LTPC Driver

This is the ALPHA version of the ltpc driver.

In order to use it, you will need at least version 1.3.3 of the netatalk package, and the Apple or Farallon LocalTalk PC card. There are a number of different LocalTalk cards for the PC; this driver applies only to the one with the 65c02 processor chip on it.

To include it in the kernel, select the CONFIG_LTPC switch in the configuration dialog. You can also compile it as a module.

While the driver will attempt to autoprobe the I/O port address, IRQ line, and DMA channel of the card, this does not always work. For this reason, you should be prepared to supply these parameters yourself. (see "Card Configuration" below for how to determine or change the settings on your card)

When the driver is compiled into the kernel, you can add a line such as the following to your /etc/lilo.conf:

```
append="ltpc=0x240,9,1"
```

where the parameters (in order) are the port address, IRQ, and DMA channel. The second and third values can be omitted, in which case the driver will try to determine them itself.

If you load the driver as a module, you can pass the parameters "io=", "irq=", and "dma=" on the command line with insmod or modprobe, or add them as options in a configuration file in /etc/modprobe.d/ directory:

```
alias lt0 ltpc # autoload the module when the interface is configured
options ltpc io=0x240 irq=9 dma=1
```

Before starting up the netatalk demons (perhaps in rc.local), you need to add a line such as:

```
/sbin/ifconfig lt0 127.0.0.42
```

The address is unimportant - however, the card needs to be configured with ifconfig so that Netatalk can find it.

The appropriate netatalk configuration depends on whether you are attached to a network that includes AppleTalk routers or not. If, like me, you are simply connecting to your home Macintoshes and printers, you need to set up netatalk to "seed". The way I do this is to have the lines:

```
dummy -seed -phase 2 -net 2000 -addr 2000.26 -zone "1033"
lt0 -seed -phase 1 -net 1033 -addr 1033.27 -zone "1033"
```

in my atalkd.conf. What is going on here is that I need to fool netatalk into thinking that there are two AppleTalk interfaces present; otherwise, it refuses to seed. This is a hack, and a more permanent solution would be to alter the netatalk code. Also, make sure you have the correct name for the dummy interface - If it's compiled as a module, you will need to refer to it as "dummy0" or some such.

If you are attached to an extended AppleTalk network, with routers on it, then you don't need to fool around with this -- the appropriate line in atalkd.conf is:

```
lt0 -phase 1
```

Card Configuration

The interrupts and so forth are configured via the dipswitch on the board. Set the switches so as not to conflict with other hardware.

Interrupts -- set at most one. If none are set, the driver uses polled mode. Because the card was developed in the XT era, the original documentation refers to IRQ2. Since you'll be running this on an AT (or later) class machine, that really means IRQ9.

SW1	IRQ 4
SW2	IRQ 3
SW3	IRQ 9 (2 in original card documentation only applies to XT)

DMA -- choose DMA 1 or 3, and set both corresponding switches.

SW4	DMA 3
SW5	DMA 1
SW6	DMA 3
SW7	DMA 1

I/O address -- choose one.

SW8	220 / 240

IP

Yes, it is possible to do IP over LocalTalk. However, you can't just treat the LocalTalk device like an ordinary Ethernet device, even

if that's what it looks like to Netatalk.

Instead, you follow the same procedure as for doing IP in EtherTalk. See Documentation/networking/ipddp.rst for more information about the kernel driver and userspace tools needed.

Bugs

IRQ autoprobing often doesn't work on a cold boot. To get around this, either compile the driver as a module, or pass the parameters for the card to the kernel as described above.

Also, as usual, autoprobing is not recommended when you use the driver as a module. (though it usually works at boot time, at least)

Polled mode is *really* slow sometimes, but this seems to depend on the configuration of the network.

It may theoretically be possible to use two LTPC cards in the same machine, but this is unsupported, so if you really want to do this, you'll probably have to hack the initialization code a bit.

Thanks

Thanks to Alan Cox for helpful discussions early on in this work, and to Denis Hainsworth for doing the bleeding-edge testing.

Bradford Johnson <bradford@math.umn.edu>

Updated 11/09/1998 by David Huggins-Daines <dhd@debian.org>