**OXFORD
SEMICONDUCTOR**

OX16C95x
OX16PCI954

HIGH-PERFORMANCE UART FAMILY

REFERENCE DRIVERS

This document describes installation and use of the Oxford Semiconductor reference drivers for the OX16C95x high performance UART family, including the OX16PCI954 Quad UART with PCI interface.

Version 2.0 (01 February 2000)

1 OVERVIEW

Reference drivers and utilities for the Oxford Semiconductor UART products are supplied to demonstrate functionality under Windows 95/98, Windows NT4 and Windows 2000. These drivers have been tested using Oxford Semiconductor development boards in a range of PC systems.

2 LICENCE AGREEMENT

This software and the accompanying files are provided "as is" and without warranties as to performance or merchantability or any other warranties whether express or implied. The receiver assumes all risk and liability for loss, damage, claims or expense resulting from use, possession or resale of any software products furnished by Oxford Semiconductor Ltd.

The receiver agrees to indemnify, defend and hold harmless Oxford Semiconductor Ltd. and its officers, agents, and employees from and against any and all claims, liability, loss, damage or expense, including reasonable attorney's fees, arising from or by reason of receivers' use, possession or resale with respect to any of the software products furnished by Oxford Semiconductor Ltd. pursuant to this agreement, and such obligation shall survive acceptance of said products therefore by receiver.

3 CONTENTS

4   OVERVIEW and INSTALLATION

4.1   OX16PCI954

4.1.1   Windows 95/98 and Windows 2000 drivers for OX16PCI954

Overview
The architecture of the Windows 9x and Windows 2000 drivers is very similar. The driver set consists of Plug and Play bus drivers, which create device objects for each UART / parallel port available. Each UART is hooked by a single-port driver, which enables the following 16C95x features:
- 128-byte receiver and transmitter FIFOs
- Adjustable receiver and transmitter interrupt trigger level
- Automatic flow control
- Quad speed (using Times clock register)
- Flexible baud rate generation up to 60Mbaud (50Mbaud on OX16C954)
- RS485 half-duplex configuration
- Memory-mapped operation (Windows 2000 driver only)

The parallel port is hooked to the generic driver supplied with Windows.

The drivers will recognise the OX16PCI954 in any of its default configurations, and also recognise the reserved device ID 0x950A as a two serial port device (see application note "Custom Drivers" for more information).

Installation
Install the hardware by following system instructions. Power up the system. Then follow the instructions below depending on which operating system is installed. **To upgrade** the Windows 9x driver from a previous release, first boot up the system with the existing drivers. Right-click "My Computer", select Properties, then Device Manager. Double-click "Multifunction" and remove all Oxford Semiconductor devices from this category. Restart the machine, then follow the instructions below.

Windows 98
1. The system will detect the device and start the "Add new hardware" wizard. Click Next to continue.
2. The system will suggest that it searches for the best driver for the device. Click Next to confirm.
3. If the driver files are on a floppy disk, insert it into the drive now. If the driver files are on the hard disk, point the wizard at the correct directory.
4. Click Next to continue.
5. The system searches for the driver. If it doesn't find the Oxford Semiconductor device, make sure the driver files are present and the Wizard knows where to find them.
6. Once the system has found the driver, click Next to continue.
7. The system installs the driver.
8. Click Finish.
9. If you are using the parallel port, the system detect it next. Re-iterate steps 2-8, but specify that the driver files are located in the 9x_parallel directory.
10. Once the main drivers are installed, the system installs the UART ports automatically.

Windows95 (OSR2¹)
Note: the system prompts may vary between different versions of Windows. If in doubt consult the user manual.
1. The system detects the device and starts the "Update Device Driver" wizard.
2. If you have the driver files on a floppy disk, insert it into the drive now. Click Next to continue.
3. The system searches for the driver. If it doesn't find the Oxford Semiconductor device, click 'Other Locations' and enter the full path to the driver files.
4. Once the system has found the driver, click Finish.
5. The system installs the device, and then goes through the same steps for the local bus / parallel port. Note: the parallel port drivers are located in the 9x_Parallel directory.
6. Once the main drivers are installed, the system installs the UART ports automatically. (Note: due to a Win95 bug you may have to re-enter the path for it to find the files).

Windows95 (original release[1])
Note: the system prompts may vary between different versions of Windows. If in doubt consult the user manual.
1. The system will detect the device and ask which driver to install. Select "Driver from disk provided by hardware manufacturer", and click OK.
2. If you have the driver files on a floppy disk, insert it into the drive. Otherwise enter the full path to the driver files. Click OK.
3. The system installs the device, and then goes through the same steps for the local bus / parallel port. Note: the parallel port drivers are located in the 9x_Parallel directory.
4. Once the main drivers are installed, the system installs the UART ports automatically. (Note: due to a Win95 bug you may have to re-enter the path for it to find the files).

