

OpenVox

深圳开源通信有限公司

OpenVox-Best Cost Effective Asterisk Cards

OpenVox B400P User Manual for mISDN



Written by: James.zhu

Email: james.zhu@openvox.cn, zhulizhong@gmail.com

Date: 15/10/2007

Version: 0.01



深圳开源通信有限公司

OpenVox-Best Cost Effective Asterisk Cards

OpenVox Communication Co. Ltd.

Address: F/2, Building No. 14, Shangsha Science & Technology Park,
No. 9283, Binhe Road, Futian District, ShenZhen, Guangdong 518048, China

Tel: +86-755-82535362, 82535095, Fax: +86-755-82535174

E-Mail: sales@openvox.com.cn

IM for Sales: betty_yjjiang@hotmail.com rubyzhangle1217@hotmail.com

IM for Technical Support: support@openvox.com.cn, zhulizhong@gmail.com, zhulizhongum@hotmail.com

Business Hours: 9:30AM-17:30PM from Monday-Friday

URL: www.openvox.com.cn

Thank You for Choosing OpenVox Products!

Table of Contents

Chapter 1	Overview.....	4
Chapter 2	Card Installation and Configuration.....	6
Chapter 3	Hardware Setting	11

Chapter 1 Overview

1. What is B200P/400P

B200P/400P series is a PCI 2.2 compliant card supporting 2/4 BRI S/T interfaces, with an onboard multi NT power feeding circuit. NT/TE mode can be independently configured on each of 2/4 ports.

B200P/400P series can be implemented for building Open Source Asterisk based systems such as ISDN PBX and VoIP gateway.

Target Applications:

High Performance ISDN PC Cards

ISDN PABX for BRI

VoIP Gateways

ISDN LAN Routers for BRI

ISDN Least Cost Routers for BRI

ISDN Test Equipment for BRI

Main Features:

Two/Four integrated S/T interfaces

ITU-T I.430 and TBR 3 certified and S/T ISDN supporting in TE and NT mode

Integrated PCI bus interface (Spec.2.2) for 3.3V and 5V signal environments

DTMF detection on all B-channels

Multiparty audio conferences bridge

Onboard power feeding

PCM bus connectors daisy chaining

Each of the 2/4 ports can be independently configured for TE or NE mode

Full software and hardware compatible with Junghanns.NET ISDN and mISDN driver

Application ready: use Asterisk to build your IP-PBX/Voicemail system

RoHS compliant

Certificates: CE, FCC

2. What is Asterisk:

The Definition of Asterisk is described as follow:

Asterisk is a complete PBX in software. It runs on Linux, BSD, Windows (emulated) and provides all of the features you would expect from a PBX and more. Asterisk does voice over IP in four protocols, and can interoperate with almost all standards-based telephony equipment using relatively inexpensive hardware.

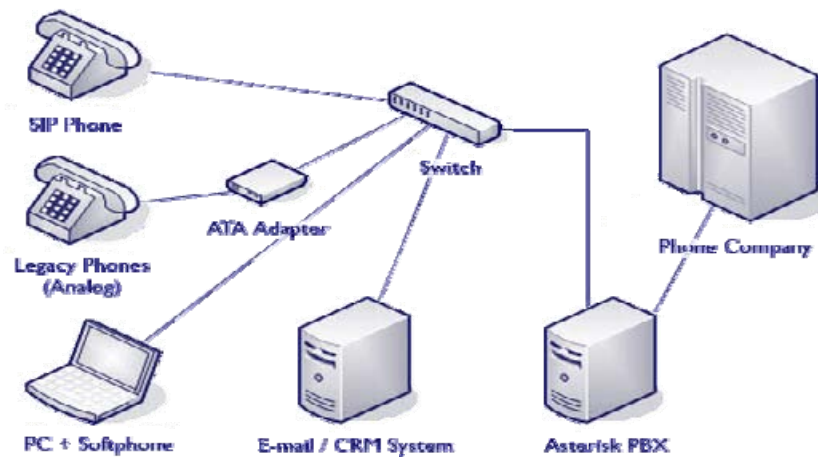


Figure 1: Asterisk Setup

Source (<http://www.siriusit.co.uk/uploads/images/consulting/asteriskSetup.gif>)

Asterisk provides Voicemail services with Directory, Call Conferencing, Interactive Voice Response, Call Queuing. It has support for three-way calling, caller ID services, ADSI, IAX, SIP, H.323 (as both client and gateway), MGCP (call manager only) and SCCP/Skinny(voip-info.org).

