



OpenVox Communication Co.Ltd



A1610E/AE1610E on DAHDI User Manual

Version: 2.2





OpenVox Communication Co.Ltd

Most Advanced Asterisk Cards

Address: F/3, Building No.127, Jindi Industrial Zone,
Shazui Road, Futian District, Shenzhen, Guangdong 518048, China
Tel:+86-755-82535461, 82535095, 82535362, Fax:+86-755-83823074

Business Contact: sales@openvox.com.cn

Technical Support: support@openvox.com.cn

Business Hours: 09:00-18:00(GMT+8) from Monday to Friday

URL: www.openvox.cn

Thank You for Choosing OpenVox Products!

Copyright

Copyright© 2011 OpenVox Inc. All rights reserved. No part of this document may be reproduced without prior written permission.

Confidentiality

Information contained herein is of a highly sensitive nature and is confidential and proprietary to OpenVox Inc. No part may be distributed, reproduced or disclosed orally or in written form to any party other than the direct recipients without the express written consent of OpenVox Inc.

Disclaimer

OpenVox Inc. reserves the right to modify the design, characteristics, and products at any time without notification or obligation and shall not be held liable for any error or damage of any kind resulting from the use of this document.

OpenVox has made every effort to ensure that the information contained in this document is accurate and complete; however, the contents of this document are subject to revision without notice. Please contact OpenVox to ensure you have the latest version of this document.

Trademarks

All other trademarks mentioned in this document are the property of their respective owners.

Contents

General Safety Instructions	4
Test Environments	5
Chapter 1 Overview	6
1.1 What is Asterisk	6
1.2 What is A1610E/AE1610E	7
Chapter 2 Hardware Setup	10
2.1 Power supply	10
2.2 Slot compatibility	10
2.3 Timing cable	11
2.4 FXO and FXS modules	11
2.5 Splitter	12
2.6 Hardware setup procedure	12
Chapter 3 Software Installation and Configuration	14
3.1 Download	14
3.2 Installation	15
3.3 Configuration	19
Chapter 4 Reference	28
Appendix A Specifications	29
Appendix B Interfaces	30

General Safety Instructions



CAUTION

1. The computers that have A1610E/AE1610E card installed must comply with the country's specific safety regulations.
2. Only service personnel should go to install A1610E/AE1610E card.
3. Before installing A1610E/AE1610E card, please unplug the power cord and remove the cover from your PC.
4. For avoiding personal injuries and damages to your machine and A1610E/AE1610E card, make sure bracket of the card is secured to the PC's chassis ground by fastening the card with a screw.
5. Electrical Surges, ESD are very destructive to the equipment. To avoid it, make sure there is a low impedance discharge path from your computer to chassis ground.
6. To reduce the risk of damage or injury, please follow all steps or procedures as instructed.

Test Environments

CentOS-5.6

Kernel version: 2.6.18-238.12.1.el5

DAHDI: dahdi-linux-complete-current

Asterisk: 1.8.0

Hardware: OpenVox A1610E/AE1610E

Chapter 1 Overview

1.1 What is Asterisk

The Definition of Asterisk is described as follows:

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 standard-based telephony equipment using relatively cost-effective hardware. Asterisk provides Voicemail services with Directory, Call Conferencing, Interactive Voice Response, Call Queuing. It supports three-way calling, caller ID services, ADSI, IAX, SIP, H323 (as both client and gateway), MGCP (call manager only) and SCCP/Skinny (voip-info.org).

