



深圳开源通信有限公司

Open Vox-Best Cost Effective Asterisk Cards

OpenVox A1610E/A1610P/AE1610E/AE1610P User Manual



A1610E/A1610P

Date:05/26/2011

Version: 1.3



深圳开源通信有限公司

Open Vox-Best Cost Effective Asterisk Cards

OpenVox Communication Co. Ltd.

Address: F/3, Block No.127, Jindi Industrial Zone,
Shazui Road, Futian district, ShenZhen, Guangdong 518048, China

Tel:+86-755-82535095,82535461,Fax:+86-755-82535174

E-Mail: sales@openvox.cn support@openvox.cn

M for Technical Support: support@openvox.cn

Business Hours: 9:00AM-18:00PM from Monday-Friday

URL: www.openvox.cn

Thank You for Choosing OpenVox Products!

Table of Contents

1. Overview.....	4
1.1 What is A1610P/A1610E.....	4
1.2 What is Asterisk:.....	5
2. Hardware Setting.....	6
3. Software Installation and Configuration.....	8
3.1 Download asterisk, dahdi, driver and firmware.....	8
3.2 Modify the configure files.....	8
3.3 Installation.....	10
3.4 Configure.....	12
4. References.....	21

1. Overview

1.1 What is A1610P/A1610E

A1610P/A1610E is a modular analog telephony interface product. AE1610P/AE1610E is A1610P/A1610E with EC module. It is designed to build IP PBX.

A1610P/AE1610P or A1610E/AE1610E must be used with FXO-400 or FXS-400 together to build a workable system. AE1610P/AE1610E has EC module, which must be used with FXO-400 or FXS-400 together.

Key Benefits:

Low CPU Payload : Firmware accelerate I/O access achieve high stability and highly decreased cpu payload

Scalable: Just add additional cards to extend system

Bus Master: Operation speed up to 132Mbytes/sec

Echo cancellation: Support high quality octasic echo cancellation DSP, each channel independent of 128ms or 1024 taps echo cancellation

RoHS compliant

Certificates: CE, FCC

Misc:

Temperature Operation: 0 to 50°C

Temperature Storage: -40 to 125°C

Humidity: 10 TO 90% NON-CONDENSING

Disclaimers

Asterisk® is a registered trademark of Digium, Inc.

1.2 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.

