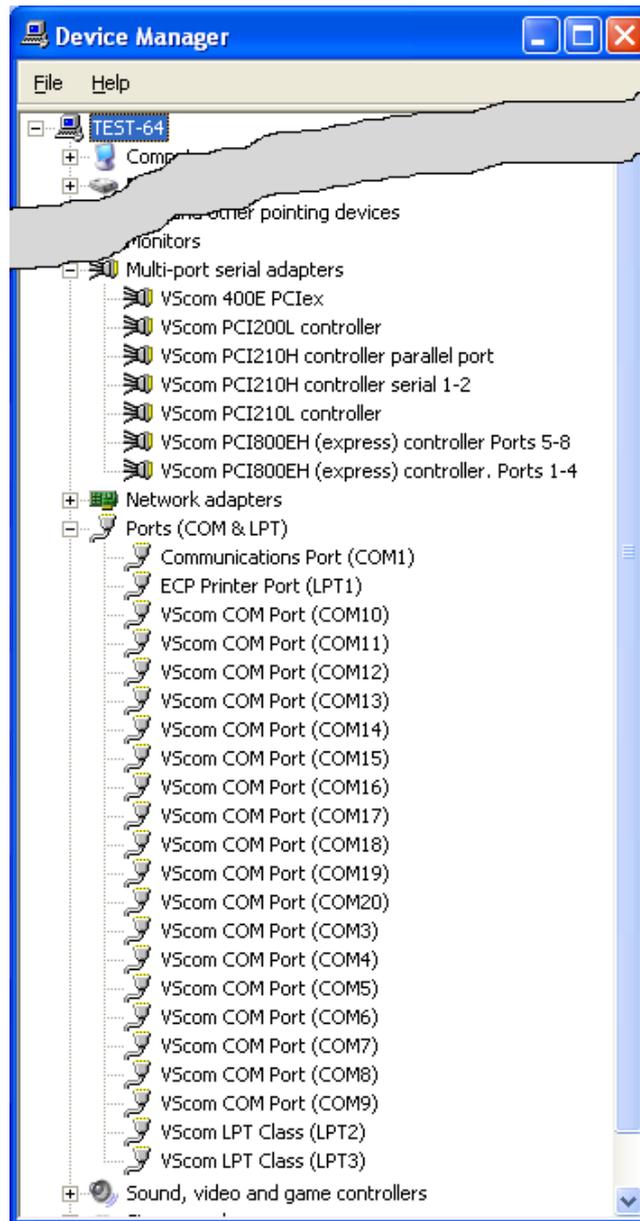


Figure 15: Cards and Ports in Device Manager

Shown here are the data of some serial and parallel VScOm cards in the Device Manager. There are many cards installed simultaneously, just to show this is possible.

The 800EH is displayed as two items in 'Multi-port serial adapters'. Because the H-series cards support two functions each card is listed as two devices. The 800EH card appears as two groups of serial ports. The same happens for the 210H card, with two serial ports in one group, and the parallel port in the other.

L-series cards like 200L or 210L put all ports in one component, so serial and parallel ports may be mixed.

### 4.2.2 Card Advanced Settings

Open the properties of a card via the Device Manager, and select the “Advanced” tab. These are the advanced settings for a VScom card, a VScom 210L UPCI in this case.

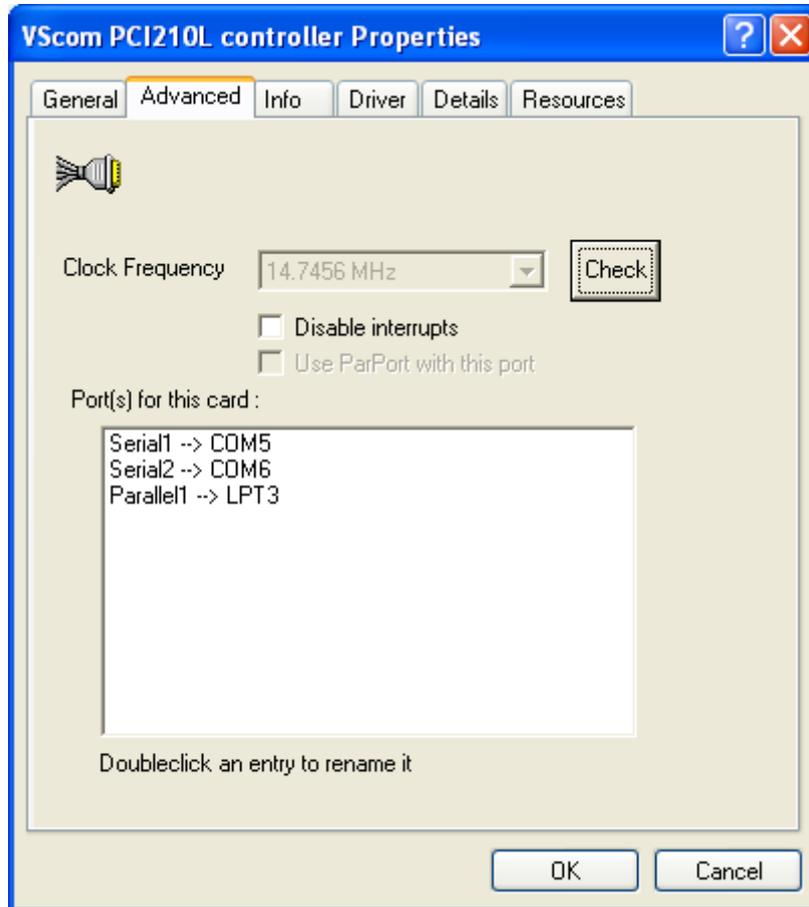


Figure 16: Card Advanced Properties

This panel displays the ports of the card, and their assigned names in Windows.

For serial (COM) ports there is the “Clock Frequency”. Some models allow to change this clock, which is the same for all ports on a card. When installing a card the driver attempts to detect the clock of the serial ports. If the system is very busy while installing the driver, the detection may yield a wrong result.

If you have communication problems after installation, click the “Check” button. The driver will measure the actual speed again, and change the entry as needed. There is no option to manually enter the clock value. Unless you have very special requests you should leave the hardware at top speed. This offers higher speed and a wider range of bitrates, especially unusual configurations.

The display of parallel (LPT) ports is similar.

Check

The COM- or LPT-assignments of each port may be changed from the initial values after installing the drivers. To change the assignment just double-click the entry in the “Advanced” panel.

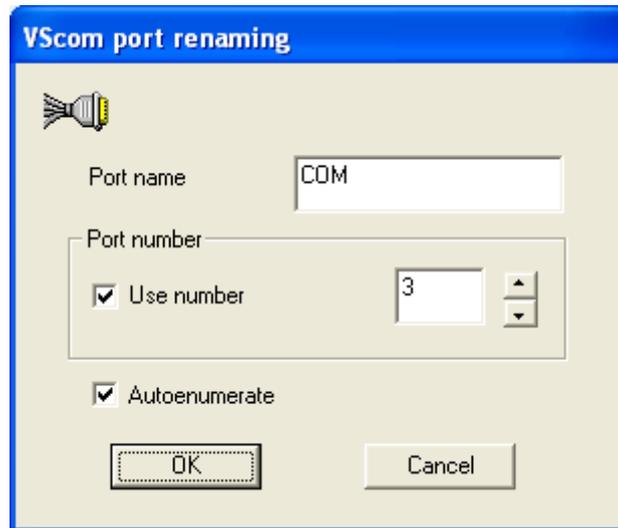


Figure 17: Port Renaming

Then this panel is opened for renaming a port.