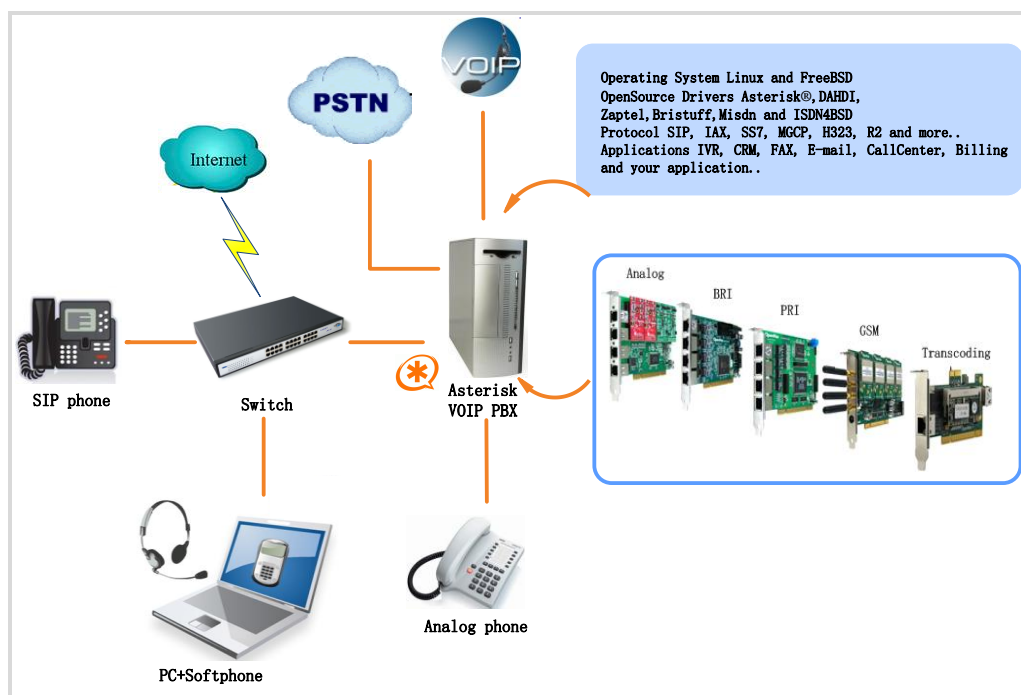


Figure 1 Topology

1.2 What is A1610E/AE1610E

A1610E is a new generation analog card and AE1610E is an A1610E with Octasic® Hardware Echo Cancellation Module on board.

A1610E/AE1610E is a new generation analog card with interchangeable quad-FXS and quad-FXO modules. It can eliminate the requirement for separate channel banks or access gateways.

The A1610E/AE1610E contains 4 module banks. Each bank supports one RJ-45 interface. The module banks may be filled with up to 4 quad-FXO or quad-FXS modules enabling the creation of any combination of ports. Scaling of an analog card solution is accomplished by simply adding additional cards.

A1610E/AE1610E works well with Asterisk®, Elastix®, FreeSWITCH™, PBX in a Flash, trixbox®, Yate™ and IPPBX/IVR projects as well as other Open Source and proprietary PBX, Switch, IVR, and VoIP gateway applications.

Target applications

- Channel Bank Replacement / Alternative
- Small Office Home Office (SOHO) applications
- Small and Medium Business (SMB) applications
- Gateway Termination to analog telephones/lines

Sample application

When you need 8 PSTN lines and 8 extension sets, for example, two FXO-400 modules and two FXS-400 modules should be installed. PSTN lines correspond to FXO modules which are red while extension lines correspond to FXS modules which are green.

