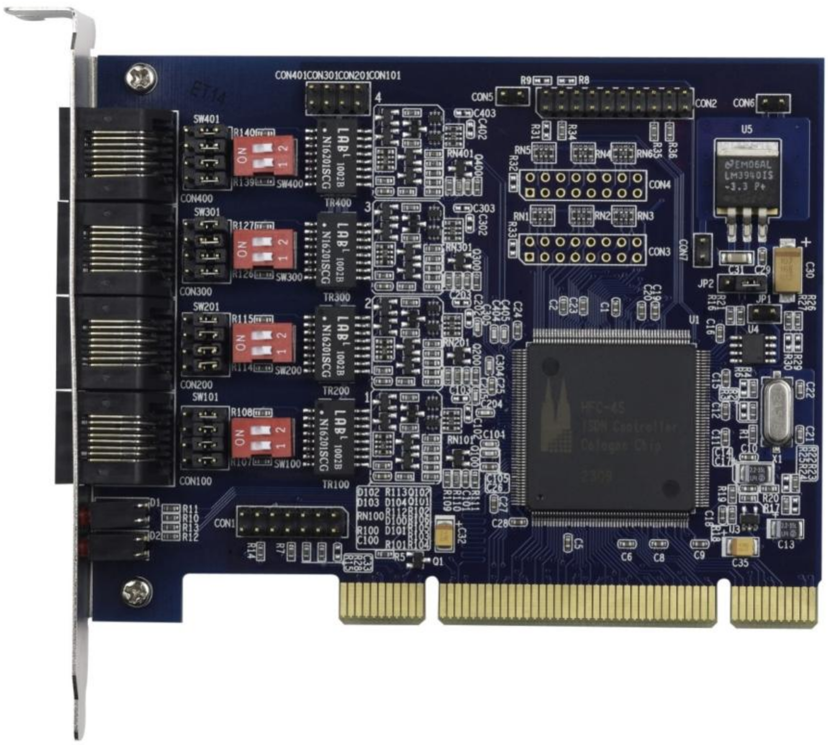

Figure: AX-4S

ATCOM® Digital Card AX-4S

Product Guide

Version: 1.0

Written by Robert.Ao

18-01-2010

The Installation of AX-4S with Centos 5.4

Content

Contact ATCOM

The Introduction of ATCOM

Founded in 1998, ATCOM technology has been always endeavoring in the R&D and manufacturing of the internet communication terminals. The product line of ATCOM includes IP Phone, USB Phone, IP PBX, VoIP gateway and Asterisk Card.

Contact sales:

Address	District C, east of 2nd floor, #3, Crown industry buildings, Chegongmiao Industry area, Futian district, Shenzhen, China
Tel	+(86)755-23487618
Fax	+(86)755-23485319
E-mail	sales@atcomemail.com

Contact Technical Support:

Tel	+(86)755-23481119
E-mail	Support@atcomemail.com

Website address: http://www.atcom.cn/

ATCOM Wiki Website: http://www.openippbx.org/index.php?title=Main_Page

Download Center: http://www.atcom.cn/download.html

Chapter 1 the Introduction of AX-4S

Overview of the AX-4S

AX-4S Asterisk card is the telephony PCI card that support four ISDN BRI ports.
Using AX-4S digital ISDN Bri card, open source Asterisk PBX and PC, users can create their IP PBX telephony solution including all the sophisticated features of traditional PBX, and extend features such as voicemail in IP PBX.