Chapter 2 Card Installation and Configuration

1. Hardware Installation and Setup

1) Configure the Jumper Settings

Please refer jumper setting for details. To install B400P, user should follow the steps.

A. Setting Card ID Switch

If user wants to install more than one cards of B400P in one pc, you should take care of the card id switch. It has three rules, which user must follow:

1. The card id of the first card must be set to 0, and the second card should be set to 1, and so on.
2. The First Card is a card that will be initialized (i.e installing driver) first when system is booting.
3. At most of cases, Linux will initialize PCI devices according to PCI slot order. The slot is nearest to the CPU will be initialized first; the slot at the far end from CPU will be initialized at last. That is to say, if user has more than one cards of B400P in pc, the one is nearest to CPU should be set to card id 0.

B. Adjusting Termination of S/T Interface (100 ohm)

1. If a port will work on NT mode, you should set jumper to CONNECT (ON).
2. If a port will work on TE mode, Theoretically it should be to OPEN(OFF), but user might connect to some non-standard isdn terminal equipments that do not have terminal resistors, for such equipments, you should set it to CONNECT(ON).

C. Power Feeding Connector

These jumpers control whether the card will feed power to the external isdn terminal. User should adjust accordingly.

- ✧ If the port will work on TE mode, user MUST set the jumper to OPEN(OFF)
- ✧ If this port will work on NT mode, the ISDN terminal requires ISDN power supply, user should set the jumper to CONNECT(ON).
ISDN terminal does not require ISDN power supply, user should set the jumper to OPEN(OFF).

D. Power Feeding Input

If one of the four power feeding connectors is CONNECT(ON), user should connect a D-type connector from pc power supply to this jack, the D-type connector is used to provide power to your CDROM and 3.5" HDD.

E. Power Supply Selection

Some newest model PCs do not provide +5V on the PCI slots, at those cases, use has to set the jumper to 3.3v.

F. PCM IN/PCM OUT

These are for future usages.

- 2) Power off PC, remembering unplug the AC power cable
- 3) Insert B400P into a 3.3v or 5.0v PCI slot
- 4) Plug the hard disk power supply cable(D style) to power feeding input jack if need providing power to external equipment, please refer jumper setting section for the detail
- 5) Plug back the AC power cable, and power on PC

2. Software Installation and Setup

B400P supports original mISDN driver form http://www.misdn.org/index.php/Installing_mISDN. Customers can download it from [that URL](#). There are few steps to install the driver drivers.

- 1) Checking the B400P hardware by command: **lspci -vvvvvvvvvv**

```
02:0d.0 ISDN controller: Cologne Chip Designs GmbH ISDN network Controller [HFC-4S] (rev 01)
Subsystem: Cologne Chip Designs GmbH Unknown device e888
Control: I/O+ Mem+ BusMaster- SpecCycle- MemWINV- VGASnoop- ParErr- Stepping- SERR+ FastB2B-
Status: Cap+ 66MHz- UDF- FastB2B- ParErr- DEVSEL=medium >TAbort- <TAbort- <MAbort- >SERR- <PERR-
Interrupt: pin A routed to IRQ 11
Region 0: I/O ports at dff0 [size=8]
Region 1: Memory at feafb000 (32-bit, non-prefetchable) [size=4K]
Capabilities: [40] Power Management version 2
Flags: PMEClk- DSI+ D1+ D2+ AuxCurrent=0mA PME(D0+,D1+,D2+,D3hot+,D3cold-)
Status: D0 PME-Enable- DSel=0 DScale=0 PME+
```