In some cases it may be useful to change the name from COM (or LPT) to something else. Do this only if you exactly know what you are doing. Usually you will change the number of the first port only. With the ‘Autoenumerate’ feature enabled the subsequent ports receive the next numbers. However it is possible to assign port numbers with gaps in between.

### 4.2.3 Ports Configuration

The available functions of the serial ports depend on the type of UART implemented on the card. Cards with 16C550A type serial ports provide less configuration than the 16C950 UART. As a consequence some panels are just missing. This is not a sign of an error.

### VScom Standard Port Settings

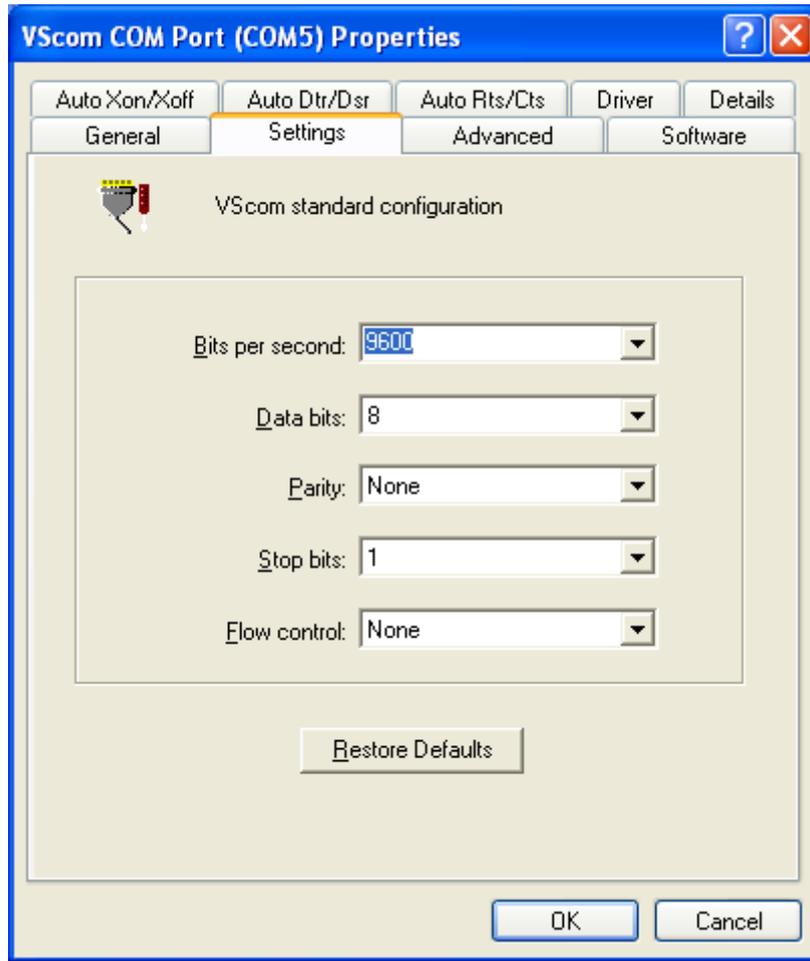


Figure 18: Standard Settings

This panel is well known to most users, it is similar to the panel used by the Microsoft driver for Com1/2.

**VScom Advanced Port Settings** The Microsoft driver for Com1/2 provides some control of FIFO configuration. The Advanced Settings are for similar purposes, but with extended functions.

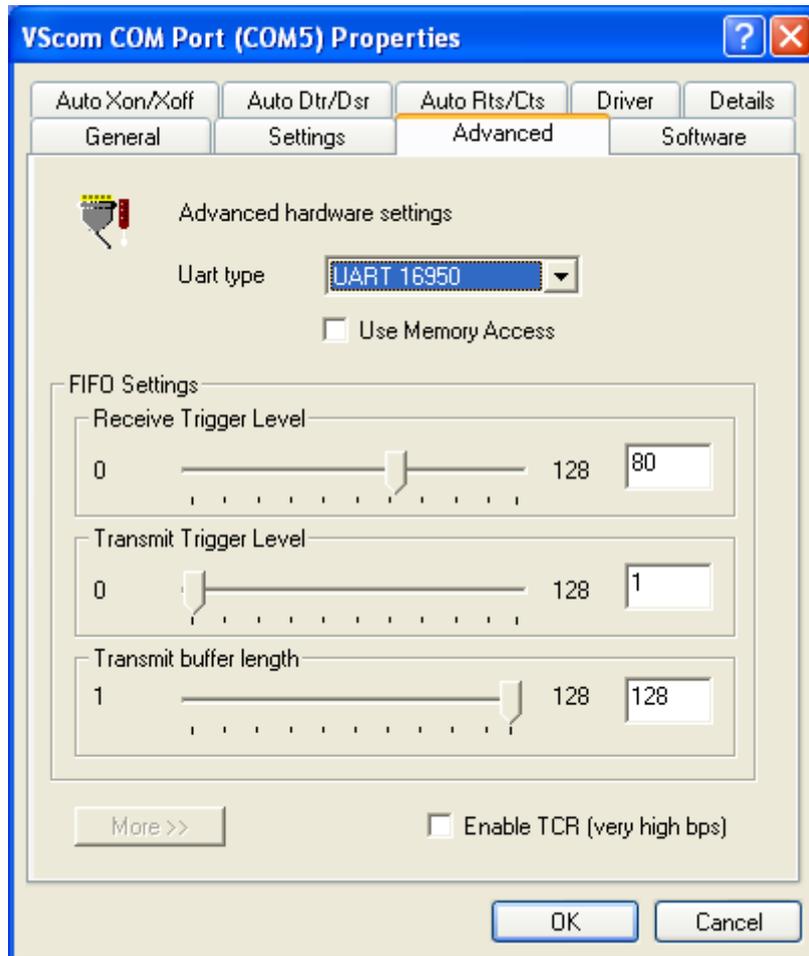


Figure 19: Advanced Settings 16C950

The VScom H- and E-series cards use a 16C950 UART to transmit data. This chip and the driver can emulate a variety of other UARTS. We suggest to use the standard value of 16950, the driver is optimized for this. If the driver is used for other VScom series, a different UART may be implemented. The available emulations depend on this. At least a 16550 is possible with all series. The possible settings for FIFO control depend on the chosen emulation.

Here the elements one by one

**UART type** This is to choose the emulation. The UARTS of type 16450 and 8250 operate without FIFO. We suggest to use the highest possible setting for best operation.

**Use Memory Access** Instead of operating the UART via I/O, it is possible to operate it via Memory Access. In critical systems this can save a little amount of performance. It may be worth a try, however VScom recommends to stay with the box unchecked.

**Receive Trigger Level** The time when the UART generates an interrupt for received data. It defines the amount of data necessary to do this. With the 16950 you may select any value from 1 to 128. Other types allow only certain values. Higher values increase performance, lower values decrease possibility for loss of data. As a reference: the usual value for a 16C550A is 8, leaving space for 8 more bytes to receive.

**Transmit Trigger Level** The time when the UART generates an interrupt for sent data. When the transmit FIFO contains less data than this, the interrupt is issued. This value is not available with all emulations. Higher values make a smoother transmission of data, but increase the system load.