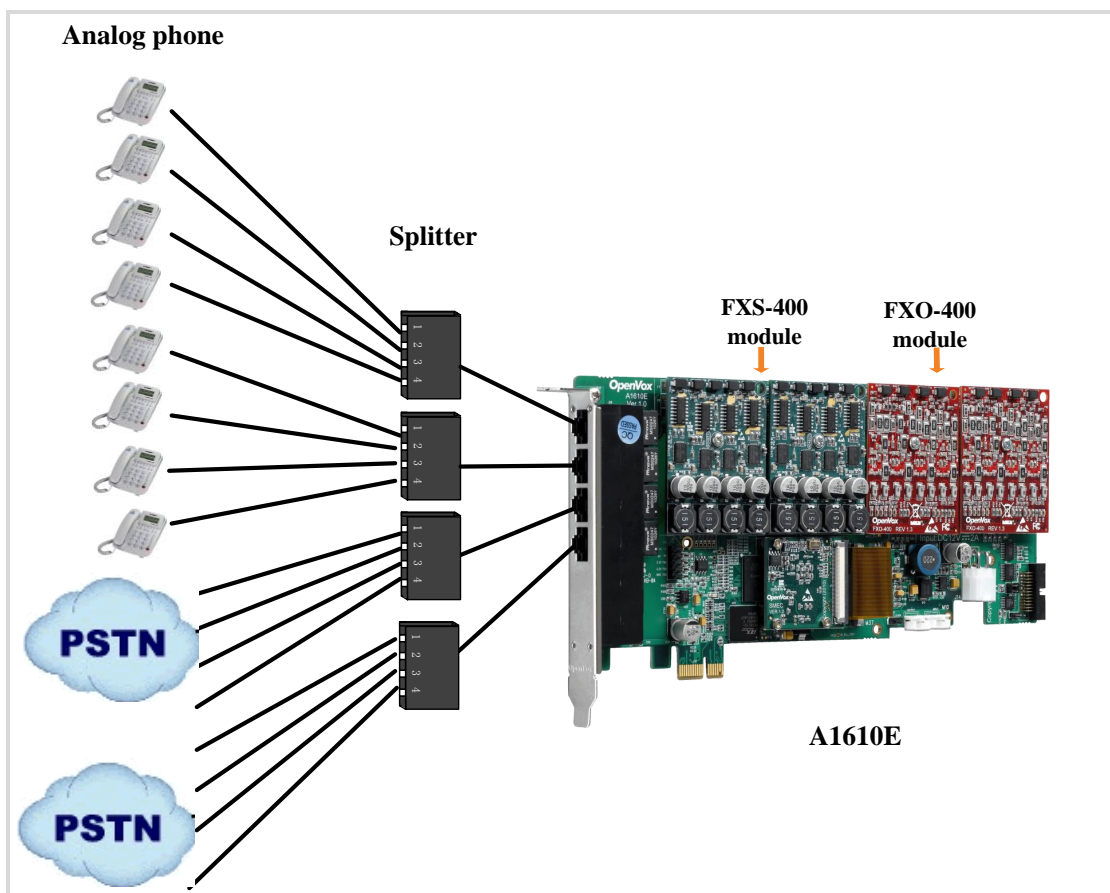


Figure 2 Sample application

Key benefits

- Takes full advantage of Octasic hardware echo cancellation module to deliver the superior voice quality on both FXO and FXS interfaces in all 16 ports

- Adjustable interrupt routing design
- Interrupts frequency adjustment
- Up to 16 simultaneous PSTN calls (per PCI-E slot)
- Compatible with all commercially available motherboards
- 3-Month “No Question Asked” return policy
- Lifetime warranty RoHS compliant
- Certificates: CE, FCC, A-Tick
- trixbox™ officially certified
- Elastix® officially certified

Features

- Caller ID and Call Waiting Caller ID
- ADSI Telephones
- Loopstart Signaling Support

Chapter 2 Hardware Setup

There are some points that should be paid attention to when set up A1610E/AE1610E.

2.1 Power supply

The board should be always powered no matter what modules are installed, please connect the power source with A1610E/AE1610E board by a 4-pin power source connector.

2.2 Slot compatibility

A1610E/AE1610E is compatible with any standard type of PCI-E $\times 1$, $\times 2$, $\times 4$, $\times 8$, and $\times 16$ slot while PCI slot is not fit for it; you should confirm your slot type and insert A1610E/AE1610E into a type of PCI-E slot as previously described.



Figure 3 PCI-E and PCI slots

2.3 Timing cable

If you have just one card in the system, all channels on that card have already run under the same clock source, so timing cable is unnecessary. But if there are more than one card, using timing cable has some advantages. Before using the clock line, each card works on its own clock, therefore precision of the clock is limited; each card will send/receive voice data at different speeds. In voice usage, this small issue can be omitted, but in data communication such as Fax/Modem, it will cause big problem. Data loss will cause communication broken or fax broken. Timing cable will force all cards to work at the same clock source, send data at the same speed, as a result no data will lost.

If you found J914 (input) and J915 (output) interfaces on the card, it means the card supports clock line, for the details, please refer to [HERE](#).

2.4 FXO and FXS modules

FXO (Foreign eXchange Office) is the office end of the line, and FXS (Foreign eXchange Station) is the station end, there is so much difference between them, they can be identified by color, the former ones are red and the latter ones are green. FXO modules use FXS signaling while FXS modules use FXO signaling. A FXO-400 module corresponds to four FXO interfaces which receive power (battery) and ring signals,

and a FXS-400 module corresponds to four FXS ports which provides power (battery) and generates ring signals.

2.5 Splitter

A RJ45 interface of A1610E/AE1610E is divided into four RJ11 ports by a specific splitter as the picture of Appendix B stated. So you should prepare for some splitters when install A1610E/AE1610E.