- 2) Checking the supporting packages

Note that if there is no kernel source in the system, user should install them. User can run **yum** again: **yum install kernel-devel**. If user uses this command **yum** will install the sources for your current version of the kernel. It is time to check for the availability of some other packages:

```
rpm -q bison
rpm -q bison-devel
rpm -q ncurses
rpm -q ncurses-devel
rpm -q zlib
rpm -q zlib-devel
rpm -q openssl
rpm -q openssl-devel
rpm -q gnutls-devel
rpm -q gcc
rpm -q gcc-c++
```

If any of those packages are not installed install them by using yum

```
yum install bison
yum install bison-devel
yum install ncurses
yum install ncurses-devel
yum install zlib
yum install zlib-devel
yum install openssl
yum install openssl-devel
yum install gnutls-devel
yum install gcc
yum install gcc-c++
```

3) Downloading, unzipping and compiling driver

- A. Download the stable version of mISDN, mISDNuser, chan_mISDN and asterisk drivers from http://www.misdn.org/index.php/Installing_mISDN, and copy the tar file to /usr/src/:

```
[root@new-host-2 src]# ls
asterisk-1.4.11      chan_misdn  linux-2.6      mISDN-1_1_2.tar.gz  mISDNuser-1_1_2.tar.gz
asterisk-1.4.11.tar.gz  kernels    mISDN-1_1_2  mISDNuser-1_1_2    redhat
[root@new-host-2 src]# cd mISDN-1_1_2
[root@new-host-2 mISDN-1_1_2]# ls
add.config  config  drivers  km_mISDN.spec  Makefile.module  misdn-init  README.misdn-init  std2kern  VERSION
CHANGES   CVS     include  Makefile       Makefile.standalone  mISDN.modprobe.d  Rules.make.ext  stddiff
```

Here, we are using mISDN-1_1_2 and mISDNuser_1_1_2. users can get more details from [mISDN.org](http://www.misdn.org).

- B. Download OpenVox patch called hfc_multi.c from www.openvox.com.cn. It contains a patched hfc_multi.c file, overwrite the original hfc_multi.c file with the new hfc_multi.c under /usr/src//usr/src/mISDN-1_1_2/drivers/isdn/hardware/mISDN:


```
[root@new-host-2 mISDN]# ls
appl.c          avmfritz.o      dsp_mec2.h      hfcsmini.o      listen.o        mISDN_12.o
appl.o          capi.c          dsp_mg2ec_const.h hfcs_usb.c      loop.c          mISDNManufacturer.h
app_plci.c      capi_enc.c      dsp_mg2ec.h     hfcs_usb.h      loop.h          mISDN_x25dte.ko
app_plci.o      capi_enc.o      dsp_tones.c     hfcsusb.ko      Makefile        mISDN_x25dte.mod.c
arcofi.c        capi.o          dsp_tones.o     hfcsusb.mod.c   Makefile.v2.4   mISDN_x25dte.mod.o
arcofi.h        channel.c       dss1.h          hfcsusb.mod.o   Makefile.v2.6   mISDN_x25dte.o
arcofi.o        channel.h       dtmf.c          hfcs_usb.o      m_capi.h        Module.symvers
asn1_address.c  channel.o       dtmf.o          hfcsusb.o       memdbg.c        ncc1.c
asn1_address.h  contr.c         fax13.c         i41_mISDN.c     memdbg.h        ncc1.o
asn1_address.o  contr.o         fsm.c           isac.c          mISDN_capi.ko   netdev.c
asn1_aoc.c      core.c          fsm.h           isac.h          mISDN_capi.mod.c netjet.c
asn1_aoc.h      core.h          fsm.o           isac.o          mISDN_capi.mod.o netjet.h
asn1_aoc.o      core.o          helper.c        isar.c          mISDN_capi.o    netjet.o
asn1_basic_service.c CVS             helper.h        isar.h          mISDN_core.ko   netjetpci.ko
asn1_basic_service.h debug.c          helper.o        isar.o          mISDN_core.mod.c netjetpci.mod.c
asn1_basic_service.o debug.h          hfc_multi.c     Kconfig.v2.6    mISDN_core.mod.o netjetpci.mod.o
asn1.c          debug.o         hfc_multi.h     l1oip.c         mISDN_core.o    netjetpci.o
asn1_comp.c     dsp_arith.h     hfcmulti.ko     l3helper.c      mISDN_dsp.ko    plci.c
asn1_comp.h     dsp_audio.c     hfcmulti.mod.c l3helper.o      mISDN_dsp.mod.c plci.o
asn1_comp.o     dsp_audio.o     hfcmulti.mod.o l3_udss1.c      mISDN_dsp.mod.o Rules.mISDN.v2.4
asn1_diversion.c dsp_biquad.h    hfc_multi.o     l3udss1.ko      mISDN_dsp.o     sed1_fax.c
asn1_diversion.h dsp_blowfish.c  hfcmulti.o      l3udss1.mod.c   mISDN_dtmf.ko   sedifax.ko
asn1_diversion.o dsp_blowfish.o  hfc_pci.c       l3udss1.mod.o   mISDN_dtmf.mod.c sedifax.mod.c
```

