
PCIe Cards User's Manual

PCIe-S118, PCIe-S148
Linux Software Manual

Warranty

All products manufactured by ICP DAS are warranted against defective materials for a period of one year from the date of delivery to the original purchaser.

Warning

ICP DAS assume no liability for damages consequent to the use of this product. ICP DAS reserves the right to change this manual at any time without notice. The information furnished by ICP DAS is believed to be accurate and reliable. However, no responsibility is assumed by ICP DAS for its use, nor for any infringements of patents or other rights of third parties resulting from its use.

Copyright

Copyright 2015 by ICP DAS. All rights are reserved.

Trademark

The names used for identification only may be registered trademarks of their respective companies.

Tables of Content

1.	LINUX SOFTWARE INSTALLATION	3
1.1	LINUX DRIVER INSTALLING PROCEDURE	3
1.2	LINUX DRIVER UNINSTALLING PROCEDURE.....	4
2.	PCIE-S1X8 SERIES LINUX DEMO	5
2.1	DEMO SB_TEST	5

1. Linux Software Installation

The PCIe-S118 & PCIe-S148 can be used in linux kernel 2.6.32 to 3.6.0. For Linux O.S, the recommended installation and uninstall steps are given in Sec 1.1 ~ 1.2

1.1 Linux Driver Installing Procedure

Step 1: Copy the linux driver “PCIe-S1x8-Series.tar.gz” in the directory “NAPDOS\MultiPort/Linux” of the companion CD or download the latest driver from our website to the linux host.

Step 2: You must use the ‘root’ identity to compile and install PCIe linux driver.

Step 3: Decompress the tarball “PCIe-S1X8-Series.tar.gz”. Please refer Fig-1.1

```
[root@localhost PCIe-S118_PCE-S148]# ls
PCIe-S1X8-Series.tar.gz
[root@localhost PCIe-S118_PCE-S148]# tar zxvf PCIe-S1X8-Series.tar.gz
async_multiport/
async_multiport/sb test.c
async_multiport/Makefile
Makefile
readme
xr17v35x.c
[root@localhost PCIe-S118_PCE-S148]# ls
async_multiport Makefile PCIe-S1X8-Series.tar.gz readme xr17v35x.c
[root@localhost PCIe-S118_PCE-S148]# █
```

Fig-1.1

Step 4: Execute the device file.

Please follow below steps to compile and install driver:

```
#make
```

```
#insmod xr17v35x.ko
```

Check xr17v35x.ko is installed.

```
#lsmod | grep xr17v35x
```

Please refer to Fig-1.2

```
[root@localhost PCIe-S118_PCIe-S148]# make
make -C /lib/modules/`uname -r`/build SUBDIRS=/root/PCIe-S118_PCIe-S148 modules
make[1]: Entering directory `/usr/src/kernels/linux-3.1'
  CC [M] /root/PCIe-S118_PCIe-S148/xr17v35x.o
/root/PCIe-S118_PCIe-S148/xr17v35x.c: In function `receive_chars':
/root/PCIe-S118_PCIe-S148/xr17v35x.c:577: warning: ISO C90 forbids mixed declara
/root/PCIe-S118_PCIe-S148/xr17v35x.c: In function `serialxr_startup':
/root/PCIe-S118_PCIe-S148/xr17v35x.c:1076: warning: suggest parentheses around a
value
Building modules, stage 2.
MODPOST 1 modules
  CC /root/PCIe-S118_PCIe-S148/xr17v35x.mod.o
  LD [M] /root/PCIe-S118_PCIe-S148/xr17v35x.ko
make[1]: Leaving directory `/usr/src/kernels/linux-3.1'
[root@localhost PCIe-S118_PCIe-S148]# insmod xr17v35x.ko
[root@localhost PCIe-S118_PCIe-S148]# lsmod | grep xr17v35x
xr17v35x          74763  16
[root@localhost PCIe-S118_PCIe-S148]# █
```

Fig-1.2

1.2 Linux Driver Uninstalling Procedure

Step 1: Type “rmmod xr17v35x” to remove PCIe-S1x8 Series driver.

Step 2: Type ‘lsmod | grep xr17v35’ to check driver is removed. Please refer to Fig-1.3

```
[root@localhost PCIe-S118_PCIe-S148]# rmmod xr17v35x
[root@localhost PCIe-S118_PCIe-S148]# lsmod | grep xr17v35x
[root@localhost PCIe-S118_PCIe-S148]# █
```

Fig-1.3

2. PCIe-S1x8 Series Linux Demo

Users can use demo “sb_test” to test comport send & receive in sub directory “async_multiport”. Please refer to Fig-2.1

```
[root@localhost PCIe-S118_PCIe-S148]# ls
async_multiport  modules.order  PCIe-S1X8-Series.tar.gz  xr17v35x.c
Makefile        Module.symvers  readme                  xr17v35x.ko
[root@localhost PCIe-S118_PCIe-S148]# cd async_multiport/
[root@localhost async_multiport]# ls
Makefile  sb_test  sb_test.c  sb_test.o
[root@localhost async_multiport]# █
```

Fig-2.1

2.1 Demo sb_test

This demo program has three mode to test: Loopback 、 Send 、 Receive, if you want to know how to use the sb_test, you just type the name without any argument, and you can see the method of usage. Please refer Fig-2.2

```
Makefile  sb_test  sb_test.c  sb_test.o
[root@localhost async_multiport]# ./sb_test
Usage: ./sb_test [Port Name] [Baudrate] [TestMode]
Port Name : /dev/ttyXR0 ~ /dev/ttyXR32
Baudrate  : 9600, 19200, ...
TestMode  : 0(Loopback) 1(Send) 2(Recv)
[root@localhost async_multiport]# █
```

Fig-2.2

Example:

```
#!/sb_test /dev/ttyMP0 9600 0
```

After you connected a loopback connector to a port, you can test using loopback mode. The test pattern are “abcdefghijklmnopqrstuvwxyz” and the program generated increase one more character from ‘a’ to ‘z’ repeatedly. Please refer Fig-2.3

```
[root@localhost async_multiport]# ./sb_test /dev/ttyXR0 9600 0
Loopback Test Mode !
a
ab
abc
abcd
abcde
abcdef
abcdefg
abcdefgh
abcdefghi
abcdefghij
abcdefghijk
abcdefghijkl
abcdefghijklm
abcdefghijklmn
abcdefghijklmno
abcdefghijklmnop
abcdefghijklmnopq
abcdefghijklmnopqr
abcdefghijklmnopqrs
abcdefghijklmnopqrst
abcdefghijklmnopqrstu
abcdefghijklmnopqrstuv
abcdefghijklmnopqrstuvw
^C
[root@localhost async_multiport]# █
```

Fig-2.3