

## README for the ISDN-subsystem

### 1. Preface

#### 1.1 Introduction

This README describes how to set up and how to use the different parts of the ISDN-subsystem.

For using the ISDN-subsystem, some additional userlevel programs are necessary. Those programs and some contributed utilities are available at

`ftp.isdn4linux.de`

`/pub/isdn4linux/isdn4k-utils-<VersionNumber>.tar.gz`

We also have set up a mailing-list:

The isdn4linux-project originates in Germany, and therefore by historical reasons, the mailing-list's primary language is german. However mails written in english have been welcome all the time.

to subscribe: write a email to `majordomo@listserv.isdn4linux.de`,  
Subject irrelevant, in the message body:  
`subscribe isdn4linux <your_email_address>`

To write to the mailing-list, write to `isdn4linux@listserv.isdn4linux.de`

This mailinglist is bidirectionally gated to the newsgroup

`de.alt.comm.isdn4linux`

There is also a well maintained FAQ in English available at  
`http://www.mhessler.de/i4lfaq/`

It can be viewed online, or downloaded in sgml/text/html format.

The FAQ can also be viewed online at

`http://www.isdn4linux.de/faq/`

or downloaded from

`ftp://ftp.isdn4linux.de/pub/isdn4linux/FAQ/`

#### 1.1 Technical details

In the following Text, the terms MSN and EAZ are used.

MSN is the abbreviation for (M)ultiple(S)ubscriber(N)umber, and applies to Euro(EDSS1)-type lines. Usually it is simply the phone number.

EAZ is the abbreviation of (E)ndgeraete(A)uswahl(Z)iffer and applies to German lTR6-type lines. This is a one-digit string, simply appended to the base phone number

The internal handling is nearly identical, so replace the appropriate term to that one, which applies to your local ISDN-environment.

## README..txt

When the link-level-module isdn.o is loaded, it supports up to 16 low-level-modules with up to 64 channels. (The number 64 is arbitrarily chosen and can be configured at compile-time --ISDN\_MAX in isdn.h). A low-level-driver can register itself through an interface (which is defined in isdnif.h) and gets assigned a slot. The following char-devices are made available for each channel:

A raw-control-device with the following functions:

- write: raw D-channel-messages (format: depends on driver).
- read: raw D-channel-messages (format: depends on driver).
- ioctl: depends on driver, i.e. for the ICN-driver, the base-address of the ports and the shared memory on the card can be set and read also the boot-code and the protocol software can be loaded into the card.

O N L Y !!! for debugging (no locking against other devices):

One raw-data-device with the following functions:

- write: data to B-channel.
- read: data from B-channel.

In addition the following devices are made available:

128 tty-devices (64 cuix and 64 ttyIx) with integrated modem-emulator: The functionality is almost the same as that of a serial device (the line-discs are handled by the kernel), which lets you run SLIP, CSLIP and asynchronous PPP through the devices. We have tested Seyon, minicom, CSLIP (uri-dip) PPP, mgetty, XCEPT and Hylafax.

The modem-emulation supports the following:

### 1.3.1 Commands:

ATA	Answer incoming call.
ATD<No.>	Dial, the number may contain: [0-9] and [, #, *WPT-S] the latter are ignored until 'S'. The 'S' must precede the number, if the line is a SPV (German 1TR6).
ATE0	Echo off.
ATE1	Echo on (default).
ATH	Hang-up.
ATH1	Off hook (ignored).
ATH0	Hang-up.
ATI	Return "ISDN for Linux...".
ATI0	"
ATI1	"
ATI2	Report of last connection.
ATO	On line (data mode).
ATQ0	Enable result codes (default).
ATQ1	Disable result codes (default).
ATSx=y	Set register x to y.
ATSx?	Show contents of register x.
ATV0	Numeric responses.
ATV1	English responses (default).
ATZ	Load registers and EAZ/MSN from Profile.
AT&Bx	Set Send-Packet-size to x (max. 4000) The real packet-size may be limited by the

# README..txt

low-level-driver used. e.g. the HiSax-Module-limit is 2000. You will get NO Error-Message, if you set it to higher values, because at the time of giving this command the corresponding driver may not be selected (see "Automatic Assignment") however the size of outgoing packets will be limited correctly.