Windows2000
1. The system will detect the device and ask which driver to install. Click Next
2. Select "Search for a suitable driver for my device", and click Next.
3. If you have the driver files on a floppy disk, insert it into the drive and check the "Floppy disk drives" box. Otherwise check the

4. If "Specify a location" was checked, you will have to enter the path to search and click OK.
5. The system will search for a driver. When it finds it, click Next. The files will be copied and the driver installed. Click Finish. The system then installs the other logical function (parallel port / local bus) in the same manner.
6. Once the PCI devices are installed, the system will install the UART ports automatically.

Note 1: To check your version of Windows 95, right-click on "My Computer" and select "Properties". The System Properties page will appear. If the version is 4.00.950 B or C, you are using OSR2. If the letter is A or is absent, you are running the original releases.

4.1.2 Windows NT4 driver for OX16PCI954

Overview
The Windows NT4 driver creates a Device Object for each port, and registers this with the operating system. The serial port driver enables the following 16C95x features:
- 128-byte receiver and transmitter FIFOs
- Optimized receiver interrupt handling – uses readable fill levels and thresholds
- Automatic clock speed detection
- Memory-mapped operation for efficient throughput

A separate parallel port driver is provided, which creates the device Object and registers it with the Printer class driver.

The Windows NT4 driver searches the PCI bus for the VendorID / DeviceID combinations specified in the oxser.inf file. If the chip has been reconfigured with the EEPROM (e.g. DeviceID reconfigured as 0x950A to be a two-serial port solution) the oxser.inf file will need to be modified (see application note "Custom Drivers" for more information).

Installation
Install the hardware by following system instructions. Power up the system. Then follow the instructions below depending on which functions are required. To **upgrade** the NT4 driver, simply install over the existing version.

Serial driver
1. Run "Install_Serial.exe" from the \NT4_Serial subdirectory on the disk.
2. When the install application starts, click Next to continue
3. Ensure 'install' is selected, then click Next.
4. Click OK to accept the licence agreement.
5. The system will install the driver and start it. The ports are immediately ready for use

Parallel driver
1. Run "Install_Parallel.exe" from the \NT4_Parallel subdirectory on the disk.
2. When the install application starts, click Next to continue
3. Ensure 'install' is selected, then click Next.
4. Click OK to accept the licence agreement.
5. The system will install the driver and prompt for a restart. The port is available for use after the restart.

4.2   OX16C95x (standard UARTs)

4.2.1   Windows 95/98 and Windows 2000 drivers for standard UARTs

<u>Overview</u>
The drivers for standard ports use the same port driver as the OX16PCI954 set, therefore the same features are available:
- 128-byte receiver and transmitter FIFOs
- Adjustable receiver and transmitter interrupt trigger level
- Automatic flow control
- Quad speed (using Times clock register)
- Flexible baud rate generation up to 60Mbps (50Mbps on OX16C954)
- RS485 half-duplex configuration

<u>Installation</u>
Windows will identify all COM ports as standard communications ports. Once a 95x COM port is installed as a generic port, update the driver as follows:
1. Click on Start / Settings / Control Panel
2. Double click "System" to bring up the system properties dialog box
3. Select the "Device Manager" tab of this dialog (with the "View devices by type" option selected.
4. Now click on the [+] next to the "Ports (COM & LPT)" icon to show the list of installed COM ports (If this section is not present or no COM ports are listed, there are no serial devices installed in Windows. To install COM port hardware, consult your Windows / Hardware documentation.)
5. Double click on the port you want to install the new driver for to bring up the settings dialog for that port
6. Select the driver tab and click the update/change driver button.
7. **Windows 95**
Click on "Have Disk" and specify the location of the driver files. Click OK
The files will now be copied and the driver installed.
8. **Windows 98**
When the "Update Device Driver Wizard" asks what you want Windows to do, select "Search for a better driver" and then click next.
When asked where to select the "Specify Location" box and type the location of the driver files. Click 'next' and the driver will be installed.
9. **Windows 2000**
The system starts the Upgrade Device Driver Wizard. Click 'Next'
Check "Search for a suitable driver for my device" and click Next. Point the wizard at the driver files (on floppy disk or fixed disk) and click Next. You may have to enter a path to the driver files.
The system will find a selection of suitable drivers. Check "Install one of the other drivers" and then 'Next'.
Install the "Oxford Semiconductor OX16C95x Communications port"
The files are copied. Click finish to exit the Wizard (and possibly restart)
Alternatively, the driver can be installed manually for a new device without running the auto-detect routines.

4.2.2   Windows NT4 note

There is no enhanced mode driver for standard UARTs in Windows NT4.0. This is because the generic driver will detect any ISA ports and use them itself.