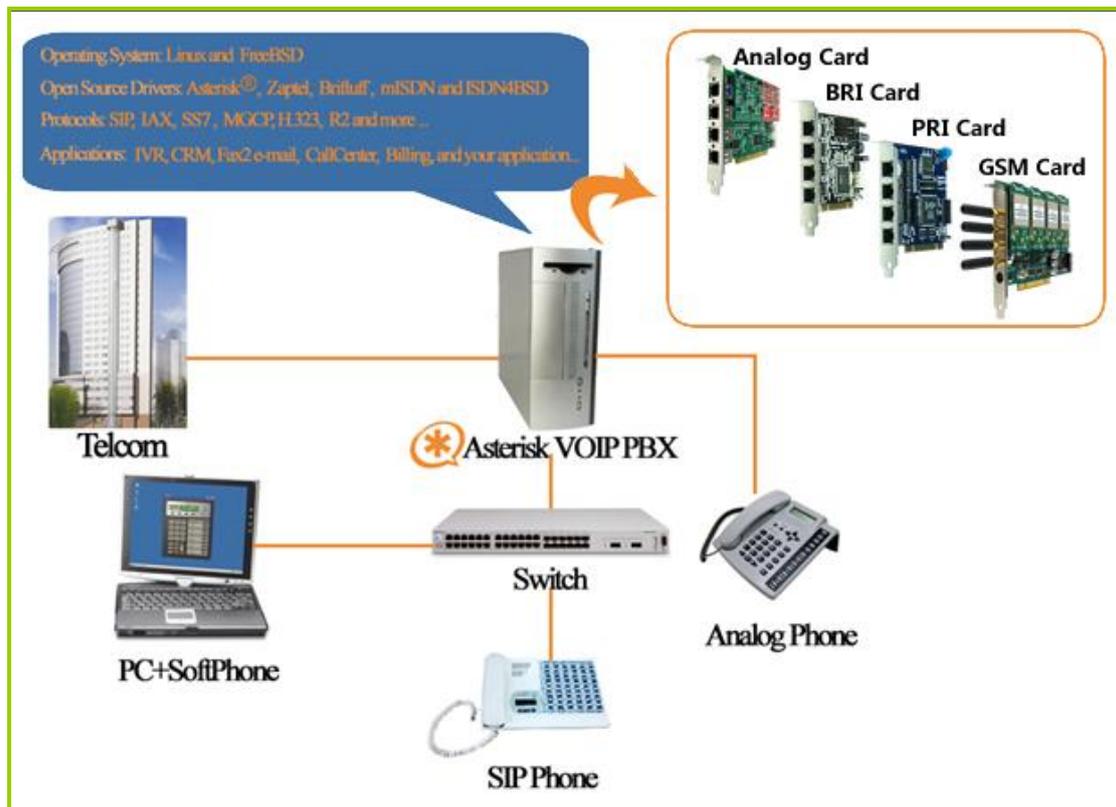


Figure 1: Asterisk_OpenVox Setup

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).

2. Hardware Setting

To set the A1610P/AE1610P, user MUST go through these steps:

1. Checking power supply: **Board must be provided power, please plug the power supply cable into power supply connector. Otherwise, you will not be able to use.** (refer figure 2 below)

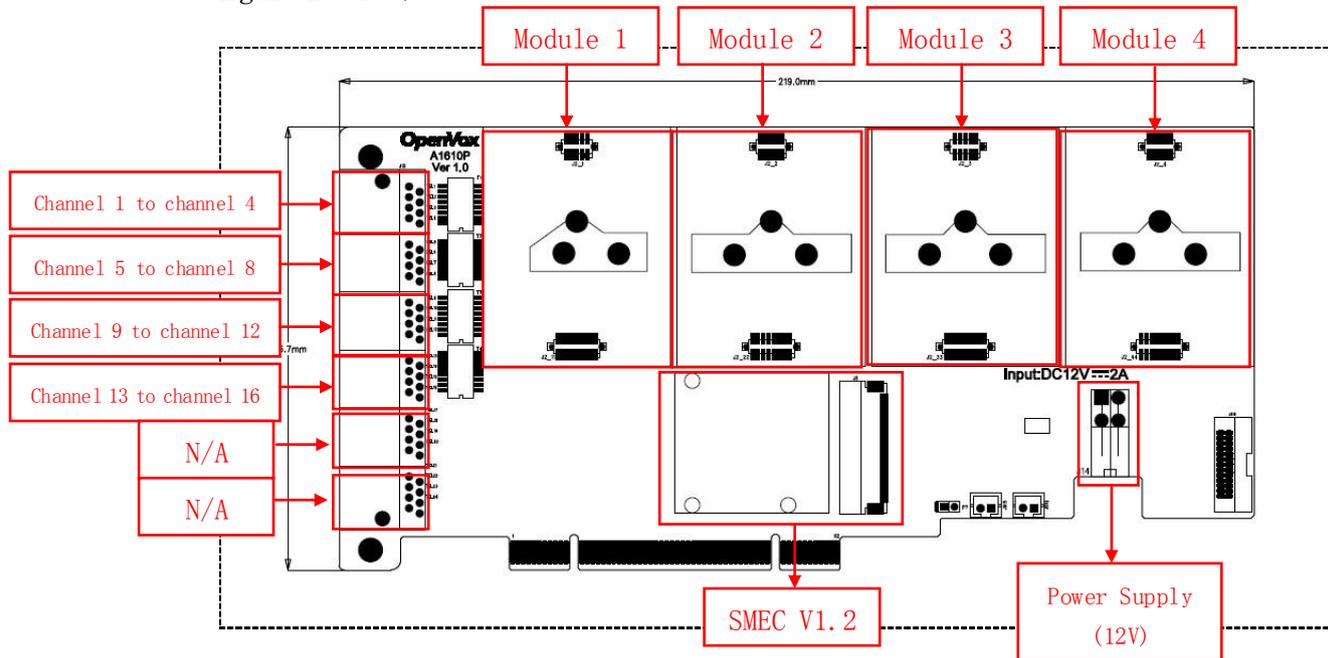


Figure 2: A1610P/AE1610P Hardware Configuration

2. Checking cabling/pin assignment: There are two modules on A1610P/AE1610P, each module (FXS400/FXO400) corresponds one RJ45 interface (refer figure 2). There are eight pins on each RJ45 interface. A1610P/AE1610P uses the two pins of it as a pair, connecting to two-wire telephone line, **so each RJ45 interface can split into four telephone lines.** Please see figure 3 for the setting of A1610P/AE1610P.