AT&D0	Ignore DTR
AT&D2	DTR-low-edge: Hang up and return to command mode (default).
AT&D3	Same as AT&D2 but also resets all registers.
AT&Ex	Set the EAZ/MSN for this channel to x.
AT&F	Reset all registers and profile to "factory-defaults"
AT&Lx	Set list of phone numbers to listen on. x is a list of wildcard patterns separated by semicolon. If this is set, it has precedence over the MSN set by AT&E.
AT&Rx	Select V.110 bitrate adaption. This command enables V.110 protocol with 9600 baud (x=9600), 19200 baud (x=19200) or 38400 baud (x=38400). A value of x=0 disables V.110 switching back to default X.75. This command sets the following Registers: Reg 14 (Layer-2 protocol): x = 0: 0 x = 9600: 7 x = 19200: 8 x = 38400: 9 Reg 18.2 = 1 Reg 19 (Additional Service Indicator): x = 0: 0 x = 9600: 197 x = 19200: 199 x = 38400: 198 Note on value in Reg 19: There is <u>NO</u> common convention for 38400 baud. The value 198 is chosen arbitrarily. Users <u>MUST</u> negotiate this value before establishing a connection.
AT&Sx	Set window-size (x = 1..8) (not yet implemented)
AT&V	Show all settings.
AT&W0	Write registers and EAZ/MSN to profile. See also iprofd (5.c in this README).
AT&X0	BTX-mode and T.70-mode off (default)
AT&X1	BTX-mode on. (S13.1=1, S13.5=0 S14=0, S16=7, S18=7,
S19=0)	
AT&X2	T.70-mode on. (S13.1=1, S13.5=1, S14=0, S16=7, S18=7,
S19=0)	
AT+Rx	Resume a suspended call with CallID x (x = 1,2,3...)
AT+Sx	Suspend a call with CallID x (x = 1,2,3...)

For voice-mode commands refer to README.audio

## 1.3.2 Escape sequence:

During a connection, the emulation reacts just like a normal modem to the escape sequence <DELAY>+++<DELAY>.

# README..txt

(The escape character - default '+' - can be set in the register 2).

The DELAY must at least be 1.5 seconds long and delay between the escape characters must not exceed 0.5 seconds.

## 1.3.3 Registers:

Nr.	Default	Description
0	0	Answer on ring number. (no auto-answer if S0=0).
1	0	Count of rings.
2	43	Escape character. (a value >= 128 disables the escape sequence).
3	13	Carriage return character (ASCII).
4	10	Line feed character (ASCII).
5	8	Backspace character (ASCII).
6	3	Delay in seconds before dialing.
7	60	Wait for carrier.
8	2	Pause time for comma (ignored)
9	6	Carrier detect time (ignored)
10	7	Carrier loss to disconnect time (ignored).
11	70	Touch tone timing (ignored).
12	69	Bit coded register: Bit 0: 0 = Suppress response messages. 1 = Show response messages. Bit 1: 0 = English response messages. 1 = Numeric response messages. Bit 2: 0 = Echo off. 1 = Echo on. Bit 3: 0 = DCD always on. 1 = DCD follows carrier. Bit 4: 0 = CTS follows RTS 1 = Ignore RTS, CTS always on. Bit 5: 0 = return to command mode on DTR low. 1 = Same as 0 but also resets all registers. See also register 13, bit 2 Bit 6: 0 = DSR always on. 1 = DSR only on if channel is available. Bit 7: 0 = Cisco-PPP-flag-hack off (default). 1 = Cisco-PPP-flag-hack on.
13	0	Bit coded register: Bit 0: 0 = Use delayed tty-send-algorithm 1 = Direct tty-send. Bit 1: 0 = T.70 protocol (Only for BTX!) off 1 = T.70 protocol (Only for BTX!) on Bit 2: 0 = Don't hangup on DTR low. 1 = Hangup on DTR low. Bit 3: 0 = Standard response messages 1 = Extended response messages Bit 4: 0 = CALLER NUMBER before every RING. 1 = CALLER NUMBER after first RING. Bit 5: 0 = T.70 extended protocol off 1 = T.70 extended protocol on Bit 6: 0 = Special RUNG Message off 1 = Special RUNG Message on