5 USING THE SERIAL DRIVER

The serial drivers for Windows 9x and Windows 2000 make use of a number of enhanced 95x UART features, and are highly configurable. This section describes the configuration utilities, accessible through Device Manager, which can be used to enable and configure the various features

The driver will install the correct number of serial ports and Windows will assign COM numbers to them (e.g. COM5). You can then attach modems etc. to the ports and make use of them in the same fashion as any other generic port. The serial driver has extra configuration options available over and above the standard 'settings' tab supplied with the generic 16550 driver. To adjust any of the UART parameters, follow these steps:
1. Click on Start / Settings / Control Panel
2. Double click "System" to bring up the system properties dialog box
3. Select the "Device Manager" tab of this dialog (with the "View devices by type" option selected
4. Now click on the [+] next to the "Ports (COM & LPT)" icon to show the list of installed COM ports
5. Double click on an Oxford Semiconductor port to display the settings dialog for that port

The enhanced features are configurable through three dialog pages:
Settings   - **Standard port settings found on normal COM port settings pages**
Data Rate   - **Advanced data rate selection options**
FIFOs   - **Device mode selection and FIFO trigger level settings**

5.1.1 Windows 95/98 configuration

Settings
This page contains the standard Baud rate / Data bits / Parity / Stop bits and Flow control options found on most COM port settings pages. These settings modify the **default** settings used by Windows. Most applications that use COM ports will override these settings with their own comms parameters. Note that the baud rate selected by an application will be scaled up if a faster crystal is used
This page also provides configurable RS485 half-duplex operation. RS232 applications will not use this, and the DTR pin should be configured as 'normal'. However if RS485 line drivers are used, the driver can configure the DTR pin to enable the transmitter, in active-high or active-low form.

Data Rate
This page provides a list of common crystal values used with COM ports. Select "Detect crystal frequency" to detect the input clock frequency to the UART. (This requires that the port is not currently in use by another application).
[Note: The "Automatic crystal selection" option will not work on COM10 and higher in Windows 9x; this is a limitation of 16-bit DLLs. For high-numbered ports the user should select the crystal frequency manually.]

The quad speed option will multiply all application selected baud rates by 4 by utilising the OX16C95x Times clock register (TCR)

Selecting "User Defined Override Speed" enables the fully variable baud rate slider and "Clock Pre-Divisor" list box for more flexible baud rate generation. The actual selected baud rate is displayed above the baud rate slider.
The clock pre-divisor is used to divide the input clock prior to baud rate generation. This means a high speed crystal (E.g 50MHz) can be pre-divided to generate standard baud rates (In this case 50 / 27.125 = 1.8433 MHz, which will emulate a 1.8432 MHz crystal with less than 0.01% bit rate error). Alternatively, the pre-divisor could be switched off to allow data rates up to 12.5Mbps to be generated.

FIFOs

This page provides three device modes. 450 mode, which forces the UART to operate in byte mode (i.e with the FIFO disabled), 550 mode which operates with 16 byte deep FIFOs, 550 trigger levels and driver generated flow control (non-automated). Finally, 950 mode, which operates with full 128 byte FIFOs, fully adjustable trigger levels and automated flow-control.

The four sliders allow adjustment of the various trigger levels in 550 and 950 modes. These are described below:
- Transmitter FIFO interrupt trigger level - When the level of data in the transmit FIFO falls below this value, a transmitter interrupt is triggered. Setting this value to zero will not trigger an interrupt until the transmitter is completely idle. In 450 and 550 mode this is fixed at 1.
- Receiver FIFO interrupt trigger level - When the level of data in the receiver FIFO reaches this value, a receiver data interrupt is triggered. In 450 mode this is fixed at 1, In 550 mode it can take 4 discrete values 1,4,8 & 14.
- Flow On flow-control limit - When the level of data in the receiver FIFO reaches this value as data is read from the FIFO, a handshake is triggered to instruct the remote transmitter that it is free to transmit data (E.g. Transmit an XON character to the remote machine). In 450 and 550 mode this level is fixed at 1.
- Flow Off flow-control limit - When the level of data in the receiver FIFO reaches this value as it fills up, a handshake is triggered to instruct the remote transmitter to stop sending data as the FIFO is becoming full. (E.g. Transmit an XOFF character to the remote machine). In 450 mode this level is fixed at 1, in 550 mode it assumes the same value as the receiver FIFO interrupt trigger level.

The FIFO trigger levels can be fine tuned to gain optimum performance, depending on system performance, baud rate used, levels of serial traffic etc. The default Receiver FIFO trigger level is preset at a value of 64; this should be increased for higher performance, but in some cases a high trigger level will result in the port not detecting PnP serial devices.

5.1.2 Windows 2000 configuration

Settings

This page contains the standard Baud rate / Data bits / Parity / Stop bits and Flow control options found on most COM port settings pages. These settings modify the **default** settings used by Windows. Most applications that use COM ports will override these settings with their own comms parameters.