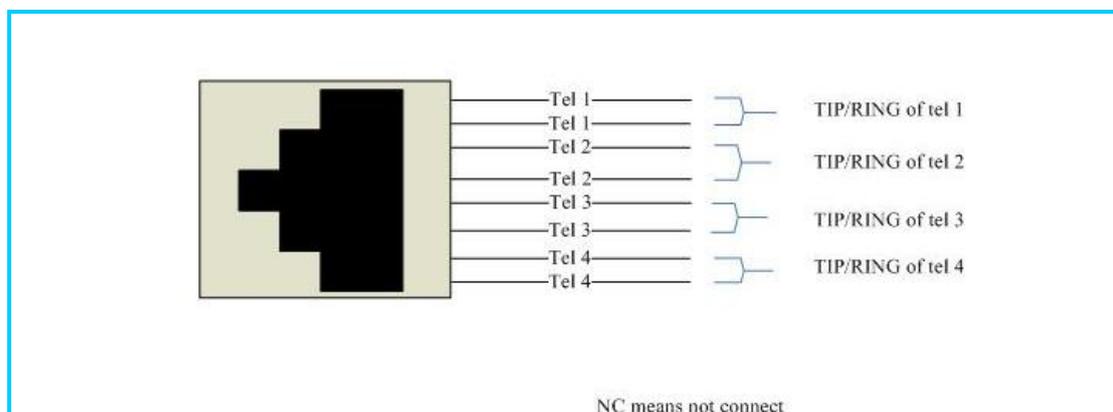


Figure 3: A1610P/AE1610P pin assignment

3. Splitting the RJ45 to RJ11: User has to use a splitter (refer figure 4) to split RJ45 interface into four RJ11 normal telephone lines. **And please connect PSTN line into FXO port and telephone into FXS port.**

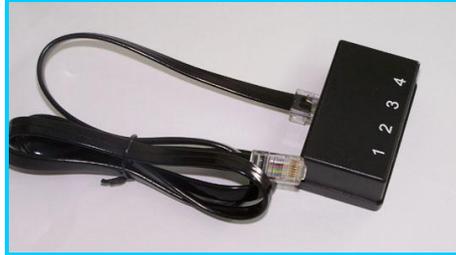


Figure 4: A1610P/AE1610P Splitter

3. Software Installation and Configuration

A1610P/AE1610P supports dahdi software device driver on Linux. To use A1610P/AE1610P, user must download, edit, install and configure dahdi and asterisk.

3.1 Download asterisk, dahdi, driver and firmware

1. Download asterisk from www.asterisk.org

Right here, take asterisk-1.8.0 for an example:

<http://downloads.asterisk.org/pub/telephony/asterisk/releases/asterisk-1.8.0.tar.gz>

2. There are two methods to download dahdi source.

- 1) Download dahdi from openvox website, which has included driver of A1610P/AE1610P, user doesn't need to modify any files.

http://downloads.openvox.cn/pub/drivers/dahdi-linux-complete/openvox_dahdi-linux-complete-current.tar.gz (if you select this method, just skip to step 3.3 after you finish this step)

- 2) Download dahdi from asterisk official website: downloads.asterisk.org, then user needs to download the driver patch of A1610/AE1610 as well. The patch is located here:

http://downloads.openvox.cn/pub/drivers/dahdi-patches/a2410p/opvxa24xx_dahdi-linux.tar.gz

There is a directory named opvxa24xx after you untar this package, please copy the whole directory to `/usr/src/dahdi-linux-xxx/drivers/dahdi/`. And you have to modify some configure files before you start compiling dahdi.

3.2 Modify the configure files

There are some steps here, make sure you have already modified them before you compile dahdi.