**Transmit Buffer Length** The amount of data sent to the UART when the transmit FIFO is empty. This is a complete software item. The value ranges from 1 to the FIFO size of the emulation. Higher values increase transmission performance. The typical value here is 16, the FIFO size of a 16550. This value must not be less than the Transmit trigger level. CAUTION: external hardware - such as a modem - must be able to handle this blocksize, very old systems may require to reduce this parameter to '1'.

**Enable TCR** The 16C950 UART used on the H-series cards has some hardware option to get very high bit rates, up to 3.6 Mbps. Of course these bit rates can not be used on RS 232, but there are other advantages. The maximum serial speed is raised, so there are more possible bit rates available in the usual range of RS 232 transmission. This allows to use some more rather unusual settings for transmission rates, some special hardware may require. Possible products with RS 422/485 line drivers can use such high bit rates. Since this option may reduce the transmission accuracy in certain environments, it is disabled by default.

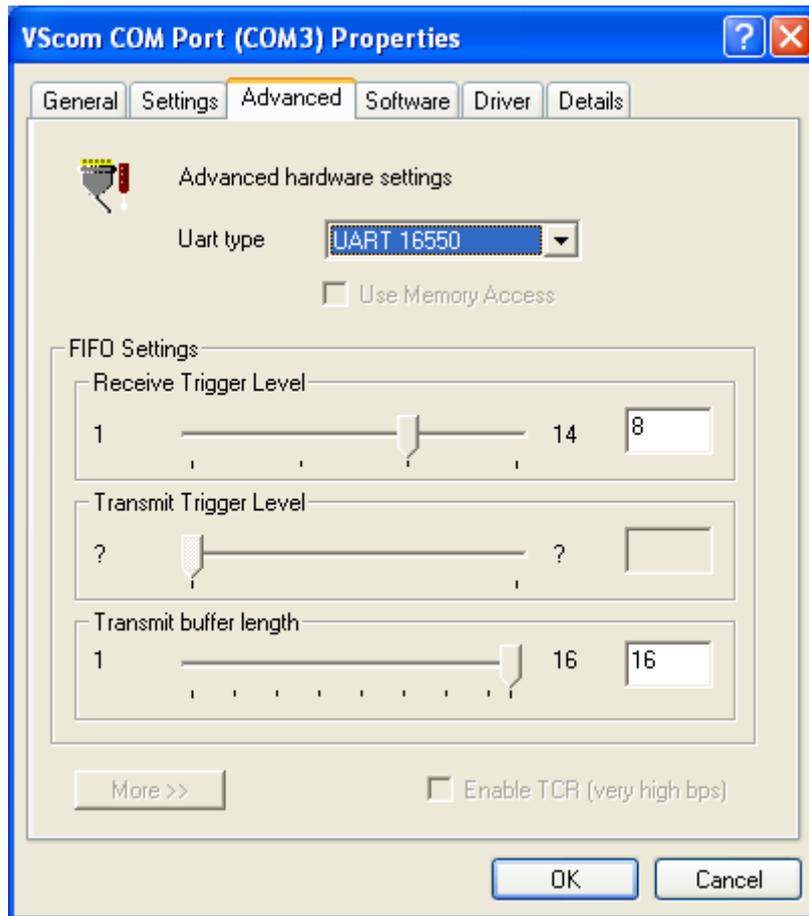


Figure 20: Advanced Settings 16C550A

The VScom L-series cards have a 16C550A compatible UART for the serial ports. The Receive Trigger Level only provides the four steps 1, 4, 8 and 14. There is no Transmit Trigger Level, hence this slider is disabled. The Transmit Buffer Length ranges from 1 to 16 only. The serial port can emulate the 16C450 and the ancient 8250 UART.

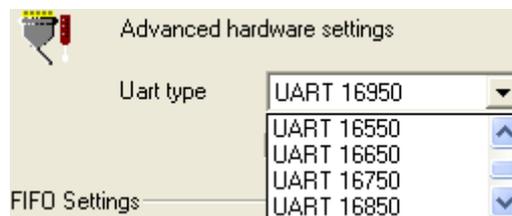


Figure 21: UART Types

**Remarks on the UART Emulation Modes** Usually the best choice for the UART type emulation is just the best available, i.e. the standard 16C950 model. However some hardware options may be configured to match the functions of less advanced types. This may be required for rare and very

special situations, though in some of those it is even better to just change the configuration of the other parameters.

**8250** The FIFO is totally switched off, as are all other options. The real 8250 UART had a speed limitation of 57600 bps, this is maintained by software here.

**16450** The most common UART in the late 1980 decade. There still is no FIFO available, the maximum bitrate is raised to 115200 bps. Sometimes this Emulation may be useful, if the connected device is overrun by the transmitted data. But usually it is better to restrict the Transmit Buffer length to 1.

**16550** The most common UART until today. There are 16 Byte FIFO for transmit and receive each, and four trigger levels for receive. The built-in serial ports of modern computers use this UART model, with a maximum speed of 115200 bps, which is 921600 bps on VScom cards.

**16650** This UART basically has 32 bytes FIFO for transmit and receive each, and also four trigger levels for receive. However this emulation uses 128 Bytes FIFO for receive. Additional function is automatic Flow Control by RTS/CTS or XON/XOFF.

**16750** This is another UART with extended FIFO compared to the 16550 model. In basic configuration there are 16/16 Byte FIFO, which can be increased to 64/64 Byte. However as an emulation it uses 128/128 byte FIFO. Also automatic Flow Control by RTS/CTS and Xon/Xoff is possible.

**16850** This is another UART model with 128/128 Byte FIFO.

**16950** This is the native mode the ports on the VScom cards are used. In most situations it is better to use this mode, and restrict the options to a suitable configuration.

## Enhanced Software Settings

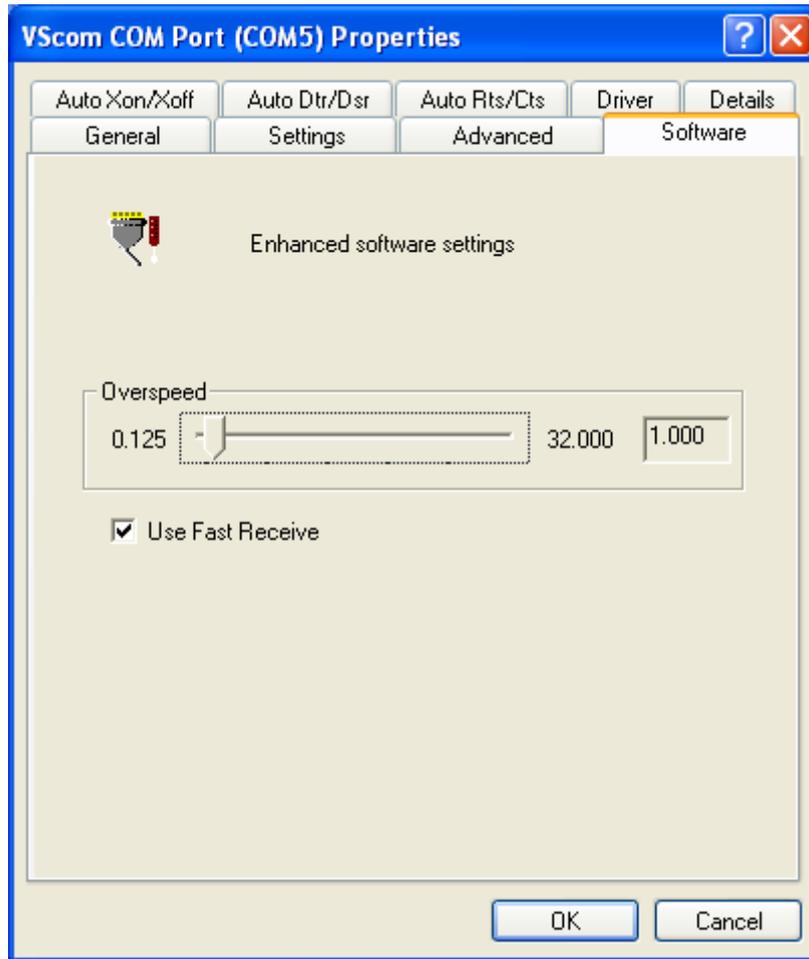


Figure 22: Software Settings

This is a special feature of the VScm driver. In older driver versions it was simply called Overspeed.