Features

4 Basic Rate Interface ports for TE and NT mode
Hardware DTMF detection
Conference Bridge
Point-to-Point (TE/NT) and Point-to-Multipoint (TE/NT) Euro ISDN protocol stack
Suitable for 3.3 volts and 5.0 volts 32 bit PCI 2.2 slots

Applications

ISDN BRI IP PBX
ISDN least cost router
Voice over IP BRI termination gateways
IVR system
Traditional Calls/VoIP Calls Conference

Hardware requirement

1.6-GHz Pentium IV
512 MB RAM
3.3V or 5V PCI 2.2 slot

PCI card dimension:

95mm (height) ×120mm (Length)

Chapter 2 Hardware Setting

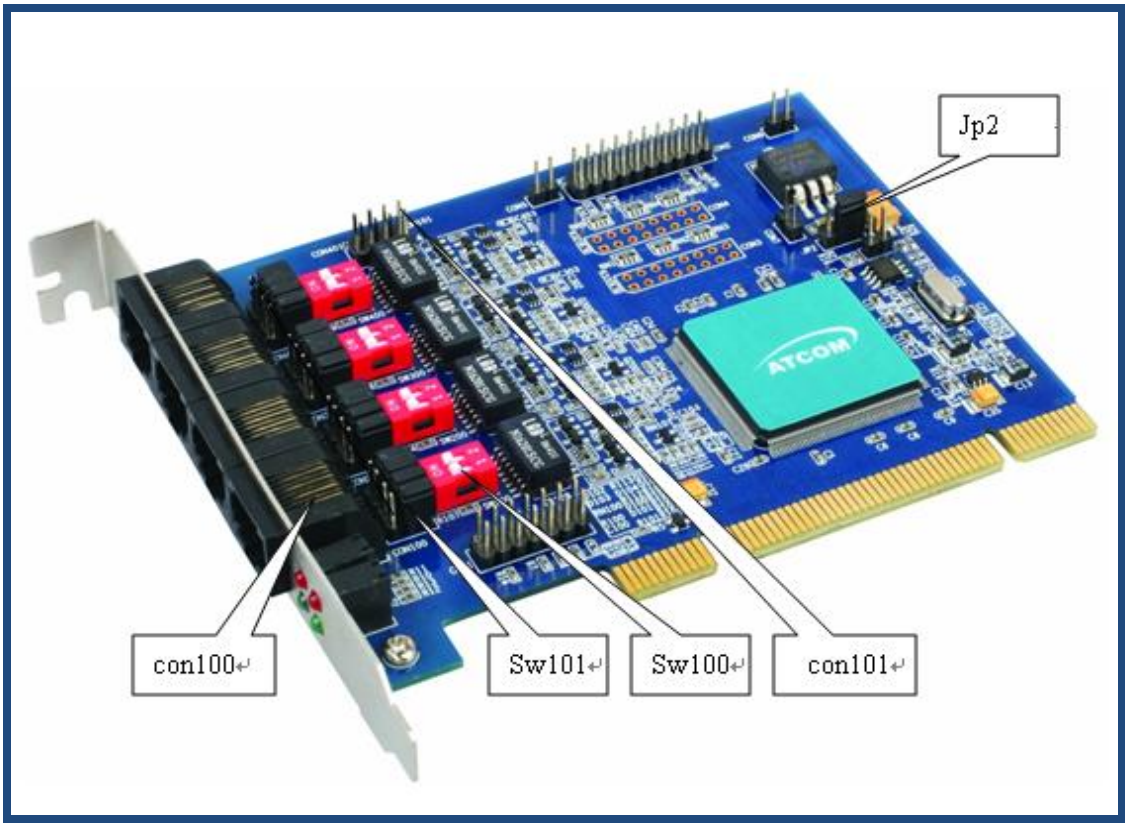

Figure 1: AX-4S

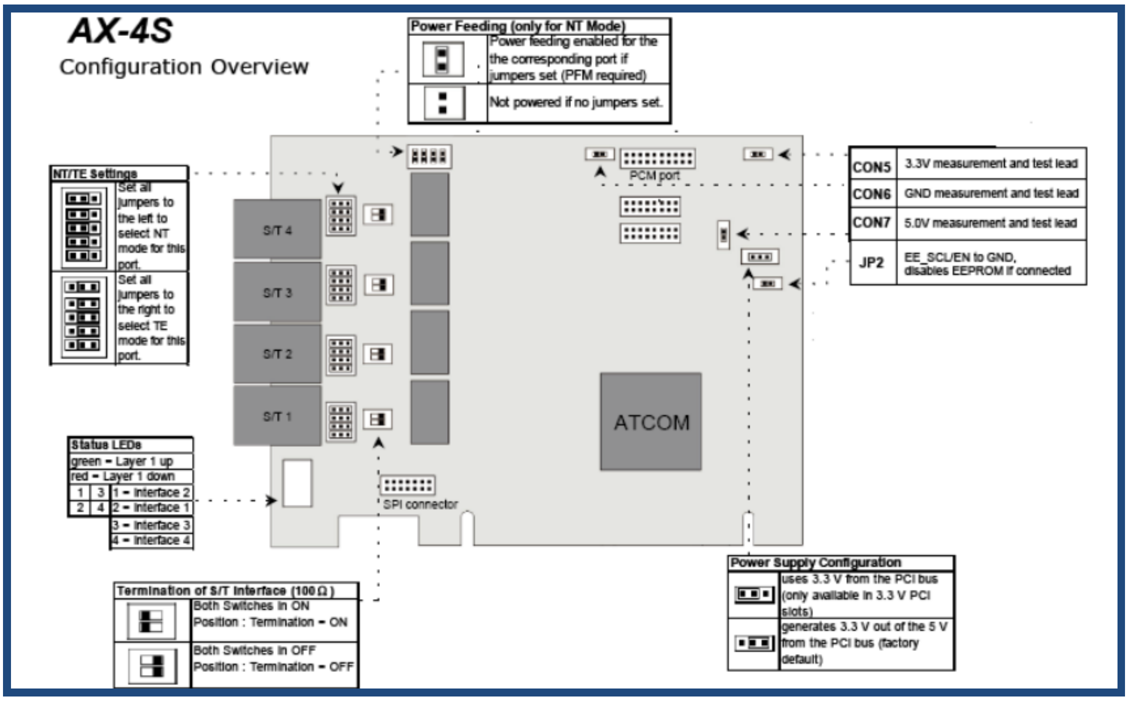

Figure 2: the Jumpers of AX-4S

SW100 100Ω termination of the 1st port

SW101 TE/NT mode switch of the 1st port

SW200 100Ω termination of the 2nd port
SW201 TE/NT mode switch of the 2nd port
SW300 100Ω termination of the 3rd port
SW301 TE/NT mode switch of the 3rd port
SW400 100Ω termination of the 4th port
SW401 TE/NT mode switch of the 4th port
Set the NT/TE mode of ports to meet your requirement.

- Ports are in NT mode with100Ω in switch position "ON".
- A S/T interface is configured to be in TE mode with the switch slide near to the termination Switch. When the slide is near to the ISDN jack, NT mode configuration is selected.

CON1 SPI connector
CON2 PCM port
CON5 3.3V measurement and test lead
CON6 GND measurement and test lead
CON7 5.0V measurement and test lead
CON100 RJ45connector of the 1st port
CON101 NT power feeding of the 1st port
CON200 RJ45 connector of the 2nd port
CON201 NT power feeding of the 2nd port
CON300 RJ45connector of the 3rd port
CON301 NT power feeding of the 3rd port
CON400 RJ45connector of the 4th port
CON401 NT power feeding of the 4th port

List of Jumpers:
JP1 EE_SCL/EN to GND, disables EEPROM if connected
JP2 3.3V power supply from PCI interface or voltage regulator

Set the NT/TE Mode for ports

Here, I take the port 1 for example, if you want it to be in NT mode, please do as the following steps:

1. Adjust the jumper of Sw101 near to the con100
2. Keep both 1 and 2 of Sw100 in position “ON”
3. Use a jumper to connect the two pins of con101, which will feed the power for terminal equipment.

If you want the port 1 to be in TE mode, please do as the following steps:

1. Adjust the jumper of Sw101 near to Sw100.
2. Both 1 and 2 of Sw100 are in “ON” or “OFF” position will be OK.
3. Do not use jumper for con101, it do not need power feeding.