# README..txt

"RUNG" is delivered on a ttyI, if an incoming call happened (RING) and the remote party hung up before any local ATA was given.

Bit 7: 0 = Don't show display messages from net  
1 = Show display messages from net  
(S12 Bit 1 must be 0 too)

14 0

Layer-2 protocol:

0 = X75/LAPB with I-frames  
1 = X75/LAPB with UI-frames  
2 = X75/LAPB with BUI-frames  
3 = HDLC  
4 = Transparent (audio)  
7 = V.110, 9600 baud  
8 = V.110, 19200 baud  
9 = V.110, 38400 baud  
10 = Analog Modem (only if hardware

supports this)

11 = Fax G3 (only if hardware supports

this)

15 0

Layer-3 protocol:

0 = transparent  
1 = transparent with audio features (e.g.  
2 = Fax G3 Class 2 commands (S14 has to be  
3 = Fax G3 Class 1 commands (S14 has to be

DSP)

set to 11)

set to 11)

16 250

Send-Packet-size/16

17 8

Window-size (not yet implemented)

18 4

Bit coded register, Service-Octet-1 to accept, or to be used on dialout:

Bit 0: Service 1 (audio) when set.

Bit 1: Service 5 (BTX) when set.

Bit 2: Service 7 (data) when set.

Note: It is possible to set more than one bit. In this case, on incoming calls the selected services are accepted, and if the service is "audio", the Layer-2-protocol is automatically changed to 4 regardless of the setting of register 14. On outgoing calls, the most significant 1-bit is chosen to select the outgoing service octet.

19 0

Service-Octet-2

20 0

Bit coded register (readonly)

Service-Octet-1 of last call.

Bit mapping is the same as register 18

21 0

Bit coded register (readonly)

Set on incoming call (during RING) to octet 3 of calling party number IE (Numbering plan)  
See section 4.5.10 of ITU Q.931

22 0

Bit coded register (readonly)

Set on incoming call (during RING) to octet 3a of calling party number IE (Screening info)  
See section 4.5.10 of ITU Q.931

## README..txt

23    0            Bit coded register:  
Bit 0:    0 = Add CPN to RING message off  
          1 = Add CPN to RING message on  
Bit 1:    0 = Add CPN to FCON message off  
          1 = Add CPN to FCON message on  
Bit 2:    0 = Add CDN to RING/FCON message off  
          1 = Add CDN to RING/FCON message on

Last but not least a (at the moment fairly primitive) device to request the line-status (/dev/isdninfo) is made available.

Automatic assignment of devices to lines:

All inactive physical lines are listening to all EAZs for incoming calls and are NOT assigned to a specific tty or network interface. When an incoming call is detected, the driver looks first for a network interface and then for an opened tty which:

1. is configured for the same EAZ.
2. has the same protocol settings for the B-channel.
3. (only for network interfaces if the security flag is set) contains the caller number in its access list.
4. Either the channel is not bound exclusively to another Net-interface, or it is bound AND the other checks apply to exactly this interface.  
(For usage of the bind-features, refer to the isdnctrl-man-page)

Only when a matching interface or tty is found is the call accepted and the "connection" between the low-level-layer and the link-level-layer is established and kept until the end of the connection.

