Some additional information for setting up a syncPPP
connection using network interfaces.

You need one thing beside the isdn4linux package:

  a patched pppd .. (I called it ipppd to show the difference)

Compiling isdn4linux with sync PPP:

To compile isdn4linux with the sync PPP part, you have
to answer the appropriate question when doing a "make config"
Don't forget to load the slhc.o
module before the isdn.o module, if VJ-compression support
is not compiled into your kernel. (e.g if you have no PPP or
CSLIP in the kernel)

Using isdn4linux with sync PPP:

Sync PPP is just another encapsulation for isdn4linux. The
name to enable sync PPP encapsulation is 'syncppp' .. e.g:

  /sbin/isdnctrl encap ippp0 syncppp

The name of the interface is here 'ippp0'. You need
one interface with the name 'ippp0' to saturate the
ipppd, which checks the ppp version via this interface.
Currently, all devices must have the name ipppX where
'X' is a decimal value.

To set up a PPP connection you need the ipppd .. You must start
the ipppd once after installing the modules. The ipppd
communicates with the isdn4linux link-level driver using the
/dev/ippp0 to /dev/ippp15 devices. One ipppd can handle
all devices at once. If you want to use two PPP connections
at the same time, you have to connect the ipppd to two
devices .. and so on.
I've implemented one additional option for the ipppd:
 'useifip' will get (if set to not 0.0.0.0) the IP address
 for the negotiation from the attached network-interface.
(also: ipppd will try to negotiate pointopoint IP as remote IP)
You must disable BSD-compression, this implementation can't
handle compressed packets.

Check the etc/rc.isdn.syncppp in the isdn4kernel-util package
for an example setup script.

To use the MPPP stuff, you must configure a slave device
with isdn4linux. Now call the ipppd with the '+mp' option.
To increase the number of links, you must use the
'addlink' option of the isdnctrl tool. (rc.isdn.syncppp.MPPP is
an example script)

enjoy it,
    michael