Adjust the jumper of JP2 to adapt to pci slot power feeding

1. If your pci slot provides 3.3V power, please use a jumper to connect the left two pins of JP2.
2. If your pci slot provides 5V power, please use a jumper to connect the right two pins of JP2.

Chapter 3 Software Installation

Test Environment:

asterisk-1.6.1.12
mISDN-1_1_9.1
mISDNuser-1_1_9.1
Centos 5.4

After inserting the card into your PCI slot and boot your server, please use the "lspci -vv" command to check the PCI bus compatibility. The correct output will like the following:

05:04.0 ISDN controller: Cologne Chip Designs GmbH ISDN network Controller [HFC-4S] (rev 01)

Subsystem: Cologne Chip Designs GmbH ISDN network Controller [HFC-4S]
Flags: medium devsel, IRQ 50
I/O ports at 1000 [size=8]
Memory at f0500000 (32-bit, non-prefetchable) [disabled] [size=4K]
Capabilities: [40] Power Management version 2

A Cologne Chip device will be found, if you can not see the Cologne Chip device, please poweroff your server and try another PCI slot, if it still does not help, you have to check the compatibility issue between the card and your PCI bus.

1. To install asterisk and mISDN in centos OS, we have to install the following prerequisite packages:
   bison bison-devel zlib zlib-devel openssl openssl-devel gnutls-devel flex gcc gcc-c++
   Please use the yum install command to install the above packages.

2. Download asterisk,mISDN and mISDNuser
   [root@localhost src]#
   wget http://downloads.asterisk.org/pub/telephony/asterisk/releases/asterisk-xx
   [root@localhost src]# wget http://www.misdn.org/downloads/mISDN.tar.gz
   [root@localhost src]# wget http://www.misdn.org/downloads/mISDNuser.tar.gz

3. Install asterisk,mISDN and mISDNuser
   Install mISDN
   1) [root@localhost src]# tar -xvzf mISDN.tar.gz
   2) [root@localhost src]# cd mISDN-1_1_9.1/
   3) [root@localhost mISDN-1_1_9.1]# make
   4) [root@localhost mISDN-1_1_9.1]# make install

Install mISDNuser

1) [root@localhost src]# tar -xvzf mISDNuser.tar.gz
2) [root@localhost src]# cd mISDNuser-1_1_9.1/
3) [root@localhost mISDNuser-1_1_9.1]# make
4) [root@localhost mISDNuser-1_1_9.1]# make install

Install asterisk

1) [root@localhost src]# tar -xvzf asterisk-1.6.1.12.tar.gz
2) [root@localhost src]# cd asterisk-1.6.1.12
3) [root@localhost asterisk-1.6.1.12]# ./configure
4) [root@localhost asterisk-1.6.1.12]# make
5) [root@localhost asterisk-1.6.1.12]# make install
6) [root@localhost asterisk-1.6.1.12]# make samples

Chapter 4 Software Configuration

1. Please add the following lines in the end of file /etc/modprobe.d/blacklist
   blacklist hisax
   blacklist hisax_fcpcipnp
   blacklist hisax_isac
   blacklist crc_ccitt
   blacklist isdn
   blacklist slhc
   blacklist capi
   blacklist capifs
   blacklist kernelcapi
   blacklist kernel_capi
   blacklist avmfritz
   blacklist hfc4s8s_l1

2. Please run misdn-init scan
   [root@localhost etc]# misdn-init scan
   The correct output are like the following:
   [OK] found the following devices:
   card=1,0x4
   [ii] run "/usr/sbin/misdn-init config" to store this information to /etc/misdn-init.conf

3. Please run misdn-init config
   [root@localhost etc]# misdn-init config
   The correct output are like the following:
   [OK] /etc/misdn-init.conf already present. backing it up to /etc/misdn-init.conf.save
   [OK] /etc/misdn-init.conf created. It's now safe to run "/usr/sbin/misdn-init start"
   [ii] make your ports (1-4) available in asterisk by editing "/etc/asterisk/misdn.conf"