C. Make links with kernel source:

```
ln -s /usr/src/kernels/2.6.18-8.el5-i686/ /usr/src/linux-2.6
```

Here, under /usr/src there is kernel source, user must create link linux-2.6 under /usr/src/. There are many files under /usr/src/, please check:

```
[root@new-host-2 src]# ls
asterisk-1.4.11  chan_misdn  linux-2.6  mISDN-1_1_2.tar.gz  mISDNUser-1_1_2.tar.gz
asterisk-1.4.11.tar.gz kernels  mISDN-1_1_2  mISDNUser-1_1_2  redhat
[root@new-host-2 src]# ls
asterisk-1.4.11  chan_misdn  linux-2.6  mISDN-1_1_2.tar.gz  mISDNUser-1_1_2.tar.gz
asterisk-1.4.11.tar.gz kernels  mISDN-1_1_2  mISDNUser-1_1_2  redhat
[root@new-host-2 src]# ls
asterisk-1.4.11  chan_misdn  linux-2.6  mISDN-1_1_2.tar.gz  mISDNUser-1_1_2.tar.gz
asterisk-1.4.11.tar.gz kernels  mISDN-1_1_2  mISDNUser-1_1_2  redhat
[root@new-host-2 src]# uname -a
Linux new-host-2 2.6.18-8.el5 #1 SMP Thu Mar 15 19:57:35 EDT 2007 i686 i686 i386 GNU/Linux
```

D. Compiling mISDN, mISDNUser, chan_mISDN and asterisk

```
cd /usr/src/mISDN-1_1_2
```

```
make
```

```
make install
```

```
std2kern
```

```
cd /usr/src/mISDN-1_1_2
```

```
make
```

```
make install
```

```
cd /usr/src/chan_misdn
make
make install
```

```
cd /usr/src/asterisk-1.4.11
./configure
make menuselect
```

now you should enable chan_misdn in the Channel Driver Section and reinstall asterisk with "make install".

After that procedure you should have the current mISDN releases installed and the current chan_misdn with asterisk.