Overspeed opens the possibility to get other speeds than the application requested for. The application requests a certain speed; this is multiplied with the value defined in here. The result is set as the real speed. Possible factors range from 0.125 (1 divided by 8) up to 32 in steps of 0.125. Of course you can not get a speed which is higher than the maximum speed of the port (921600 bps / 3.6 Mbps). The normal value is 1 of course, and it should be left there. This is a feature with some risk. With a modem you very easy get a speed the modem can not handle.

Examples:

- Wrong: Overspeed is set to 8. The modem has a maximum interface rate of 460800 Baud. When the application requests 115200, the result is 921600 Baud. The modem can not handle this.

- Wrong: Overspeed is set to 2. The application requests 38400, the result is 76800. Even when this is below the maximum speed of the modem, it does not recognize this speed in many cases. Operation is not possible.
- Wrong: Windows searches for Modems at certain speeds. Modem detection may fail, because of wrong settings.
- Correct: The application is limited to 19200. Overspeed may be 6 or 12 to get 115200 or even 230400 with a modern modem.

### Automatic Software Flow Control

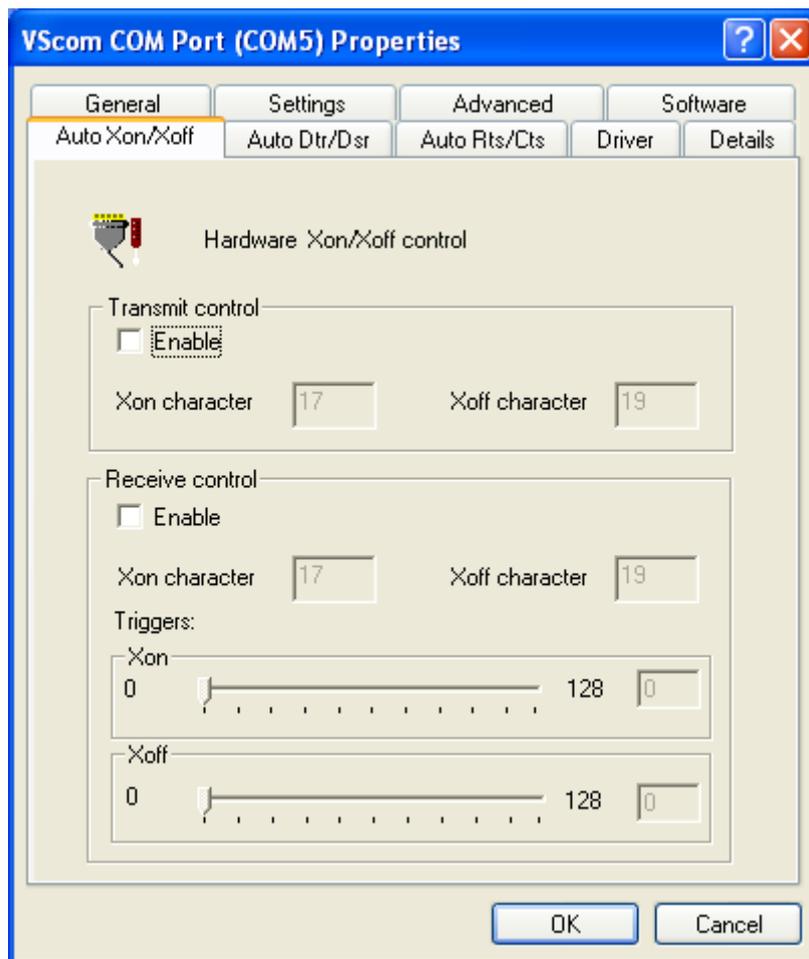


Figure 23: Auto XON/XOFF

This panel configures the On-Chip Software Flow Control provided by the UART. Only some UART types support the Flow Control directly on the chip. Use the 16950 emulation to allow rich possibilities.

**Transmit control Enable** This enables the automatic control when sending data. If the receiving device sends a stop (Xoff) signal, transmission is halted. When a continue (Xon) signal is received, transmission continues. This reduces the probability of sending too much data to the receiver. This does not need any processing by the main processor.

**Transmit control Xon/Xoff Character** The special characters acting as continue and stop signals as above. Normally stop (Xoff) is a ^S (dec: 19, hex: \$13), continue (Xon) is a ^Q (dec: 17, hex: \$11). You should not change these values.

**Receive control Enable** This enables the automatic control when receiving data. When too much data is received and not read by the processor, the UART generates a stop signal to the other side. When enough data is read by the processor, the UART generates a continue signal.

**Receive control Xon/Xoff Character** Same function as Transmit control above. Even when it is possible to use other values than above, we strongly recommend to use the standard values.

**Receive control Triggers Xon** The level when the UART generates the continue signal. When the processor reads the received data, the amount remaining in the UART will eventually fall below this limit. At that time the continue signal is issued to the other side.

**Receive control Triggers Xoff** The level when the UART generates the stop signal. When this amount of data is received and stored in the FIFO, the UART issues the stop signal to hold further transmission by the other side. This value must be higher than the Xon Trigger. Depending on the global situation (processor speed, system load, transmission parameters, transmitting device, ...) possible values can be: 32 (quarter of FIFO) for Xon and 64 (half) for Xoff.