In all other cases no connection is established. Isdn4linux can be configured to either do NOTHING in this case (which is useful, if other, external devices with the same EAZ/MSN are connected to the bus) or to reject the call actively. (isdnctrl busreject ...)

For an outgoing call, the inactive physical lines are searched. The call is placed on the first physical line, which supports the requested protocols for the B-channel. If a net-interface, however is pre-bound to a channel, this channel is used directly.

This makes it possible to configure several network interfaces and ttys for one EAZ, if the network interfaces are set to secure operation. If an incoming call matches one network interface, it gets connected to it. If another incoming call for the same EAZ arrives, which does not match a network interface, the first tty gets a "RING" and so on.

## 2 System prerequisites:

ATTENTION!

Always use the latest module utilities. The current version is named in Documentation/Changes. Some old versions of insmod are not capable of setting the driver-Ids correctly.

## 3. Lowlevel-driver configuration.

Configuration depends on how the drivers are built. See the

README.<yourDriver> for information on driver-specific setup.

#### 4. Device-inodes

The major and minor numbers and their names are described in Documentation/devices.txt. The major numbers are:

- 43 for the ISDN-tty's.
- 44 for the ISDN-callout-tty's.
- 45 for control/info/debug devices.

#### 5. Application

a) For some card-types, firmware has to be loaded into the cards, before proceeding with device-independent setup. See README.<yourDriver> for how to do that.

b) If you only intend to use ttys, you are nearly ready now.

c) If you want to have really permanent "Modem"-settings on disk, you can start the daemon iprofd. Give it a path to a file at the command-line. It will store the profile-settings in this file every time an AT&W0 is performed on any ISDN-tty. If the file already exists, all profiles are initialized from this file. If you want to unload any of the modules, kill iprofd first.

d) For networking, continue: Create an interface:  
isdnctrl addif isdn0

e) Set the EAZ (or MSN for Euro-ISDN):  
isdnctrl eaz isdn0 2

(For lTR6 a single digit is allowed, for Euro-ISDN the number is your real MSN e.g.: Phone-Number)

f) Set the number for outgoing calls on the interface:  
isdnctrl addphone isdn0 out 1234567  
... (this can be executed more than once, all assigned numbers are tried in order)  
and the number(s) for incoming calls:  
isdnctrl addphone isdn0 in 1234567

g) Set the timeout for hang-up:  
isdnctrl huptimeout isdn0 <timeout\_in\_seconds>

h) additionally you may activate charge-hang-up (= Hang up before next charge-info, this only works, if your isdn-provider transmits the charge-info during and after the connection):  
isdnctrl chargeup isdn0 on

i) Set the dial mode of the interface:  
isdnctrl dialmode isdn0 auto  
"off" means that you (or the system) cannot make any connection (neither incoming or outgoing connections are possible). Use this if you want to be sure that no connections will be made.  
"auto" means that the interface is in auto-dial mode, and will

## README..txt

attempt to make a connection whenever a network data packet needs the interface's link. Note that this can cause unexpected dialouts, and lead to a high phone bill! Some daemons or other pc's that use this interface can cause this.

Incoming connections are also possible.

"manual" is a dial mode created to prevent the unexpected dialouts.

In this mode, the interface will never make any connections on its own. You must explicitly initiate a connection with "isdnctrl dial isdn0". However, after an idle time of no traffic as configured for the huptimeout value with isdnctrl, the connection will be ended.

If you don't want any automatic hangup, set the huptimeout value to 0.

"manual" is the default.

j) Setup the interface with ifconfig as usual, and set a route to it.

k) (optional) If you run X11 and have Tcl/Tk-wish version 4.0, you can use the script tools/tcltk/isdnmon. You can add actions for line-status changes. See the comments at the beginning of the script for how to do that. There are other tty-based tools in the tools-subdirectory contributed by Michael Knigge (imon), Volker Götz (imontty) and Andreas Kool (isdnmon).

l) For initial testing, you can set the verbose-level to 2 (default: 0). Then all incoming calls are logged, even if they are not addressed to one of the configured net-interfaces:  
isdnctrl verbose 2

Now you are ready! A ping to the set address should now result in an automatic dial-out (look at syslog kernel-messages). The phone numbers and EAZs can be assigned at any time with isdnctrl. You can add as many interfaces as you like with addif following the directions above. Of course, there may be some limitations. But we have tested as many as 20 interfaces without any problem. However, if you don't give an interface name to addif, the kernel will assign a name which starts with "eth". The number of "eth"-interfaces is limited by the kernel.