E. Modifying and loading modules for mISDN

vi /etc/mISDN.conf, and edit the mISDN.conf like this:

```
<?xml version="1.0"?>
<!--
-----
Card Type: BN2SO, BN4SO, BN8SO
Card Attributes: ulaw=(yes|no), dtmf=(yes|no), pcm_slave=(yes|no),
                  ignore_pcm_frameclock=(yes|no), rxclock=(yes|no),
                  crystalclock=(yes|no), watchdog=(yes|no)
Port Attributes: mode=(te|nt), link=(ptp|ptmp), master-clock=(yes|no),
                  capi=(yes|no)
-----
Card Type: BN2E1
Card Attributes: ulaw=(yes|no), dtmf=(yes|no), pcm_slave=(yes|no),
                  ignore_pcm_frameclock=(yes|no), rxclock=(yes|no),
                  crystalclock=(yes|no), watchdog=(yes|no)
Port Attributes: mode=(te|nt), link=(ptp|ptmp), optical=(yes|no), los=(yes|no),
                  ais=(yes|no), slip=(yes|no), nocrc4=(yes|no), capi=(yes|no)
-----
Card Type: hfcmulti, avmfritz, w6692pci
Port Attributes: mode=(te|nt), link=(ptp|ptmp), capi=(yes|no)
-----
Module: hfcmulti
Options: poll=<number>, pcm=<number>, debug=<number>, timer=(yes|no)
-----
Module: mISDN_dsp
Options: debug=<number>, options=<number>, poll=<number>,
        dtmfthreshold=<number>
-----
-->
<mISDNconf>
  <module poll="128" debug="0" timer="no">hfcmulti</module>
  <module debug="0" options="0">mISDN_dsp</module>
  <devnode user="root" group="root" mode="644">mISDN</devnode>
  <card type="BN4SO">
    <port mode="te" link="ptp">1</port>
    <port mode="te" link="ptp">2</port>
    <port mode="te" link="ptp">3</port>
    <port mode="te" link="ptp">4</port>
  </card>
</mISDNconf>
```

vi /etc/asterisk/misdn.conf, the part of it looks like this:

```
;;[Intern]
;ports=3,4
;context=Intern
[outcall]
ports=1
context=from-isdn
[isdn]
; define your ports, e.g. 1,2 (depends on mISDN-driver loading order)
ports=2,3,4
; context where to go to when incoming Call on one of the above ports
context=demo

;[internPP]
;
; adding the postfix 'ptp' to a port number is obsolete now, chan_misdn
; parses /etc/misdn-init.conf and sets the ptp mode to the corresponding
; configs. For backwards compatibility you can still set ptp here.
;
;ports=3
```

Vi /etc/asterisk/extensions.conf, the dialplan shows as the follow:

```
[from-isdn]
exten => s,1,Answer()
exten => s,2,NoOp("coming calls")
exten => s,3,Dial(SIP/600)
[demo]
;
; We start with what to do when a call first comes in.
;
exten => 100,1,Dial(mISDN/g:isdn/13570807100)
exten => 100,2,Hangup()
```

The dialplan is linked with misdn.conf. The example show that the port 1 is used for inbound calls, and port 2,3 and 4 are used for outbound calls

F. Running mISDN and asterisk:

Execute: mISDN start

Execute: asterisk -vvvvvvvgc

Loading the chan_misdn.so

The above screens shows the asterisk makes outbound call, inbounds and loads chan_misdn modules:

```

root@new-host-2 src# asterisk -vvvvvvvvv
Asterisk 1.4.11, Copyright (C) 1999 - 2007 Digium, Inc. and others.
Created by Mark Spencer <markster@digium.com>
Asterisk comes with ABSOLUTELY NO WARRANTY; type 'core show warranty' for details.
This is free software, with components licensed under the GNU General Public
license version 2 and other licenses; you are welcome to redistribute it under
certain conditions. Type 'core show license' for details.
=====
== Parsing '/etc/asterisk/asterisk.conf': Found
== Parsing '/etc/asterisk/extconfig.conf': Found
Connected to Asterisk 1.4.11 currently running on new-host-2 (pid = 2235)
Verbosity is 3 and is now 8
-- Remote UNIX connection
new-host-2*CLI> load chan_misdn.so
The 'load' command is deprecated and will be removed in a future release. Please use 'module load' instead.
== Parsing '/etc/asterisk/misdn.conf': Found
[Oct 16 10:34:17] WARNING[3093]: misdn_config.c:921 _build_general_config: misdn.conf: "watcher_timeout=1" (section: general) is out of range. Please edit your misdn.conf and then do a "misdn reload".
[Oct 16 10:34:17] WARNING[3093]: misdn_config.c:977 _build_port_config: misdn.conf: "echocancelwhenbridged=no" (section: port) is out of range. Please edit your misdn.conf and then do a "misdn reload".
[Oct 16 10:34:17] WARNING[3093]: misdn_config.c:977 _build_port_config: misdn.conf: "echotraining=no" (section: default) is out of range. Please edit your misdn.conf and then do a "misdn reload".
P[ 0] Got: 1,2,3,4 from get_ports
== Registered channel type 'mISDN' (Channel driver for mISDN Support (Bri/Pri))
== Registered application 'misdn_set_opt'
== Registered application 'misdn_facility'
== Registered application 'misdn_check_1211'
P[ 0] -- mISDN Channel Driver Registered --
Loaded chan_misdn.so => (Channel driver for mISDN Support (BRI/PRI))