2.6 Hardware setup procedure

- Power off your PC, remember unplug the AC power cable
- Insert A1610E/AE1610E into a PCI-E slot
- Put timing cable correctly if necessary, for more details, please refer to [HERE](#)
- Fix the board by a screw
- Please plug PSTN lines into FXO ports and extension telephone lines into FXS interfaces before you have detected your PSTN line works well
- Power on PC

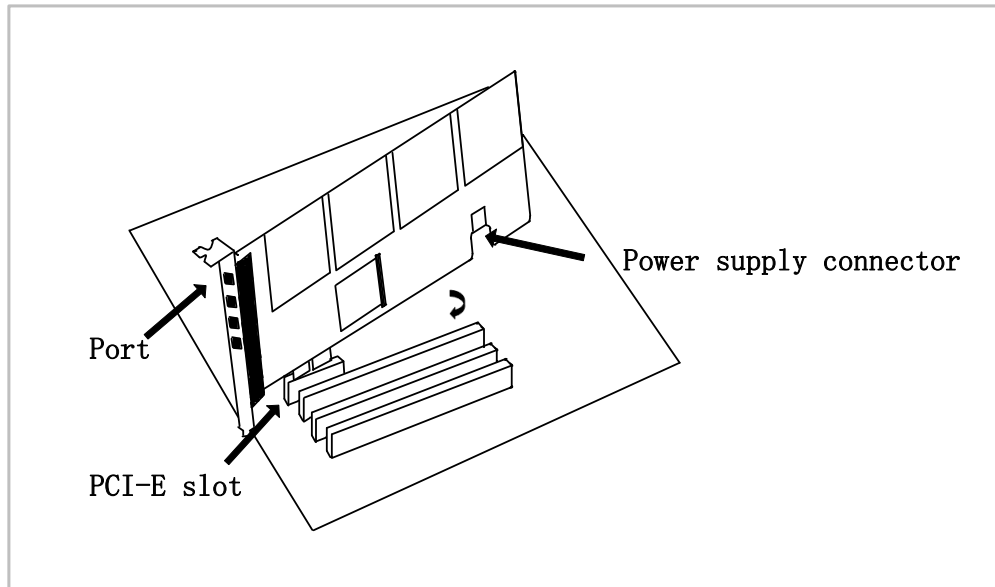


Figure 4 Hardware setup



Caution: During the above processes, an ESD wrist strap is needed. Once power is on, you must not attempt to install or take down the board. Do not forget to connect PSTN lines into analog phones directly to make sure the lines are available before inserting the PSTN lines into FXO ports. After hardware setup, it is time to install software.

Chapter 3 Software Installation and Configuration

A1610E/AE1610E supports DAHDI software driver on Linux. To make full use of A1610E/AE1610E, you should download, compile, install and configure DAHDI and Asterisk. Let's make the specific DAHDI and Asterisk version for an example to explain how to install software.

3.1 Download

DAHDI software packages are available on OpenVox official website or Digium. Some patches should be added while the driver sources are from Digium, therefore, it is recommended that downloading the DAHDI driver package from OpenVox official website. Let's make the specific DAHDI and Asterisk versions for an expatiation.

Gain DAHDI source package from OpenVox:

http://downloads.openvox.cn/pub/drivers/dahdi-linux-complete/openvox_dahdi-linux-complete-current.tar.gz

Get Asterisk software package from Digium official website:

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

Execute the following commands under the directory of /usr/src/ in general, the former two below are used for downloading these two

packages and the later two are for unzipping them.

```
# wget http://downloads.openvox.cn/pub/drivers/dahdi-linux-complete/openvox_dahdi-linux-complete-current.tar.gz
```

```
# wget http://downloads.asterisk.org/pub/telephony/asterisk/releases/asterisk-1.8.0.tar.gz
```

```
# tar -xvzf openvox_dahdi-linux-complete-current.tar.gz
```

```
# tar -xvzf asterisk-1.8.0.tar.gz
```

3.2 Installation

1. Hardware detection

```
# lspci -vvvv
```

Check the outcome and confirm your system has recognized A1610E/AE1610E. If it has been recognized, "communication controller" will be displayed in the output information like that:


```
01:02.0 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: 16 bytes
  Interrupt: pin A routed to IRQ 225
  Region 0: Memory at ded80000 (32-bit, non-prefetchable)
[size=512K]
  Kernel driver in use: opvxa24xx
  Kernel modules: opvxa24xx
```

Figure 5 Hardware detection

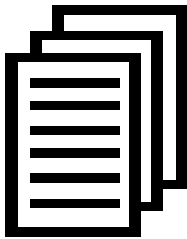
If A1610E/AE1610E is not recognized by the system, you have to power off and take out the card, then try to insert it into other PCI-E slot.