## 5. Additional options for isdnctrl:

"isdnctrl secure <InterfaceName> on"

Only incoming calls, for which the caller-id is listed in the access list of the interface are accepted. You can add caller-id's With the command "isdnctrl addphone <InterfaceName> in <caller-id>"

Euro-ISDN does not transmit the leading '0' of the caller-id for an incoming call, therefore you should configure it accordingly.

If the real number for the dialout e.g. is "09311234567" the number to configure here is "9311234567". The pattern-match function works similar to the shell mechanism.

?     one arbitrary digit  
\*     zero or arbitrary many digits  
[123] one of the digits in the list  
[1-5] one digit between '1' and '5'  
a '^' as the first character in a list inverts the list



## README..txt

"isdnctrl secure <InterfaceName> off"  
Switch off secure operation (default).

"isdnctrl ihup <InterfaceName> [on|off]"  
Switch the hang-up-timer for incoming calls on or off.

"isdnctrl eaz <InterfaceName>"  
Returns the EAZ of an interface.

"isdnctrl delphone <InterfaceName> in|out <number>"  
Deletes a number from one of the access-lists of the interface.

"isdnctrl delif <InterfaceName>"  
Removes the interface (and possible slaves) from the kernel.  
(You have to unregister it with "ifconfig <InterfaceName> down" before).

"isdnctrl callback <InterfaceName> [on|off]"  
Switches an interface to callback-mode. In this mode, an incoming call will be rejected and after this the remote-station will be called. If you test this feature by using ping, some routers will re-dial very quickly, so that the callback from isdn4linux may not be recognized. In this case use ping with the option -i <sec> to increase the interval between echo-packets.

"isdnctrl cbdelay <InterfaceName> [seconds]"  
Sets the delay (default 5 sec) between an incoming call and start of dialing when callback is enabled.

"isdnctrl cbhup <InterfaceName> [on|off]"  
This enables (default) or disables an active hangup (reject) when getting an incoming call for an interface which is configured for callback.

"isdnctrl encap <InterfaceName> <EncapType>"  
Selects the type of packet-encapsulation. The encapsulation can be changed only while an interface is down.

At the moment the following values are supported:

rawip	(Default) Selects raw-IP-encapsulation. This means, MAC-headers are stripped off.
ip	IP with type-field. Same as IP but the type-field of the MAC-header is preserved.
x25iface	X.25 interface encapsulation (first byte semantics as defined in ../networking/x25-iface.txt). Use this for running the linux X.25 network protocol stack (AF_X25 sockets) on top of isdn.
cisco-h	A special-mode for communicating with a Cisco, which is configured to do "hdlc"
ethernet	No stripping. Packets are sent with full MAC-header. The Ethernet-address of the interface is faked, from its IP-address: fc:fc:il:i2:i3:i4, where il-4 are the IP-addr.-values.
syncppp	Synchronous PPP
uihdlc	HDLC with UI-frame-header (for use with DOS ISPA, option -h1)

NOTE: x25iface encapsulation is currently experimental. Please

README..txt  
read README.x25 for further details

Watching packets, using standard-tcpdump will fail for all encapsulations except ethernet because tcpdump does not know how to handle packets without MAC-header. A patch for tcpdump is included in the utility-package mentioned above.

"isdnctrl l2\_prot <InterfaceName> <L2-ProtocolName>"

Selects a layer-2-protocol.