- A. Forward to directory `/usr/src/dahdi-linux-xxx/build_tools`, and modify file `live_dahdi` like this:

```

91 #FIXME better automation of the voicebus
92 # dependency:
93 MODULES_LOAD= $MODULES_LOAD voicebus/dahdi_voicebus $mod/$mod
94 EXTRA_MODS= $EXTRA_MODS firmware_class
95
96 wct4xxp | wctel2xp | wctc4xp | wcb4xxp opvxa24xx
97 MODULES_LOAD= $MODULES_LOAD $mod/$mod

```

Figure 5 live_dahdi

- B. Forward to directory `/usr/src/dahdi-linux-xxx/drivers/dahdi`, and modify file `Kbuild` like this:

```

9 obj-$(DAHDI_BUILD_ALL)$(CONFIG_DAHDI_WCT4XXP) += wct4xxp/
10 obj-$(DAHDI_BUILD_ALL)$(CONFIG_DAHDI_WCTC4XXP) += wctc4xxp/
11 obj-$(DAHDI_BUILD_ALL)$(CONFIG_DAHDI_WCTDM24XXP) += wctdm24xxp/
12 obj-$(DAHDI_BUILD_ALL)$(CONFIG_DAHDI_OPVXA24XX) += opvxa24xx/

```

Figure 6 Kbuild

C. Forward to directory /usr/src/dahdi-linux-xxx/drivers/dahdi, and insert the following info in the red circle into the file Kconfig near line 156.

```

156 config DAHDI_OPVXA24XX
157     tristate "OpenVox 24 ports analog card Support"
158     depends on DAHDI && PCI
159     default DAHDI
160     ---help---
161     This driver provides support for the following OpenVox
162     Wildcard products:
163     * A2410P (PCI)
164     * A1610P (PCI)
165     * A810P (PCI)
166     To compile this driver as a module, choose M here: the
167     module will be called opvxa24xx.
168     If unsure, say Y.
169
170 config DAHDI_VOICEBUS

```

Figure 7 Kconfig

D. Forward to directory /usr/src/dahdi-tools-xxx/xpp/perl_modules/Dahdi/Hardware, and add the following info to the file PCI.pm.

```

16 # from opvxa24xx
17 '1b74:1610' => { DRIVER => 'opvxa24xx', DESCRIPTION => 'OpenVox A1610P' },
18
19 # from pciradio

```

Figure 8 PCI.pm

E. Forward to directory /usr/src/dahdi-tools-xxx/xpp/perl_modules/Dahid, and modify file Chans.pm like this:

```

195 sub probe_type($) {
196     my $self = shift;
197     my $fqm = $self->fqm;
198     my $num = $self->num;
199     my $type;
200
201     if ($fqm =~ m:WCTDM/|WRTDM/|OPVXA1200/|OPVXA24XX/;) {
202         my $maybe;
203

```

Figure 9 Chans.pm

F. Forward to directory `/usr/src/dahdi-tools-xxx`, and add the following info to file `modules.sample`.

```
15 wct4xxp
16 # OpenVox A1610: up to 16 analog ports
17 opvxa24xx
```

Figure 9 `modules.sample`

G. Forward to directory `/usr/src/dahdi-tools-xxx`, and modify `blacklist.sample` like this:

```
12 blacklist opvxa24xx
13 blacklist wctc4xxp
14 blacklist wcb4xxp
```

Figure 9 `blacklist.sample`

Note: If you select CentOS 5.6 you have to patch dahdi, please see [here](#) for the patch.

3.3 Installation

Before installing dahdi and asterisk, please make sure that some supporting packages have been installed.

Note that if there is no kernel source in the system, user should install it. User can run `yum` again: *`yum install kernel-devel`*.

It is time to check for the availability of some supporting 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 // make sure the version is higher than 4.0`

`rpm -q gcc-c++`

`rpm -q libxml2`

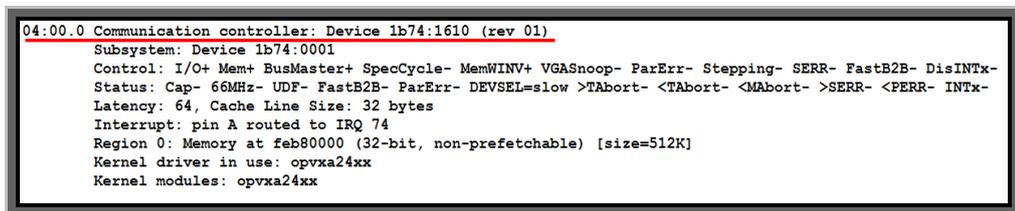
If any of those packages are not installed, please install them by using `yum`.

```
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++  
yum install libxml2
```

User can install the driver via the following steps (assuming user has the source code of dahdi device driver installed in /usr/src/dahdi-2.2.XX directory):