```

```

Verbosity is at least 8
-- Remote UNIX connection
-- Registered SIP '600' at 192.168.2.155 port 62174 expires 3600
-- Saved useragent "X-Lite release 1011s stamp 41150" for peer 600
[Oct 16 10:48:14] NOTICE[2333]: chan_sip.c:12331 handle_response_peerpoke: Peer '600' is now Reachable. (2ms / 2000ms)
new-host-2*CLI> !
root@new-host-2 asterisk# vi extensions.conf
root@new-host-2 asterisk# asterisk -r
Asterisk 1.4.11, Copyright (C) 1999 - 2007 Digium, Inc. and others.
Created by Mark Spencer <markster@digium.com>
Asterisk comes with ABSOLUTELY NO WARRANTY; type 'core show warranty' for details.
This is free software, with components licensed under the GNU General Public
license version 2 and other licenses; you are welcome to redistribute it under
certain conditions. Type 'core show license' for details.
=====
Connected to Asterisk 1.4.11 currently running on new-host-2 (pid = 2235)
Verbosity is at least 8
-- Executing [1000@demo:1] Dial("SIP/600-08d5eef8", "mISDN/g:isdn/1357080") in new stack
-- Called g:isdn/1357080 : ''
-- mISDN/7-u3 is proceeding passing it to SIP/600-08d5eef8
-- mISDN/7-u4 is ringing
-- mISDN/7-u4 answered SIP/600-08d5eef8
-- mISDN/7-u4 answered SIP/600-08d5eef8

```

```

-- codec gsm: using generic PLC
-- Reloading module 'cdr_manager.so' (Asterisk Manager Interface CDR Backend)
== Parsing '/etc/asterisk/sip_notify.conf': Found
== MGCP Listening on 0.0.0.0:2727
== Using TOS bits 0
== Starting mISDN/2-u25 at from-isdn,,1 failed so falling back to exten 's'
-- Executing [s@from-isdn:1] Answer("mISDN/2-u25", "") in new stack
-- Executing [s@from-isdn:2] NoOp("mISDN/2-u25", "coming calls") in new stack
-- Executing [s@from-isdn:3] Dial("mISDN/2-u25", "SIP/600") in new stack
-- Called 600
-- SIP/600-08d81608 is ringing
-- SIP/600-08d81608 answered mISDN/2-u25
== Spawn extension (from-isdn, s, 3) exited non-zero on 'mISDN/2-u25'

```

```

new-host-2*CLI> misdn show stacks
BEGIN STACK LIST:
* Port 1 Type TE Prot. PTP L2Link DOWN L1Link:DOWN Blocked:0 Debug:0
* Port 2 Type TE Prot. PTP L2Link DOWN L1Link:DOWN Blocked:0 Debug:0
* Port 3 Type TE Prot. PTP L2Link DOWN L1Link:DOWN Blocked:0 Debug:0
* Port 4 Type TE Prot. PTP L2Link UP L1Link:UP Blocked:0 Debug:0

```

Notes:

Test environments:

OS: Centos 5

Kernel version: 2.6.18-8.15

Asterisk version: Asterisk-1.4.11

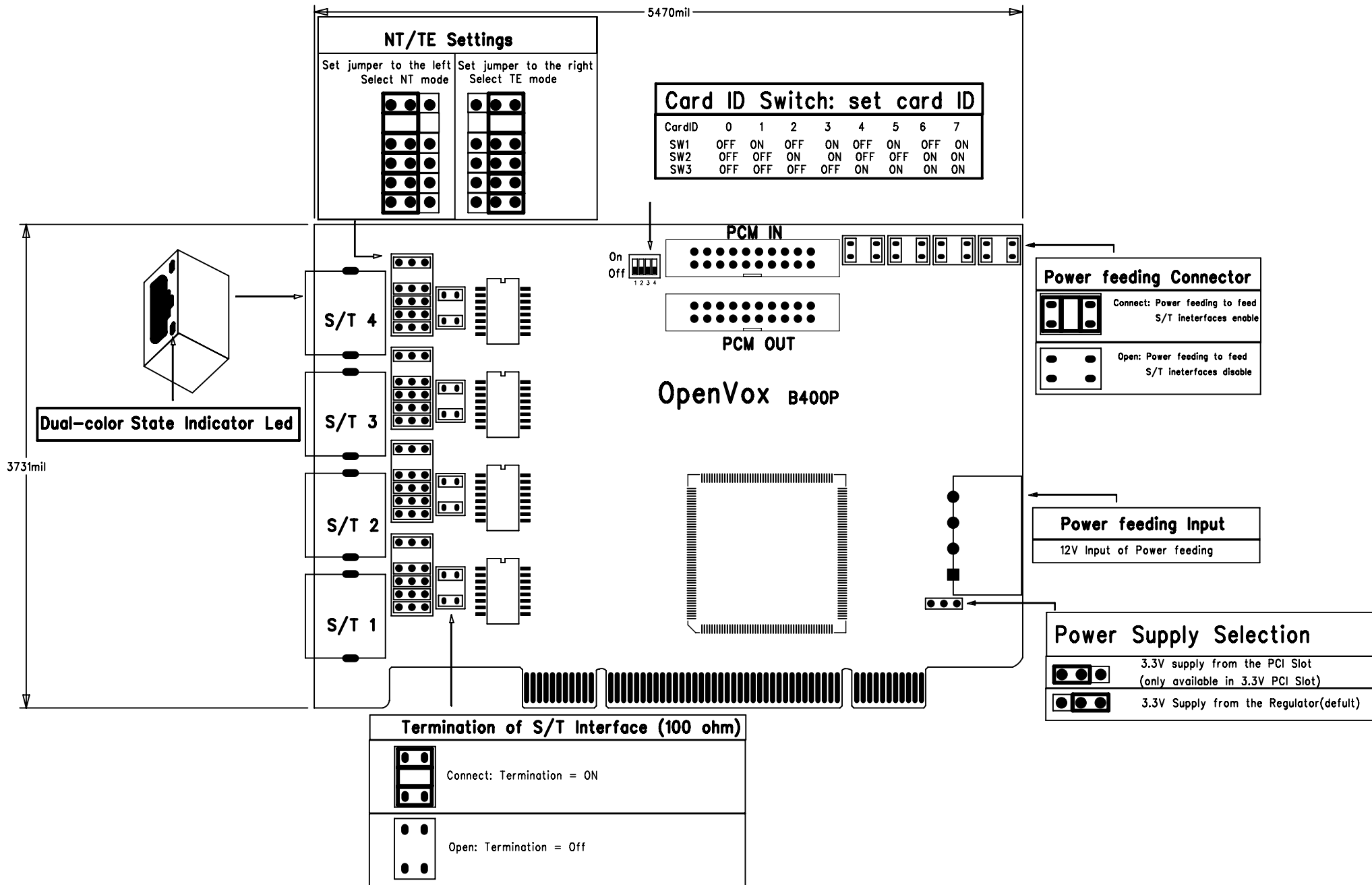
mISDN version: mISDN-1_1_2

Hardware: OpenVox B400P

Some Issues:

- 1) LEDs will turn into red and blink if the drivers are loaded.
- 2) Misdn show channels commands show nothing.
- 3) When calls coming, the LED will be turned into green status for very short while
- 4) If ISDN plugs into the port, the LED will not blink, but in red color.

Chapter 3 Hardware Setting



OpenVox B400P Jumper Setting