(With the ICN-driver and the HiSax-driver, "x75i" and "hdlc" is available. With other drivers, "x75ui", "x75bui", "x25dte", "x25dce" may be possible too. See README.x25 for x25 related l2 protocols.)

isdnctrl l3\_prot <InterfaceName> <L3-ProtocolName>

The same for layer-3. (At the moment only "trans" is allowed)

"isdnctrl list <InterfaceName>"

Shows all parameters of an interface and the charge-info.

Try "all" as the interface name.

"isdnctrl hangup <InterfaceName>"

Forces hangup of an interface.

"isdnctrl bind <InterfaceName> <DriverId>, <ChannelNumber> [exclusive]"

If you are using more than one ISDN card, it is sometimes necessary to dial out using a specific card or even preserve a specific channel for dialout of a specific net-interface. This can be done with the above command. Replace <DriverId> by whatever you assigned while loading the module. The <ChannelNumber> is counted from zero. The upper limit depends on the card used. At the moment no card supports more than 2 channels, so the upper limit is one.

"isdnctrl unbind <InterfaceName>"

unbinds a previously bound interface.

"isdnctrl busreject <DriverId> on|off"

If switched on, isdn4linux replies a REJECT to incoming calls, it cannot match to any configured interface.

If switched off, nothing happens in this case.

You normally should NOT enable this feature, if the ISDN adapter is not the only device connected to the SO-bus. Otherwise it could happen that isdn4linux rejects an incoming call, which belongs to another device on the bus.

"isdnctrl addslave <InterfaceName> <SlaveName>

Creates a slave interface for channel-bundling. Slave interfaces are not seen by the kernel, but their ISDN-part can be configured with isdnctrl as usual. (Phone numbers, EAZ/MSN, timeouts etc.) If more than two channels are to be bundled, feel free to create as many as you want. InterfaceName must be a real interface, NOT a slave. Slave interfaces start dialing, if the master interface resp. the previous slave interface has a load of more than 7000 cps. They hangup if the load goes under 7000 cps, according to their "huptimeout"-parameter.

"isdnctrl sdelay <InterfaceName> secs."

## README..txt

This sets the minimum time an Interface has to be fully loaded, until it sends a dial-request to its slave.

"isdnctrl dial <InterfaceName>"

Forces an interface to start dialing even if no packets are to be transferred.

"isdnctrl mapping <DriverId> MSN0,MSN1,MSN2,...MSN9"

This installs a mapping table for EAZ<->MSN-mapping for a single line. Missing MSN's have to be given as "-" or can be omitted, if at the end of the commandline.

With this command, it's now possible to have an interface listening to mixed lTR6- and Euro-Type lines. In this case, the interface has to be configured to a lTR6-type EAZ (one digit). The mapping is also valid for tty-emulation. Seen from the interface/tty-level the mapping CAN be used, however it's possible to use single tty's/interfaces with real MSN's (more digits) also, in which case the mapping will be ignored. Here is an example:

You have a lTR6-type line with base-nr. 1234567 and a Euro-line with MSN's 987654, 987655 and 987656. The DriverId for the Euro-line is "EURO".

```
isdnctrl mapping EURO -,987654,987655,987656,-,987655
```

```
...
isdnctrl eaz isdn0 1      # listen on 12345671(ltr6) and 987654(euro)
```

```
...
isdnctrl eaz isdn1 4      # listen on 12345674(ltr6) only.
```

```
...
isdnctrl eaz isdn2 987654 # listen on 987654(euro) only.
```

Same scheme is used with AT&E... at the tty's.

6. If you want to write a new low-level-driver, you are welcome. The interface to the link-level-module is described in the file INTERFACE. If the interface should be expanded for any reason, don't do it on your own, send me a mail containing the proposed changes and some reasoning about them. If other drivers will not be affected, I will include the changes in the next release. For developers only, there is a second mailing-list. Write to me (fritz@isdn4linux.de), if you want to join that list.

Have fun!

-Fritz