### Automatic Hardware Flow Control by RTS/CTS

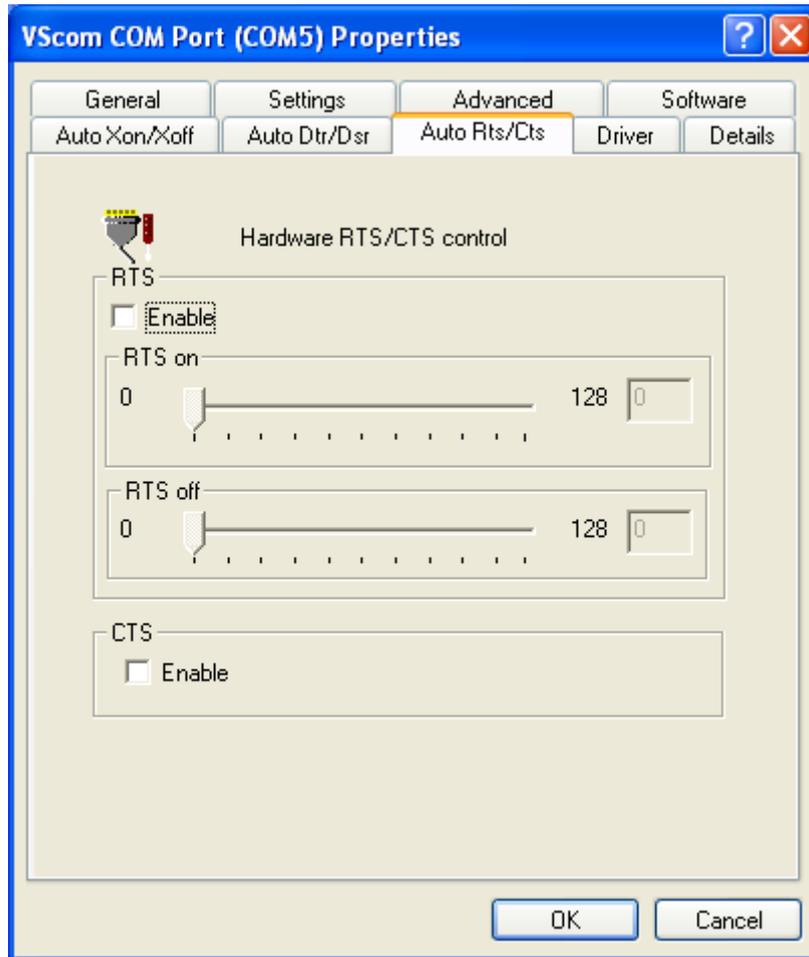


Figure 24: Auto RTS/CTS

This panel configures the On-Chip RTS/CTS Hardware Flow Control provided by the UART. The function of this panel is similar to the software Flow control panel. However, the stop and start signals are provided by the RTS and CTS signals. RTS controls the input (receive) direction, while CTS controls output.

Basics of hardware flow control: When CTS is asserted by the other side, transmission is allowed. If deasserted, transmission is halted. When the system is ready to receive data, it asserts the RTS line to the other side. If it needs a pause, it deasserts RTS.

**RTS Enable** Selects the On-Chip receive Flow control.

**RTS on** Function is similar to the Xon slider from [above](#).

**RTS off** Function is similar to the Xoff slider from [above](#).

**CTS Enable** Selects the On-Chip transmit Flow control, similar to the [Transmit Control above](#).

## Automatic Hardware Flow Control by DTR/DSR

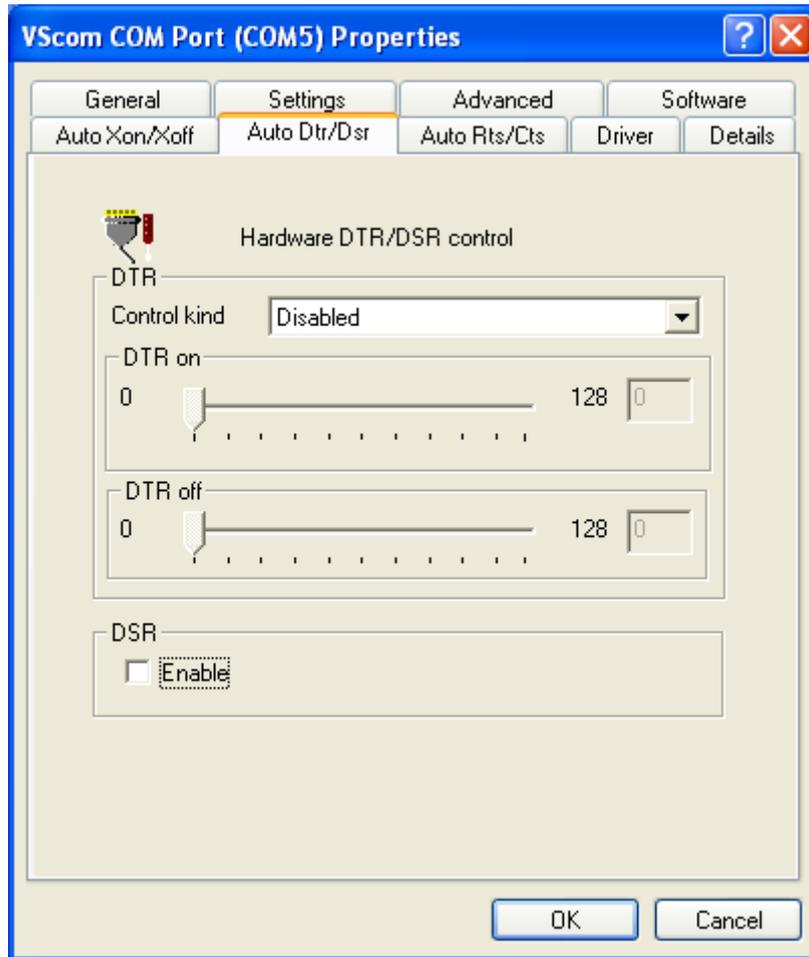


Figure 25: Auto DTR/DSR

This panel selects the function of DTR/DSR Hardware Flow control. This function of the UART offers more options.

**Control kind** How DTR is intended to operate.

**Disabled:** No Flow control for receive.

**Triggered:** Flow control similar to RTS Enable.

**RS485 Low Active:** This can be used to control a converter RS232⇒RS485, see [below](#). DTR is low while data is transmitted.

**RS485 High Active:** This can be used to control a converter RS232⇒RS485, see [below](#). DTR is high while data is transmitted.

**DTR on** Compare the slider of RTS on.

**DTR off** Compare the slider of RTS off.

**DSR enable** Enables transmission control via the DSR input.

**RS 485 operation** RS 485 very often uses a shared medium (one pair of wires) to send and receive data. This requires the ability to disable the transmission unit. It must be off to receive data, and must be switched on when transmitting. After transmit is finished it must be off again as soon as possible.<sup>5</sup> Very often this switching is controlled via signals like RTS or DTR. But in many environments it is difficult for an application to do this in a reasonable time. Now this option can help.



Figure 26: Auto-DTR Modes

The UART controls the DTR by itself. As long as data is in the transmission FIFO the DTR signal is held High (or Low). When the transmission FIFO is empty the DTR changes. An external converter may monitor this signal to act as needed.

RS485 usage can be compared to communication over handheld transceiver, many people are familiar with this. To talk one has to press a button. If two or more persons talk at the same time, no one can hear clearly. For listening the button has to be released. Now the DTR operates like an “automatic” button, which presses itself as soon as one starts to talk. And it releases afterwards while the user is silent.

This feature is very useful for older applications. Often there are applications written for a half-duplex protocol<sup>6</sup> over RS 232. Now this must operate over a longer distance. So RS 485 is chosen to transmit the data. But the application does not know about this, and does not control the RTS or DTR signal. The driver can do this invisible to the application.

### 4.3 Windows NT 4.0

Windows NT 4.0 is different from Windows 2000 up to the latest versions of Vista, and also the no longer supported<sup>7</sup> strand of 9x systems (Windows 95 up to ME). These other are all Plug & Play systems, meaning they will detect<sup>8</sup> added hardware themselves. Users are asked to provide the required driver files. In Windows NT drivers are installed, and fail to load if the hardware is not available.

The VSCOM drivers for Windows NT implement the Plug & Play functions in a limited sense. There is a supervisor utility, restricted to detect VSCOM PCI cards.