This page also provides configurable RS485 half-duplex operation. RS232 applications will not use this, and the DTR pin should be configured as 'normal'. However if RS485 line drivers are used, the driver can configure the DTR pin to enable the transmitter, in active-high or active-low form. Note: if RS422/485 line drivers are selected, the driver will not allow DTR/DSR flow control, as these pins are not defined in RS422 protocols.

Data Rate

This page provides a list of common crystal values used with COM ports. Select "Detect crystal frequency" to detect the input clock frequency to the UART. (This requires that the port is not currently in use by another application).

The baud rate can optionally be adjusted according to the data rate required. To enable the advanced baud rate configuration options, deselect the "Use default baud rate" box. In normal operation, the driver will generate the desired baud rate from the crystal frequency. The quad speed option will multiply all application selected baud rates by 4 by utilising the OX16C95x Times clock register (TCR). The driver can multiply this baud rate, or divide it using the Clock Prescaler register (CPR).

The clock pre-divisor is used to divide the input clock prior to baud rate generation. This means a high speed crystal (E.g 50MHz) can be pre-divided to generate standard baud rates (In this case 50 / 27.125 = 1.8433 MHz, which will emulate a 1.8432 MHz crystal with less than 0.01% bit rate error). Alternatively, the pre-divisor could be switched off to allow data rates up to 12.5Mbps to be generated.

FIFOs
This page is used to configure the FIFO trigger levels, i.e. at what fill levels the device will generate an interrupt, or apply automatic flow control. In addition, the FIFOs can be completely disabled; although this is not recommended for normal operation.

The four sliders allow adjustment of the various trigger levels in 550 and 950 modes. These are described below:
- Transmitter FIFO interrupt trigger level - When the level of data in the transmit FIFO falls below this value, a transmitter interrupt is triggered. Setting this value to zero will not trigger an interrupt until the transmitter is completely idle.
- Receiver FIFO interrupt trigger level - When the level of data in the receiver FIFO reaches this value, a receiver data interrupt is triggered.
- Flow On flow-control limit - When the level of data in the receiver FIFO reaches this value as data is read from the FIFO, a handshake is triggered to instruct the remote transmitter that it is free to transmit data (E.g. Transmit an XON character to the remote machine).
- Flow Off flow-control limit - When the level of data in the receiver FIFO reaches this value as it fills up, a handshake is triggered to instruct the remote transmitter to stop sending data as the FIFO is becoming full. (E.g. Transmit an XOFF character to the remote machine).

The FIFO trigger levels can be fine tuned to gain optimum performance, depending on system performance, baud rate used, levels of serial traffic etc.

6   Parallel port driver in Windows 95/98

The parallel port driver will set the hardware to use legacy addresses 378h if available, otherwise it will be forced to use the address allocated by the system BIOS. Windows will allocate an LPT number (e.g. LPT2) to the port; then it can be used in the same fashion as any generic parallel port. Note: some peripheral devices such as ZIP drives will not operate correctly if the parallel port is not located at address 278 or 378. In this case, the conflicting devices such as the motherboard port will need to be moved to other addresses.

Configuration
The parallel port interrupt can be enabled / disabled using the configuration applet found in Device Manager. This may be necessary if using Windows 98 and Direct Cable connection. To adjust the setting, follow these steps:
1. Click on Start / Settings / Control Panel
2. Double click "System" to bring up the system properties dialog box
3. Select the "Device Manager" tab of this dialog (with the "View devices by type" option selected)
4. Now click on the [+] next to the "Multifunction Devices" icon to show the list of Oxford Semiconductor parts
5. Double click on the Oxford Semiconductor PCI Parallel port to display the settings dialog for that port
6. Click the "settings" tab of this dialog, and enable / disable the interrupt as required

It is also possible for the port to use address 278h/678h. However, many systems do not support this option due to a contention with ISAPNP bus enumeration. Check this option to enable it; however be prepared to change it back if the port ceases to work.

7   CONTACT INFORMATION

For further information please contact:

Oxford Semiconductor Ltd.
69 Milton Park
Abingdon
Oxfordshire
OX14 4RX
United Kingdom

Telephone:	+44 (0)1235 824900
Fax:	+44 (0)1235 821141
Sales e-mail:	sales@oxsemi.com
Tech support e-mail:	support@oxsemi.com
Web site:	http://www.oxsemi.com

8   DISCLAIMER

Oxford Semiconductor believes the information contained in this document to be accurate and reliable. However, it is subject to change without notice. No responsibility is assumed by Oxford Semiconductor for its use, nor for infringement of patents or other rights of third parties. No part of this publication may be reproduced, or transmitted in any form or by any means without the prior consent of Oxford Semiconductor Ltd. Oxford Semiconductor's terms and conditions of sale apply at all times.