1. Checking the A1610P/AE1610P hardware by command: *lspci -vvvvv*

From the following, user can see that there is a device called communication controller interface be found.



```
04:00.0 Communication controller: Device 1b74:1610 (rev 01)  
Subsystem: Device 1b74:0001  
Control: I/O+ Mem+ BusMaster+ SpecCycle- MemWINV+ VGASnoop- ParErr- Stepping- SERR- FastB2B- DisINTx-  
Status: Cap- 66MHz- UDF- FastB2B- ParErr- DEVSEL=slow >TAbort- <TAbort- <MAbort- >SERR- <PERR- INTx-  
Latency: 64, Cache Line Size: 32 bytes  
Interrupt: pin A routed to IRQ 74  
Region 0: Memory at feb80000 (32-bit, non-prefetchable) [size=512K]  
Kernel driver in use: opvxa24xx  
Kernel modules: opvxa24xx
```

Figure 10: hardware detect

2. Compiling and installing dahdi

- 1) a. If user uses dahdi-linux-complete-xxx.tar.gz, execute these commands:

```
cd /usr/src/  
tar -xvzf dahdi-linux-complete-xxx.tar.gz  
cd dahdi-linux-complete-xxx  
make  
make install  
make config
```

- b. If user uses dahdi-linux-xxx.tar.gz and dahdi-tools-xxx.tar.gz, execute these commands: (make sure you have already finished step 3.2 before you carry out the following commands).

```
cd /usr/src  
tar -xvzf dahdi-linux-xxx.tar.gz  
cd dahdi-linux-xxx,  
make  
make install
```

```
cd /usr/src/  
tar -xzvf dahdi-tools-xxx.tar.gz  
cd dahdi-tools-xxx,  
./configure  
make  
make install  
make config
```

2) Installing asterisk

```
cd /usr/src/  
tar -xzvf asterisk-xxx.tar.gz  
cd asterisk-xxx  
configure  
make  
make install  
make samples
```

3.4 Configure

1. Loading modules for opvxa24xx:

```
modprobe dahdi  
modprobe opvxa24xx opermode=YOUR_COUNTRY //the driver name is the same  
as //opvxa24xx
```

openvox_dahdi-linux-complete 2.2.0 or higher versions allows user to adjust how long to initiate once IRQ. User is able to adjust time to initiate IRQ by the following way:

```
modprobe opvxa24xx opermode=YOUR_COUNTRY ms_per_irq=2
```

ms_per_irq=2 means every 2 millisecond initiate once IRQ.

The valid values of ms_per_irq are 1,2,4,8,16, the default value is 1.

dahdi-linux-complete-2.4.0 or higher version supports this function.

Execute **dmesg** command to check if you have made the EC module worked.

If user uses AE1610P/AE1610E, from figure 14 below, user will be able to see EC module has been detected by the system.

```
dahdi: Telephony Interface Registered on major 196
dahdi: Version: 2.4.0
OpenVox A1610P version: 1.3
Module 0: Installed -- AUTO FXO (FCC mode)
Module 1: Installed -- AUTO FXO (FCC mode)
Module 2: Installed -- AUTO FXO (FCC mode)
Module 3: Installed -- AUTO FXO (FCC mode)
Module 4: Installed -- AUTO FXO (FCC mode)
Module 5: Installed -- AUTO FXO (FCC mode)
Module 6: Installed -- AUTO FXO (FCC mode)
Module 7: Installed -- AUTO FXO (FCC mode)
Module 8: Installed -- AUTO FXS/DPO
Module 9: Installed -- AUTO FXS/DPO
Module 10: Installed -- AUTO FXS/DPO
Module 11: Installed -- AUTO FXS/DPO
Module 12: Installed -- AUTO FXS/DPO
Module 13: Installed -- AUTO FXS/DPO
Module 14: Installed -- AUTO FXS/DPO
Module 15: Installed -- AUTO FXS/DPO
OpenVox VPM: echo cancellation supports 32 channels
OpenVox VPM: echo cancellation for 32 channels
OpenVox VPM: hardware DTMF disabled.
OpenVox VPM: Present and operational servicing 1 span(s)
Found an OpenVox A1610P: Version 1.3 (16 modules)
```

Figure 11: EC detection

2. Checking the configure files

Run the command `vi /etc/dahdi/genconf_parameters` to disable the `softecho` parameter.