<sup>5</sup>Think of it as talking by Walkie-Talkie

<sup>6</sup>XMODEM and YMODEM are good examples

<sup>7</sup>By Microsoft that is, they ceased support for all versions up to Windows 2000

<sup>8</sup>You may even configure the hardware via Windows

### 4.3.1 Installing in Windows NT 4.0

You need to have administrator privileges to install any new drivers under Windows NT 4.0. To install the driver or update the configuration please log on to Windows NT 4.0 as “Administrator” or ask your system administrator to install the VScom H-Series Card and driver. Please proceed with the following steps to install the driver:

1. Switch on the computer and start Windows NT 4.0.
2. Insert the supplied CD-ROM into the drive.
3. If your system is configured for Auto-start of CD-ROM, go to the products page, find your product and select driver. You’ll find a link to the NT4 drivers directory.
4. Otherwise double click on “My Computer”, and choose your CD-ROM drive.
5. Double Click “Windows NT 4.0”.
6. Double Click “PCI\_Driver”.
7. Double click “vsinst.EXE”.
8. The Installer starts and requests confirmation of installation. You may change the target directory for the software. If there is a space in the name, please use the short name (e.g. PROGRA~1 for “Program Files”) instead.
9. Click Yes, and Reboot when requested.



**Automatic Detection of PCI Cards** The system detects a new VScom Card when an Administrator logs on to Windows NT 4.0



Figure 27: NT Plug & Play

The default option is to immediately install and configure the new card. However the upper check-box allows to delay the configuration to the next logon. The lower check-box disables the check for new hardware until explicitly enabled again.

**Configuration via Control Panel** The installation program also adds the “VScm PCI” applet to the Control Panel of Windows NT. Open it, to get access again to the configuration panel (figure 27).



**System Configuration** This panel opens when a new hardware is detected, or when the configuration is explicitly opened via the control panel.

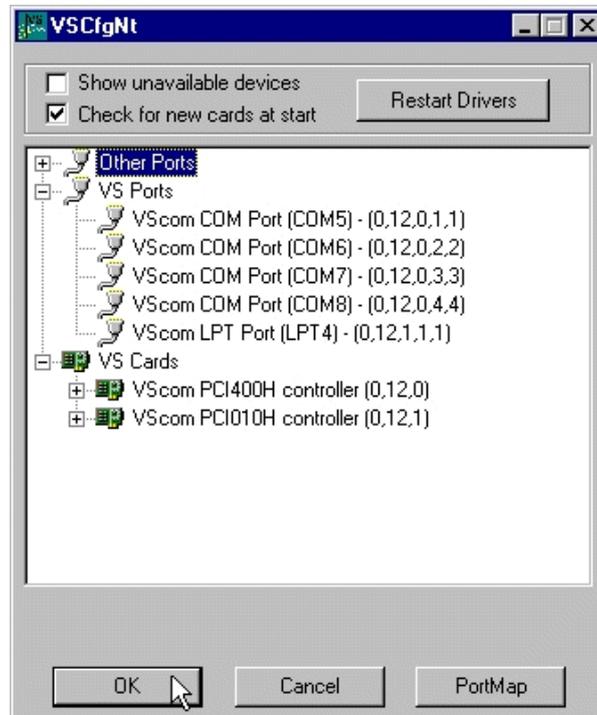


Figure 28: NT Device list

This is the basic configuration panel for the VScm PCI driver in Windows NT 4.0; the lower check-box re-enables the automatic check for new hardware. Here is the place to revoke the change done in the detection panel.

The upper check-box enables to uninstall the drivers for already removed cards, without a complete uninstall of the drivers. Different from other Windows, in NT all installed drivers load and check for there hardware. This is not efficient when the hardware is removed some time ago, and causes error messages in the event viewer. The properties of each installed VScm PCI card and its ports are associated with the PCI slot where the card is placed. When a card is removed or displaced, it is necessary to display the settings via this check-box. Then the board may be selected and removed.

**Card and Port Configuration** Open the panel as shown in figure 28. Select a VScm controller under “VScm Cards” to gain access to the card properties. Or select an individual port under “VScm Ports” to change its settings. The options are explained in the general sections of this documentation.

**Uninstall Drivers from Windows NT** To remove installed files and Windows registry information

1. Go to the Control Panel.
2. Open the “Add/Remove programs” applet.
3. Select “Vscm drivers for ...”.
4. Click “Remove”.
5. “VScm install” requests to confirm the uninstall, Click “Yes”.
6. Reboot the computer when requested.

However Windows NT 4.0 does not perform such an automated search. Users have to run the ‘VSINST.EXE’ program from the driver set, to install the cards. This installs the drivers to the system, and also a service for automatic detection of VScm PCI cards. Restricted to these cards, the usage is as handy as with modern Windows versions.

**Checking Installation in Windows NT** You can now verify the installation by looking at the “Administrative Tools” section of the “Windows NT Diagnostics (also called WinMSD)” (Go there by Start - Setting - Windows NT Diagnostics - Administrative Tools - Resources). There you will find new device “vscnt” listed.

Double click on it, you will see the IRQ and I/O information.

Click on the right tab, you can see all the detailed information of IRQ, Input/Output & Memory Range used by PCI Ports.

Open the “VScm PCI” applet (see 4.3.1) in the control panel (Go there by Start - Setting - Control panel). Open “VS Cards” and double click on the detected controller to open the properties. In the “advanced” tab, click the “Check” button to verify the speed setting from the hardware installation.



You may also rename the serial ports here.

## 5 Linux Driver

VScom PCI cards in Linux do not need a special driver. The driver 'serial' supplied with your Distribution is quite capable of controlling the serial ports on VScom cards. Driver 'serial' is configured to detect the old-fashioned ports known as Com1 to Com4 in DOS systems, and support them. Since the serial ports on VScom PCI cards use the same type of hardware, driver 'serial' is well capable to support the new serial ports. After installation of the card the driver 'serial' just needs to know about the new hardware.

Since Kernel 2.6 things are no longer as simple as written above. Some programmers invented a limit to the maximum number of serial ports. This is explained below in [5.3](#).

### 5.1 Manual Installation

Usually the command 'setserial' is used to provide the required information to the driver, thus adding the port.

```
setserial ttys4 port 0x100 irq 5 uart 16550A baud_base 921600
```

Such a command would be good for an ISA card, but with PCI it is not totally as easy as that. Because PCI is a Plug&Play system, the parameters 'port' and 'irq' are unknown in advance; they are assigned at boot time. Depending on your installation of Linux, you find the basic parameters in /proc/pci or via the lspci command. Given some hardware details about the card, port and irq for each serial port are calculated easy. These details are seen in [table 14](#).

This is just mentioned here, the better way of installation is automatic.

### 5.2 Automatic Installation

Depending on your version of the driver 'serial', some VScom PCI cards are already known. So the ports are detected, installed and usable. If the card in question is not already known to the driver (new model or old driver/kernel), the program 'vscardcfg' is supplied. This program scans /proc/pci or other information for parameters of VScom PCI cards.

#### 5.2.1 Known Cards

PCI components are detected by BIOS and system routines at boot time. Later they are basically identified by the parameters Vendor- and Device-ID, on some hardware also by SubVendor- and SubDevice-ID. For identification of VScom PCI hardware, the Sub-parameters are not required. Not each specific hardware model has its unique identification, but at least for purposes of driver configuration different identification is supplied. So driver and configuration software just has to look up the identification in a table, and then controls the card perfectly.

If the VScom PCI card is known to the driver 'serial', users do not have to do any extra configuration. The system detects the card, identifies it and installs additional ports.

### 5.2.2 New Models

If the model of VScom PCI card is new to the driver, users add the program `vscardcfg` to their init scripts. This program detects the cards and the assigned parameters. Since this program knows about all VScom PCI cards, it also knows the required parameters for the `setserial` command mentioned in section 5.1. Then it performs the same tasks as users would do when manually invoking `setserial`. Especially the parameters `'port'`, `'irq'`, `'uart'` and `'baud_base'` are given according to the model and configuration.

