OXFORD SEMICONDUCTOR

OX16PCI954

QUAD UART WITH PCI INTERFACE

CUSTOM DRIVERS

This document describes recommended procedures to reconfigure the OX16PCI954 hardware and reference drivers in order to identify custom add-in card configurations to device drivers.

Version 1.1 (25 August 1999)

1   OVERVIEW

The primary function of the OX16PCI954 is to provide four serial ports, and a local bus (with more serial ports) or a parallel port. However the chip is very flexible, so it is possible to use it for other solutions, for example:-

- 2 serial, 1 parallel card
- 1 parallel card
- 4 serial ports + 1 custom FPGA synchronous I/O controller on the local bus

If a designer wishes to use the OX16PCI954 chip for another purpose, it is necessary to change the identification fields in PCI Configuration space. These are the VENDOR ID, DEVICE ID, SUBSYSTEM VENDOR ID and SUBSYSTEM ID fields. In this way, it is possible for the Plug 'n' Play system to identify different boards (even though they use the same chip) and load the correct drivers.

There are three recommended approaches to uniquely identify an add-in card using the OX16PCI954.

1. The best solution is for the board manufacturer to obtain their own VENDOR ID from the PCI special interest group (PCISIG). Then they should use the EEPROM facility on the OX16PCI954 to reprogram the VENDOR ID and SUBSYSTEM VENDOR ID fields with this value, and they choose any DEVICE ID and SUBSYSTEM ID value.
2. Another recommended approach is to leave the VENDOR ID and DEVICE ID as the default values set by Oxford Semiconductor. The SUBSYSTEM VENDOR ID should be set to board manufacturer's unique value assigned by the PCISIG, then they can choose the SUBSYSTEM ID.
3. For small vendors, it is sometimes possible to use Oxford Semiconductor's identification fields. In these cases, Oxford Semiconductor will assign a subsystem ID to the board vendor, which is the only value that needs to be changed using the EEPROM. To request a subsystem ID value, please contact Oxford Semiconductor with a full functional description of the add-in card being designed.

The first solution is the best because Windows 95 and Windows NT do not recognise subsystem ID values. Therefore they will not be able to distinguish between different add-in cards. Windows 98 and Windows 2000 will be able to identify different cards, provided that the driver .inf file specifies the subsystem ID values, in the form of:

```
PCI\VEN_1415&DEV_9501&SUBSYS_00011415
```

Board vendors should NEVER assign a new DEVICE ID or SUBSYSTEM ID unless they have obtained their own VENDOR ID from the PCISIG. Oxford Semiconductor reserve the right to refuse to allocate a subsystem ID to a board vendor.

2   Examples

2.1   4-serial, 2 parallel card

To do this it is necessary to use the internal UARTs, and have two external parallel port chips on the local bus. In this case the board vendor can use the default ID for the serial ports, but must have a different ID for the local bus. They can contact Oxford Semiconductor for a subsystem ID for the local bus, or obtain their own VENDOR ID and define their own drivers for the entire card. The three possible approaches are therefore:-

Use the board vendor's own PCI VENDOR ID (in this example case, 0x1234)

	4 serial ports (internal UARTs) FUNCTION 0	2 parallel ports (Local bus) FUNCTION 1
VENDOR ID	0x1234	0x1234
DEVICE ID	Chosen by vendor	Chosen by vendor
SUBSYSTEM VENDOR ID	0x1234	0x1234
SUBSYSTEM ID	Chosen by vendor	Chosen by vendor

In this case, the .inf file needs to identify the devices as:-
```
PCI\VEN_1234&DEV_XXXX
PCI\VEN_1234&DEV_YYYY
```

Use the board vendor's own PCI VENDOR ID as the SUBSYSTEM VENDOR ID (in this case, 0x1234)

	4 serial ports (internal UARTs) FUNCTION 0	2 parallel ports (Local bus) FUNCTION 1
VENDOR ID	0x1415 (default)	0x1415 (default)
DEVICE ID	0x9501 (default)	0x9511 (default)
SUBSYSTEM VENDOR ID	0x1234	0x1234
SUBSYSTEM ID	Chosen by vendor	Chosen by vendor

In this case, the .inf file needs to identify the devices as:-
```
PCI\VEN_1415&DEV_9501&SUBSYS_XXXX1234
PCI\VEN_1415&DEV_9511&SUBSYS_YYYY1234
```

Obtain subsystem ID from Oxford Semiconductor (example 0x00FF), and use other values as default

	4 serial ports (internal UARTs) FUNCTION 0	2 parallel ports (Local bus) FUNCTION 1
VENDOR ID	0x1415 (default)	0x1415 (default)
DEVICE ID	0x9501 (default)	0x9511 (default)
SUBSYSTEM VENDOR ID	0x1415 (default)	0x1415 (default)
SUBSYSTEM ID	0x0001 (default)	0x00FF

In this case, the .inf file needs to identify the devices as:-
```
PCI\VEN_1415&DEV_9501&SUBSYS_00011415
PCI\VEN_1415&DEV_9511&SUBSYS_00FF1415
```

2.2   2-serial, 1 parallel card

It is not possible to limit the hardware to only enable two serial ports, so this solution merely requires a driver in the enumerator driver, which ensures that it only registers two ports. Oxford Semiconductor have assigned a DEVICE ID of 0x950A for this, so the recommended solution is:-

	2 serial ports (internal UARTs) FUNCTION 0	1 parallel ports (Local bus) FUNCTION 1
VENDOR ID	0x1415	0x1415
DEVICE ID	0x950A	0x9513
SUBSYSTEM VENDOR ID	0x1415	0x1415
SUBSYSTEM ID	Default	Default

The OX16PCI954 drivers version 2.51 and greater include identification of the 0x950A device ID as 2-serial ports.
The card designer should use the first two ports as these are the ones which will be used by the driver.
For Windows 9x, setting the device ID using the EEPROM will be the only modification required. For Windows NT4, the oxser.inf file needs to be modified (all occurences of 9501 need to be changed to 950A, BEFORE INSTALLATION OF THE DRIVERS. Please contact support@oxsemi.com for further details.

2.3   1 parallel port only

It is not possible to disable the UARTs on the device, so this solution will require the user to install a null driver for the UARTs. Again Oxford Semiconductor already have assigned a DEVICE ID for this (value 0x9500), so the recommended solution is:-

	NULL function FUNCTION 0	1 parallel ports (Local bus) FUNCTION 1
VENDOR ID	0x1415	0x1415
DEVICE ID	0x9500	0x9513
SUBSYSTEM VENDOR ID	0x1415	0x1415
SUBSYSTEM ID	Default	Default

The OX16PCI954 reference drivers will recognise the device ID 0x9500 as a null function, and load a null driver for this device.