2. Software installation

Some dependencies are crucial. If any of them is absent, the software installation process would not go through successfully. Let's run "yum install XX" (XX stands for the dependency's name) to check the availability of dependencies.

```
# 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
# yum install libxml2-devel
```



If there is no kernel-devel source in the system, users should run the following command to install the kernel-devel to peer current kernel:

```
# yum install kernel-devel-`uname -r`
```

While if there is no matched kernel-devel found, you should download matched RPM package to install it, or execute the following command to update to the latest and stable kernel version:

```
# yum install kernel kernel-devel
```

After installed, please reboot your machine to apply the new kernel and install the dependencies. If the dependency has been installed, system indicates that nothing to do which means you could go to next one directly. Otherwise, the system will keep on installing it.

After install the dependencies, please change to the directory

dahdi-linux-complete-XX (XX represents DAHDI version), then perform the following commands one by one to install DAHDI.

```
# cd /usr/src/dahdi-linux-complete-XX
# make
# make install
# make config
```



Caution: If there is something wrong after "make", please refer to [HERE](#). In the url link, the moderator introduces you a method how to patch. After patching, save your changes and exit. Then run "make" again, if successfully, you are going to install Asterisk.

Please operate those commands to install Asterisk.

```
# cd asterisk-1.8.0
# ./configure
# make
# make install
# make samples
```



"make samples" will install the standard sample configuration file in the directory /etc/asterisk. As a freshman, you should perform make samples, that is to say, it is unnecessary to perform make samples every time. Because once performed, it will cover the old sample configuration files you have installed.

3.3 Configuration

1. Driver loading

After compiling and installing DAHDI and Asterisk, please load the driver by running:

```
# modprobe dahdi
# modprobe opvxa24xx opermode=CHINA
# dahdi_genconf
```



After running "modprobe dahdi" or "modprobe opvxa24xx opermode=CHINA", there is not any indication information displayed if loaded normally and successfully. "opvxa24xx" is the driver module name of A1610E/AE1610E. "opermode" applies to FXO port and is invalid for FXS port, and you are allowed to take place of "CHINA" to other mode

name which is available in the file:

```
../dahdi-linux-XX/linux/drivers/dahdi/fxo_modules.h
```

If there is any error, please trace the cause. Until all errors are clear up, you could execute "dahdi_genconf" again, and then go to the next step. By running "dahdi_genconf", it will generate /etc/dahdi/system.conf and etc/asterisk/dahdi-channels.conf automatically. Checking whether the generated files information agrees with your hardware setup, if not, you should modify to your specific requirements. Do not forget to confirm dahdi-channels.conf is included in chan_dahdi.conf, if not, run command:

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

FXO ports use FXS signaling, while FXS ports adopt FXO signaling. A part of system.conf which is one of the basic channel configuration files is displayed.

```
# Span 1: OPVXA24XX/24 "OpenVox A1610 Board 25" (MASTER)
fxoks=1
fxoks=2
fxoks=3
fxoks=4
fxoks=5
fxoks=6
fxoks=7
fxoks=8
fxsks=9
fxsks=10
fxsks=11
fxsks=12
fxsks=13
fxsks=14
fxsks=15
fxsks=16

# Global data

loadzone      = us
defaultzone   = us
```

Figure 6 A part of system.conf

2. Country mode modification

In order to match your country pattern, you need to change parameters loadzone and defaultzone to your country. For example, your system is in CHINA, you would like them change to:

```
loadzone = cn
```

```
defaultzone = cn
```



Some zonedata is available in the file

../dahdi-XX/tools/zonedata.c, you can refer to it to match your country mode. Meanwhile, you also need to modify another parameter which is in file

/etc/asterisk/indications.conf.

country=cn

A part of file /etc/asterisk/dahdi-channels.conf is showed as below.