If users use AE1610P, please set `echo_can` to `none` as following:

```
echo_can          none
```

If users use A1610P, just ignore that step above.

Then run these commands:

```
dahdi_genconf
dahdi_cfg -vvvv
```

The output might be the same as the following:

```
[root@localhost ~]# dahdi_cfg -vvvv
DAHDI Tools Version - 2.4.0

DAHDI Version: 2.4.0
Echo Canceller(s): MG2
Configuration
=====

Channel map:

Channel 01: FXS Kewlstart (Default) (Echo Canceler: mg2) (Slaves: 01)
Channel 02: FXS Kewlstart (Default) (Echo Canceler: mg2) (Slaves: 02)
Channel 03: FXS Kewlstart (Default) (Echo Canceler: mg2) (Slaves: 03)
Channel 04: FXS Kewlstart (Default) (Echo Canceler: mg2) (Slaves: 04)
Channel 05: FXS Kewlstart (Default) (Echo Canceler: mg2) (Slaves: 05)
Channel 06: FXS Kewlstart (Default) (Echo Canceler: mg2) (Slaves: 06)
Channel 07: FXS Kewlstart (Default) (Echo Canceler: mg2) (Slaves: 07)
Channel 08: FXS Kewlstart (Default) (Echo Canceler: mg2) (Slaves: 08)
Channel 09: FXO Kewlstart (Default) (Echo Canceler: mg2) (Slaves: 09)
Channel 10: FXO Kewlstart (Default) (Echo Canceler: mg2) (Slaves: 10)
Channel 11: FXO Kewlstart (Default) (Echo Canceler: mg2) (Slaves: 11)
Channel 12: FXO Kewlstart (Default) (Echo Canceler: mg2) (Slaves: 12)
Channel 13: FXO Kewlstart (Default) (Echo Canceler: mg2) (Slaves: 13)
Channel 14: FXO Kewlstart (Default) (Echo Canceler: mg2) (Slaves: 14)
Channel 15: FXO Kewlstart (Default) (Echo Canceler: mg2) (Slaves: 15)
Channel 16: FXO Kewlstart (Default) (Echo Canceler: mg2) (Slaves: 16)

16 channels to configure.

Setting echocan for channel 1 to mg2
Setting echocan for channel 2 to mg2
Setting echocan for channel 3 to mg2
Setting echocan for channel 4 to mg2
Setting echocan for channel 5 to mg2
Setting echocan for channel 6 to mg2
Setting echocan for channel 7 to mg2
Setting echocan for channel 8 to mg2
Setting echocan for channel 9 to mg2
Setting echocan for channel 10 to mg2
Setting echocan for channel 11 to mg2
Setting echocan for channel 12 to mg2
Setting echocan for channel 13 to mg2
Setting echocan for channel 14 to mg2
Setting echocan for channel 15 to mg2
Setting echocan for channel 16 to mg2
```

Figure 12: dahdi_cfg -vvvv screenshot

The command **dahdi_genconf** will automatically generate the system.conf under directory /etc/dahdi and dahdi-channels.conf under /etc/asterisk. Please check the setting of system.conf, it looks like the following:

```
# Autogenerated by /usr/sbin/dahdi_genconf on Fri May 27 15:52:47 2011
# If you edit this file and execute /usr/sbin/dahdi_genconf again,
# your manual changes will be LOST.
# Dahdi Configuration File
#
# This file is parsed by the Dahdi Configurator, dahdi_cfg
#
# Span 1: OPVXA24XX/24 "OpenVox A1610 Board 25" (MASTER)
fxsks=1
echocanceller=mg2,1
fxsks=2
echocanceller=mg2,2
fxsks=3
echocanceller=mg2,3
fxsks=4
echocanceller=mg2,4
fxsks=5
echocanceller=mg2,5
fxsks=6
echocanceller=mg2,6
fxsks=7
echocanceller=mg2,7
fxsks=8
echocanceller=mg2,8
fxoks=9
echocanceller=mg2,9
fxoks=10
echocanceller=mg2,10
fxoks=11
echocanceller=mg2,11
fxoks=12
echocanceller=mg2,12
fxoks=13
echocanceller=mg2,13
fxoks=14
echocanceller=mg2,14
fxoks=15
echocanceller=mg2,15
fxoks=16
echocanceller=mg2,16

# Global data

loadzone      = us
defaultzone   = us
```