## 5.3 Problems since Kernel 2.6

In Linux systems based on Kernel 2.4 or earlier the installation of VScom PCI cards has been quite easy. This has changed since kernel 2.6 came out. Not because the driver `'serial'` was removed or could no longer support PCI serial ports. This happened because some programmers invented a limit to the maximum number of serial ports installed by driver `'serial'`. And they selected a default value ridiculously low (typically 4), which caused problems again and again. So over time they have been forced to offer several methods of bypassing this limit.

### 5.3.1 Compile a new Kernel

In early versions the driver `'serial'` required a configuration with a new limit. Then it had to be compiled with this limit. Since usually the driver is not loaded as a module, this required to compile the kernel.

Following kernel compile settings have to be set		
<code>CONFIG_SERIAL_8250_NR_UARTS=32</code>	Maximum number of 8250/16550 UARTs	32
<code>CONFIG_SERIAL_8250_RUNTIME_UARTS=32</code>	Number of 8250/16550 serial ports to register at runtime	32
<code>CONFIG_SERIAL_8250_EXTENDED=y</code>	Extended serial port options	mark
<code>CONFIG_SERIAL_8250_MANY_PORTS=y</code>	Support more than 4 legacy serial ports	mark
<code>CONFIG_SERIAL_8250_SHARE_IRQ=y</code>	Support for sharing serial interrupts	mark

Table 13: Kernel Compile Options

Depending on your distribution there may be different methods to define these parameters. Menu-config should be a good option.

### 5.3.2 Kernel Start Parameters

Today the implementation of the limit is in a way which allows to raise the defined value at boot time. This is done via Kernel parameters. How this is done depends on the way you boot your system. Following is one example to do the task in Ubuntu 8.10 on AMD64 with Grub as boot loader, basically Debian should operate the same way. Other distributions or loaders may require modifications.

1. In “/boot/grub/menu.lst” is a (Comment-) line with “kopt=”-Entry and text “8250.nr\_uarts”. This line shall remain there as a comment, the new value should be “8250.nr\_uarts=32”. Or any number appropriate for the new configuration.
2. Run update-grub. After that check if the above parameter is available as Kernel Entry in the file “menu.lst”. The entries from the kopt-line shall appear as extension after running update-grub.
3. Copy “vscardcfg64” for 64 bit systems to “/usr/local/bin”.
4. Program “/usr/local/bin/vscardcfg64” is added to “/etc/rc.local”, to run it at boot time. You may add an “echo” to show an explaining text when the system boots.

On 32 bit systems the program is ‘vscardcfg’ of course. Additionally steps 3 and 4 are only required if ‘serial’ does not identify the card. Most models have been submitted as patch information to ‘serial’, later models will get direct support later.

## 5.4 Card Hardware Parameters

This is a list of all VScom cards available at time of writing. The given values will show how to identify the model, get the configuration of the card and thus extract the parameters of the serial ports.

For example in “/proc/pci” you’ll find information about a VScom 800L UPCI card like this:

```
Bus 0, device 11, function 0:
  Serial controller: Unknown vendor Unknown device (rev 0).
  Vendor id=14d2. Device id=8080.
  Medium devsel. Fast back-to-back capable. IRQ 10.
  I/O at 0xb000.      [BAR0]
  I/O at 0xa800.      [BAR1]
  I/O at 0xa400.      [BAR2]
  I/O at 0xa000.      [BAR3]
  I/O at 0x9c00.      [BAR4]
```

The text in brackets is added here as a comment. Please note Vendor- and Device-ID, as well as the IRQ. I/O addresses are given as BAR<n>, where <n> is from 0 to 5, depending on the model of the card. Serial ports are supported via I/O ranges, and all ports on one card share the same IRQ.

Each BAR may represent several serial ports, or none at all. If there is one serial port, the given address is the address of the port. If there are more serial ports, they are located at the given address, plus an offset of 8 from port to port. In this example of VScom 800L UPCI the serial ports are located at 0xA800, 0xA400, 0x9C00, 0x9C08, 0x9C10, 0x9C18, 0x9C20 and 0x9C28. The IRQ is 10, obviously.

In the table below this is given like BAR1, BAR2, BAR4:6. The table lists the card name, Vendor- and Device-ID (14d2:8080 here) and the uart model. For all VScom PCI cards baud\_base is 921600, unless this is explicitly configured otherwise (chapter ??).

Card	Ident	addresses	uart	remark
200S PCI	10b5:1103	BAR1, BAR2	16550A	phased out
210S PCI	10b5:1078	BAR1, BAR2	16550A	phased out
400S PCI	10b5:1077	BAR1:4	16550A	phased out
800S PCI	10b5:1076	BAR1:8	16550A	phased out
100L	14d2:8010	BAR1	16550A	
110L	14d2:8011	BAR1	16550A	
200L	14d2:8020	BAR1, BAR2	16550A	
210L	14d2:8021	BAR1, BAR2	16550A	
400L	14d2:8040	BAR1, BAR2, BAR4:2	16550A	
420L	14d2:8042	BAR1, BAR2, BAR4:2	16550A	
430L	14d2:8043	BAR1, BAR2, BAR4:2	16550A	
800L	14d2:8080	BAR1, BAR2, BAR4:6	16550A	
100H	14d2:a001	BAR0	16950	
110H	14d2:a001	BAR0	16950	
100HV2	14d2:e010	BAR0	16950	
110HV2	14d2:e010	BAR0	16950	
200H	14d2:a005	BAR0:2	16950	
210H	14d2:a005	BAR0:2	16950	
200HV2	14d2:e020	BAR0, BAR1	16950	
210HV2	14d2:e020	BAR0, BAR1	16950	
400H	14d2:a003	BAR0:4	16950	
410H	14d2:a003	BAR0:4	16950	
800H	14d2:a003	BAR0:4	16950	ports 1-4
800H	14d2:a004	BAR0:4	16950	ports 5-8
200I	14d2:8028	BAR4:2	16550A	
200I-SI	14d2:8028	BAR4:2	16550A	
400I	14d2:8048	BAR4:4	16550A	
800I	14d2:8088	BAR4:8	16550A	

Table 14: Hardware Parameters

## 6 Connector Definitions

The VScom PCI cards provide various connectors for the signals of the serial ports. Either DB9 male connectors are directly available on the card, or a common connector for all ports is mounted on the bracket. This way each model only needs one slot to fit in.

The signal assignment for RS 232 on DB9 male connectors is defined by the term “RS 232” already. The same is for the female type, as well as for DB25 male and female connectors. Below is the signal definition for all connectors used by the VScom H-Series Cards. DB9 male signals are mentioned for completeness.

### 6.1 DB62 female

This connector is used for many of VScom products with eight serial ports. So customers can easily change from an old product for ISA bus to PCI, without worrying about the external components.

The connector is common, so each signal name appears eight times, it will have an index like (3) for the third serial port of the card. The serial ports are neither isolated nor isolated from each other, so GND is common and has no index.