(Modification, if it is not agree with the hardware setup)



```
; Span 1: OPVXA24XX/24 "OpenVox A1610 Board 25" (MASTER)
;;; line="1 OPVXA24XX/24/0 FXOKS"
Signalling=fxo_ks           // FXS modules use FXO signaling
callerid="Channel 1" <4001>
mailbox=4001
group=5
context=from-internal
channel => 1
callerid=
mailbox=
group=
context=default

;;; line="2 OPVXA24XX/24/1 FXOKS"
signalling=fxo_ks
callerid="Channel 2" <4002>
mailbox=4002
group=5
context=from-internal
channel => 2
callerid=
mailbox=
group=
context=default
.....
.....
.....
;;; line="9 OPVXA24XX/24/8 FXSKS"
Signalling=fxs_ks           // FXO modules use FXS signaling
callerid=asreceived
group=0
context=from-pstn
channel => 9
callerid=
group=
context=default

;;; line="10 OPVXA24XX/24/9 FXSKS"
signalling=fxs_ks
callerid=asreceived
group=0
context=from-pstn
channel => 10
callerid=
group=
context=default
.....
.....
.....
```

Figure 7 A part of dahdi-channels.conf

After modifying the country mode, please execute the following

command:

```
# dahdi_cfg -vvvvvv
```

The command is used for reading and loading parameters in the configuration file `system.conf` and writing to the hardware. A part of outputs are showed in the following figure.

Channel map:

```
Channel 01: FXO Kewlstart (Default) (Echo Canceler: none) (Slaves: 01)
Channel 02: FXO Kewlstart (Default) (Echo Canceler: none) (Slaves: 02)
Channel 03: FXO Kewlstart (Default) (Echo Canceler: none) (Slaves: 03)
Channel 04: FXO Kewlstart (Default) (Echo Canceler: none) (Slaves: 04)
Channel 05: FXO Kewlstart (Default) (Echo Canceler: none) (Slaves: 05)
Channel 06: FXO Kewlstart (Default) (Echo Canceler: none) (Slaves: 06)
Channel 07: FXO Kewlstart (Default) (Echo Canceler: none) (Slaves: 07)
Channel 08: FXO Kewlstart (Default) (Echo Canceler: none) (Slaves: 08)
Channel 09: FXS Kewlstart (Default) (Echo Canceler: none) (Slaves: 09)
Channel 10: FXS Kewlstart (Default) (Echo Canceler: none) (Slaves: 10)
Channel 11: FXS Kewlstart (Default) (Echo Canceler: none) (Slaves: 11)
Channel 12: FXS Kewlstart (Default) (Echo Canceler: none) (Slaves: 12)
Channel 13: FXS Kewlstart (Default) (Echo Canceler: none) (Slaves: 13)
Channel 14: FXS Kewlstart (Default) (Echo Canceler: none) (Slaves: 14)
Channel 15: FXS Kewlstart (Default) (Echo Canceler: none) (Slaves: 15)
Channel 16: FXS Kewlstart (Default) (Echo Canceler: none) (Slaves: 16)
```

16 channels to configure.

```
Setting echocan for channel 1 to none
Setting echocan for channel 2 to none
Setting echocan for channel 3 to none
Setting echocan for channel 4 to none
Setting echocan for channel 5 to none
Setting echocan for channel 6 to none
Setting echocan for channel 7 to none
Setting echocan for channel 8 to none
Setting echocan for channel 9 to none
Setting echocan for channel 10 to none
Setting echocan for channel 11 to none
Setting echocan for channel 12 to none
Setting echocan for channel 13 to none
Setting echocan for channel 14 to none
Setting echocan for channel 15 to none
Setting echocan for channel 16 to none
```

Figure 8 Channel map

3. Asterisk initiation

```
# asterisk -vvvvvvvgc
```

If Asterisk is already activate, run "asterisk -r" instead. In the CLI, please run the following command:

```
localhost*CLI> dahdi show channels
```

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

Figure 9 channels show

If DAHDI channels are found, it means they have been loaded into

Asterisk. You are going to edit dialplan by your requirements.

4. Dialplan edit

Users must make sure that the context "from-pstn" and "from-internal"

are in extensions.conf, here a simple example is given:

```
# vim /etc/asterisk/extensions.conf
```

```
[from-pstn]
exten => s,1,Answer()
exten => s,n,Dial(dahdi/1,,r)
exten => s,n,Hangup()

[from-internal]
exten => 200,1,Dial(dahdi/9/outgoing_number)
exten => 200,2,Hangup()
```

Figure 10 dialplan show



You should write the destination number instead of the `outgoing_number` in the above dial plan. The dial plan achieves that when an extension telephone dials 200, Asterisk will transfer through channel 9 to the destination.