Figure 13: system.conf

In order to match your country pattern, you will need to change the parameters **load zone** and **default zone** to your country. For example, your system is in **CHINA**, then, you will change their value like this:

```
loadzone      = cn
defaultzone   = cn
```

Meanwhile, you also need to modify another parameter: **country** in file `/etc/asterisk/indications.conf`

```
[general]
country=us           ; default location
```

Figure 14: indications.conf

After loading dahdi and opvxa24xx driver, user should check the chan_dahdi.conf and dahdi-channels.conf under /etc/asterisk. Please make sure dahdi-channels.conf has been included into chan_dahdi.conf. If not, run the command:

```
echo "#include dahdi-channels.conf" >> /etc/asterisk/chan_dahdi.conf
```

Make sure that the context in dahdi-channels.conf should exist in extensions.conf. File dahdi-channels.conf should look like as the following:

FXO ports use FXS signaling, the configure file looks like this:

```
Autogenerated by /usr/sbin/dahdi_genconf on Fri May 27 15:52:47 2011
; If you edit this file and execute /usr/sbin/dahdi_genconf again,
; your manual changes will be LOST.
; Dahdi Channels Configurations (chan_dahdi.conf)
;
; This is not intended to be a complete chan_dahdi.conf. Rather, it is intended
; to be #include-d by /etc/chan_dahdi.conf that will include the global settings
;

; Span 1: OPVXA24XX/24 "OpenVox A1610 Board 25" (MASTER)
;;; line="1 OPVXA24XX/24/0 FXSKS (SWEC: MG2)"
signalling=fxs_ks
callerid=asreceived
group=0
context=from-pstn
channel => 1
callerid=
group=
context=default

;;; line="2 OPVXA24XX/24/1 FXSKS (SWEC: MG2)"
signalling=fxs_ks
callerid=asreceived
group=0
context=from-pstn
channel => 2
callerid=
group=
context=default
```

FXS ports use FXO signalling, the configure file looks like this:

```

;;; line="9 OPVXA24XX/24/8 FXOKS (SWEC: MG2)"
signalling=fxo_ks
callerid="Channel 9" <4009>
mailbox=4009
group=5
context=from-internal
channel => 9
callerid=
mailbox=
group=
context=default

;;; line="10 OPVXA24XX/24/9 FXOKS (SWEC: MG2)"
signalling=fxo_ks
callerid="Channel 10" <4010>
mailbox=4010
group=5
context=from-internal
channel => 10
callerid=
mailbox=
group=
context=default

```

Figure 15: dahdi-channels.conf

3. Starting asterisk and test calls

Checking the dahdi channel loading from asterisk console:

asterisk -vvvvvvvgc

Entering asterisk console, run command: *dahdi show channels*. If dahdi channels can be shown, which means the dahdi channels have been loaded into asterisk.

```

*CLI> dahdi show channels

```

Chan	Extension	Context	Language	MOH Interpret	Blocked	State
pseudo		default		default		In Service
1		from-pstn		default		In Service
2		from-pstn		default		In Service
3		from-pstn		default		In Service
4		from-pstn		default		In Service
5		from-pstn		default		In Service
6		from-pstn		default		In Service
7		from-pstn		default		In Service
8		from-pstn		default		In Service
9		from-internal		default		In Service
10		from-internal		default		In Service
11		from-internal		default		In Service
12		from-internal		default		In Service
13		from-internal		default		In Service
14		from-internal		default		In Service
15		from-internal		default		In Service
16		from-internal		default		In Service

Figure 16: dahdi show channels

When user makes an inbound call, CLI will show as the following:

```

localhost*CLI> -- Starting simple switch on 'DAHDI/1-1'
-- Starting simple switch on 'DAHDI/1-1'
localhost*CLI> -- Executing [s@from-pstn:1] Answer("DAHDI/1-1", "") in new stack
-- Executing [s@from-pstn:1] Answer("DAHDI/1-1", "") in new stack
localhost*CLI> -- Executing [s@from-pstn:2] NoOp("DAHDI/1-1", "" <982535362>") in new stack
-- Executing [s@from-pstn:3] Dial("DAHDI/1-1", "dahdi/9") in new stack
-- Called 9
-- DAHDI/9-1 is ringing
-- Executing [s@from-pstn:2] NoOp("DAHDI/1-1", "" <982535362>") in new stack
-- Executing [s@from-pstn:3] Dial("DAHDI/1-1", "dahdi/9") in new stack
-- Called 9
-- DAHDI/9-1 is ringing
localhost*CLI> -- DAHDI/9-1 is ringing
-- DAHDI/9-1 is ringing
localhost*CLI> -- DAHDI/9-1 answered DAHDI/1-1
-- Native bridging DAHDI/1-1 and DAHDI/9-1
-- DAHDI/9-1 answered DAHDI/1-1
-- Native bridging DAHDI/1-1 and DAHDI/9-1
localhost*CLI> -- Hanging up on 'DAHDI/9-1'
-- Hungup 'DAHDI/9-1'
-- Hanging up on 'DAHDI/9-1'
-- Hungup 'DAHDI/9-1'