Pin	RS 232	Pin	RS 232	Pin	RS 232
1	TxD (1)	22	RxD (1)	43	CTS (1)
2	DTR (1)	23	DSR (1)	44	RTS (1)
3	RxD (2)	24	DCD (1)	45	GND
4	DSR (2)	25	TxD (2)	46	CTS (2)
5	DCD (2)	26	DTR (2)	47	RTS (2)
6	TxD (3)	27	RxD (3)	48	CTS (3)
7	DTR (3)	28	DSR (3)	49	RTS (3)
8	RxD (4)	29	DCD (3)	50	GND
9	DSR (4)	30	TxD (4)	51	CTS (4)
10	DCD (4)	31	DTR (4)	52	RTS (4)
11	RxD (5)	32	GND	53	CTS (5)
12	DSR (5)	33	TxD (5)	54	RTS (5)
13	DCD (5)	34	DTR (5)	55	GND
14	TxD (6)	35	RxD (6)	56	CTS (6)
15	DTR (6)	36	DSR (6)	57	RTS (6)
16	RxD (7)	37	DCD (6)	58	GND
17	DSR (7)	38	TxD (7)	59	CTS (7)
18	DCD (7)	39	DTR (7)	60	RTS (7)
19	RxD (8)	40	GND	61	CTS (8)
20	DSR (8)	41	TxD (8)	62	RTS (8)
21	DCD (8)	42	DTR (8)		

Table 15: DB62 female Connector

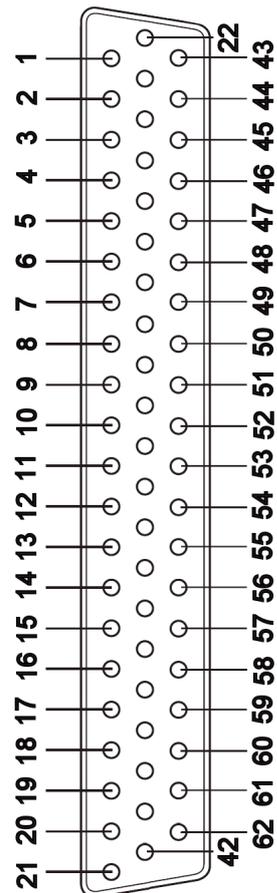


Figure 29: DB62 female Connector

## 6.2 DB37 female

This common connector is for four serial ports. So each signal name appears four times, it will have an index like (3) for the third serial port of the card. The serial ports are neither isolated nor isolated from each other, so GND is common and has no index. Pin 1 is a no-connect pin.

Pin	RS 232	Pin	RS 232
1	.nc.	20	RI (3)
2	DCD (3)	21	DTR (3)
3	GND	22	DSR (3)
4	CTS (3)	23	RTS (3)
5	RxD (3)	24	TxD (3)
6	RI (4)	25	DCD (4)
7	DTR (4)	26	GND
8	DSR (4)	27	CTS (4)
9	RTS (4)	28	RxD (4)
10	TxD (4)	29	RI (2)
11	DCD (2)	30	DTR (2)
12	GND	31	DSR (2)
13	CTS (2)	32	RTS (2)
14	RxD (2)	33	TxD (2)
15	RI (1)	34	DCD (1)
16	DTR (1)	35	GND
17	DSR (1)	36	CTS (1)
18	RTS (1)	37	RxD (1)
19	TxD (1)		

Table 16: DB37 female Connector

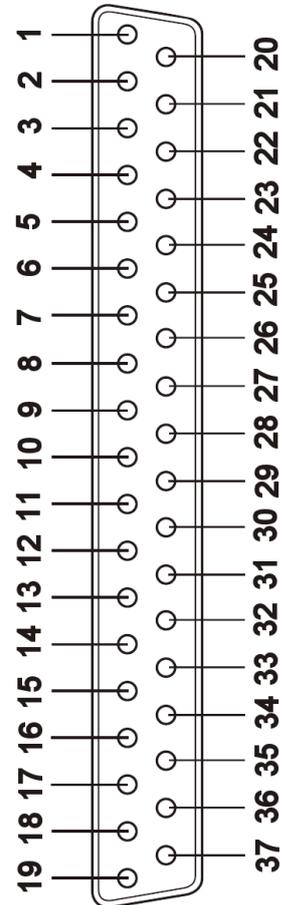


Figure 30: DB37 female Connector

### 6.3 DB9 male

Usually this type of connector is used by customers.

Pin	RS 232	Pin	RS 232
1	DCD	6	DSR
2	RxD	7	RTS
3	TxD	8	CTS
4	DTR	9	RI
5	GND		

Table 17: DB9 male Connector

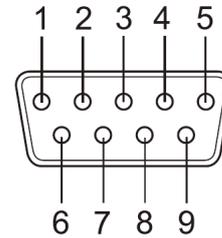


Figure 31: DB9 male Connector

### 6.4 DB25 male RS 232

This type of connector is only available on the VScom 210L UPCI card, but there are adapter options providing such. The pins not mentioned are a no-connect on the DB25, or specified in adapters documentation.

Pin	RS 232	Pin	RS 232
2	TxD	7	GND
3	RxD	8	DCD
4	RTS	20	DTR
5	CTS	22	RI
6	DSR		

Table 18: DB25 male RS 232

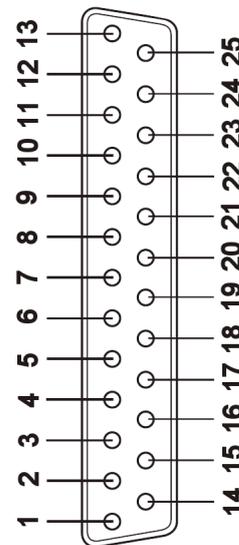


Figure 32: DB25 male RS 232

## 6.5 DB25 female parallel (LPT)

The parallel port is also known as printer port or LPT. The signal assignment has been defined in the original IBM PC, and did not change since then. The LPT port is a subset of the Centronics interface.

Pin	LPT	Pin	LPT
1	/STROBE	14	/AUTOFD
2	D0	15	INIT
3	D1	16	/ERROR
4	D2	17	/SELIN
5	D3	18	GND
6	D4	19	GND
7	D5	20	GND
8	D6	21	GND
9	D7	22	GND
10	/ACK	23	GND
11	BUSY	24	GND
12	PE	25	GND
13	SEL		

Table 19: DB25 parallel Connector

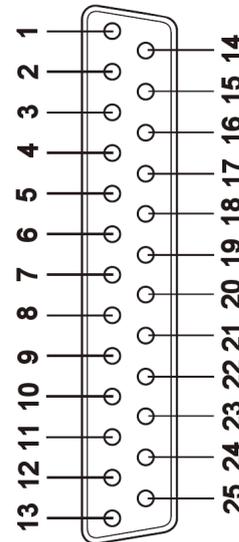


Figure 33: DB25 female LPT

## 6.6 DB25 female RS 232

This type of connector is not available on any card, however there are adapter options providing such. The pins not mentioned are a no-connect on these adapters, or specified in the adapters documentation.

This connector looks exactly the same as the parallel port connector 6.5. Here is the signal assignment.

Pin	RS 232	Pin	RS 232
2	RxD	7	GND
3	TxD	8	DCD
4	CTS	20	DSR
5	RTS	22	RI
6	DTR		

Table 20: DB25 female RS 232

## 6.7 RJ45 for RS 232

This type of connector is available on the VScom 400H RJ45 PCI and the VScom 400L RJ UPCI card. Adapter cables to DB9 are available as an option.

Pin	RS 232	Pin	RS 232
1	RTS	5	RxD
2	DTR	6	DCD
3	GND	7	DSR
4	TxD	8	CTS

Table 21: RJ45 for RS 232

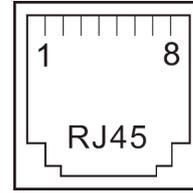


Figure 34: RJ45 for RS 232

## 7 History

**March\_2009** First Release of this manual

**July 2009** Added VScom 400H RJ45 PCI

**February 2010** Installation in Windows 7 added