While a call comes from PSTN line, Asterisk answers firstly, and then gets through to the extension set which connects to channel 1.

After saving your dialplan, please run "`asterisk -r`", then execute "`reload`" in the CLI. Next you are able to make calls.

Additional function

Users should run command "`cat /proc/interrupts`" to check A1610E/AE1610E has independent interrupt. If A1610E/AE1610E shares interrupt with other devices, it may cause some problems even cannot work normally. While A1610E/AE1610E allows users to modify interrupt pin during firmware upgrade for avoiding interrupt conflict.

Chapter 4 Reference

www.openvox.cn

www.digium.com

www.asterisk.org

www.voip-info.org

www.asteriskguru.com

Tips

Any questions during installation please consult in our forum or look up for answers from the following websites:

[Forum](#)

[wiki](#)

Appendix A Specifications

• Weight and size

Weight: 129g (A1610E) 10g (EC module)

Size: 219×111.2×18mm³

• Interfaces

PCI-E Bus: Be compatible with standard PCI-E ×1, ×2, ×4, ×8, ×16 slot

Power Supply Connector: 4-pin connector

RJ45 connector

• Environment

Temperature: 0 ~50 °C (Operation)

 - 40 ~125 °C (Storage)

Humidity: 10 ~90% NON-CONDENSING

• Power consumption

Power: 8.21W Minimum, 88.24W Maximum

• Hardware and software requirements

RAM 128 + MB

Linux kernel 2.4.X or 2.6.X

CPU 800+ MHZ

Appendix B Interfaces

There are up to 4 FXS-400/FXO-400 modules on every

A1610E/AE1610E, a module corresponds to a RJ45 port which

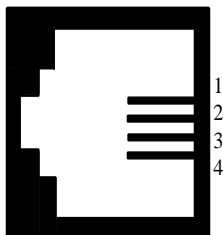
A1610E/AE1610E takes 2 of 8 pins for a pair connector to your 4-pin or 6-pin telephone line, so each RJ45 socket is divided into 4 telephone lines

by a splitter. Either 4-pin or 6-pin RJ11 port is compatible with the

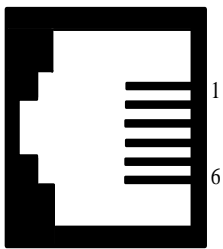
splitter, let's illustrate pin assignments of RJ11 and RJ45 port by the

following tables and figure.

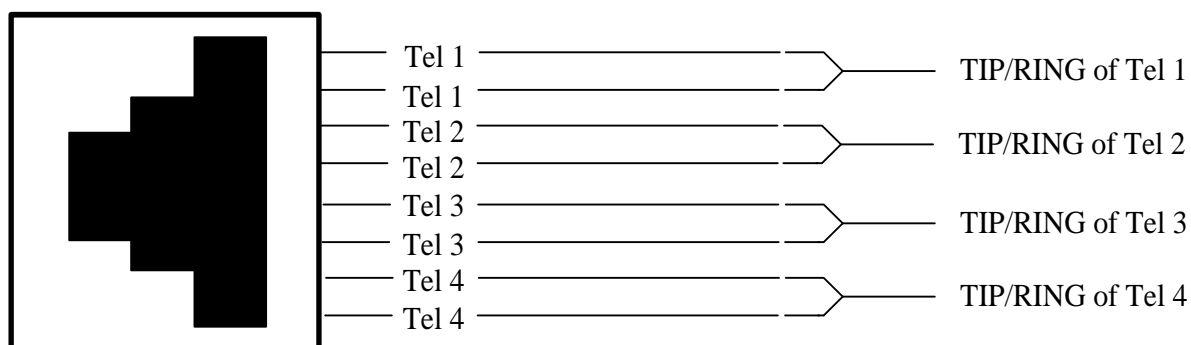
4-pin RJ11 port

4-pin RJ11 port	PIN	Description
	1	Not used
	2	Tip
	3	Ring
	4	Not used

6-pin RJ11 port

6-pin RJ11 port	PIN	Description
	1	Not used
	2	Not used
	3	Tip
	4	Ring
	5	Not used
	6	Not used

RJ-45 interface



Splitter