```

Figure 17: inbound call

When user makes an outbound call, CLI will show as the following:

```

localhost*CLI> -- Starting simple switch on 'DAHDI/9-1'
-- Starting simple switch on 'DAHDI/9-1'
localhost*CLI> -- Executing [982535461@from-internal:1] Dial("DAHDI/9-1", "dahdi/1/982535461") in new stack
-- Called 1/982535461
-- Executing [982535461@from-internal:1] Dial("DAHDI/9-1", "dahdi/1/982535461") in new stack
-- Called 1/982535461
localhost*CLI> Dialing T982535461w on 1
-- DAHDI/1-1 answered DAHDI/9-1
-- Native bridging DAHDI/9-1 and DAHDI/1-1
-- DAHDI/1-1 answered DAHDI/9-1
-- Native bridging DAHDI/9-1 and DAHDI/1-1
localhost*CLI> -- Hanging up on 'DAHDI/1-1'
-- Hanging up on 'DAHDI/1-1'
-- Hungup 'DAHDI/1-1'
localhost*CLI> -- Hungup 'DAHDI/1-1'
== Spawn extension (from-internal, 982535461, 1) exited non-zero on 'DAHDI/9-1'
localhost*CLI> == Spawn extension (from-internal, 982535461, 1) exited non-zero on 'DAHDI/9-1'
-- Hanging up on 'DAHDI/9-1'
localhost*CLI> -- Hanging up on 'DAHDI/9-1'
-- Hungup 'DAHDI/9-1'
-- Hungup 'DAHDI/9-1'

```

Figure 18: outbound call

The status of channel looks like the following. If user uses AE1610P, EC Status of active channel should be ON; otherwise it would be OFF.

```
*CLI> dahdi show channel 9
Channel: 9
File Descriptor: 16
Span: 1
Extension:
Dialing: no
Context: from-internal
Caller ID: 4009
Calling TON: 0
Caller ID name: Channel 9
Mailbox: 4009
Destroy: 0
InAlarm: 0
Signalling Type: FXO Kewlstart
Radio: 0
Owner: DAHDI/9-1
Real: DAHDI/9-1
Callwait: <None>
Threeway: <None>
Confno: -1
Propagated Conference: -1
Real in conference: 1
DSP: yes
Busy Detection: no
TDD: no
Relax DTMF: no
Dialing/CallwaitCAS: 0/0
Default law: ulaw
Fax Handled: no
Pulse phone: no
Gains (RX/TX): 0.00/0.00
Dynamic Range Compression (RX/TX): 0.00/0.00
DND: no
Echo Cancellation:
    128 taps
    currently ON
Wait for dialtone: Ums
Slave Channel: 1
Actual Confinfo: Num/1, Mode/0x0009
Actual Confmute: No
Hookstate (FXS only): Offhook
```

Figure 19: channel status

Notes:

User can use command `cat /proc/interrupts` to confirm A1610P/AE1610P has independent IRQ. If A1610P/AE1610P shares IRQ with other device. To avoid IRQ conflict, A1610P/AE1610P enable user to adjust pin number when update firmware, please refer this manual for more details.

http://downloads.openvox.cn/pub/misc/opvx-update_user_manual_en.pdf

Test environments are:

CentOS-5.6

Kernel version: 2.6.18-238.el5

Dahdi: dahdi-linux-complete-2.4.0+2.4.0

Asterisk: 1.8.0

Hardware: OpenVox A1610P/AE1610P

4. References

www.openvox.cn

www.digium.com

www.asterisk.org

www.voip-info.org

www.asteriskguru.com