4. Please run misdn-init start
   [root@localhost etc]# misdn-init start
   The correct output are like the following:

   Loading module(s) for your misdn-cards:

   /sbin/modprobe --ignore-install hfcmulti type=0x4 protocol=0x2,0x2,0x2,0x2 layermask=0xf,0xf,0xf,0xf poll=128 debug=0
   /sbin/modprobe mISDN_dsp debug=0x0 options=0 poll=128 dtmfthreshold=100

5. Please run misdnportinfo

[root@localhost etc]# misdnportinfo

The correct output are like the following:

```
Port    1: NT-mode BRI S/T interface port (for phones)
-> Interface can be Poin-To-Point/Multipoint.
--------
Port    2: TE-mode BRI S/T interface line (for phone lines)
-> Protocol: DSS1 (Euro ISDN)
-> childcnt: 2
--------
Port    3: TE-mode BRI S/T interface line (for phone lines)
-> Protocol: DSS1 (Euro ISDN)
-> childcnt: 2
--------
Port    4: TE-mode BRI S/T interface line (for phone lines)
-> Protocol: DSS1 (Euro ISDN)
-> childcnt: 2
--------
mISDN_close: fid(3) isize(131072) inbuf(0x8573060) irp(0x8573060) iend(0x8573060)
```

6. Leds Status

If the card driver has been loaded correctly, two of the leds are green, and the other two of the leds are red.

Chapter 5 Testing

In the following procedures, we will use one straight-through cable to test ports. Here, I plug the straight-through cable into the port 1 and port 4, set the port 1 as NT mode and port 4 as TE mode. After configurating correctly, we can register sip phones and call each other through the port 1 and port 4 connected by straight-through cable.

1. Please edit misdn-init.conf file

   [root@localhost etc]# vim misdn-init.conf

   Please find the te_ptmp=1,2,3,4 line, and change it into the following two lines:

   te_ptmp=2,3,4

   nt_ptmp=1

2. Please edit misdn.conf file

   [root@localhost asterisk]# vim misdn.conf

   Please disable all the default ports, and add the following paragraph at the end of the file:

   [from-atcom]

   ports=2,3,4

   context=from-isdn

   [from-internal]

   ports=1

   context=from-internal

3. Please configure dial-plan in extensions.conf file

   I add a dial-plan like the following:

   [from-internal]

   exten =>200,1,Answer()

   exten =>200,2,Dial(misdn/1/1000)

   exten =>200,3,Hangup()

   [from-isdn]

   exten => _x.,1,Playback(demo-instruct)

   exten => _x.,2,Goto(1)

   exten => _x.,3,Hangup()

4. Please run asterisk

   asterisk –vvvvvvvvgc

   reload

5. Please run misdn show stacks

*CLI> misdn show stacks

The correct output are like the following:

BEGIN STACK_LIST:

* Port 1 Type NT Prot. PMP L2Link UP L1Link:UP Blocked:0 Debug:0
* Port 2 Type TE Prot. PMP L2Link DOWN L1Link:DOWN Blocked:0 Debug:0
* Port 3 Type TE Prot. PMP L2Link DOWN L1Link:DOWN Blocked:0 Debug:0
* Port 4 Type TE Prot. PMP L2Link DOWN L1Link:UP Blocked:0 Debug:0

*CLI>

From the above output, we can see that the links of both port 1 and port 4 are in up status. It proves that the cable connection between the two ports are correct.

6. Please use a registered sip phone to call 200, then we can get the following output:

-- Executing [200@from-internal:1] Answer("SIP/300-00000000", "") in new stack

-- Executing [200@from-internal:2] Dial("SIP/300-00000000", "misdn/1/1000") in new stack

-- Called 1/1000

-- Executing [1000@from-isdn:1] Playback("mISDN/7-u1", "demo-instruct") in new stack

-- <mISDN/7-u1> Playing 'demo-instruct.gsm' (language 'en')

P[1] We already have a channel (1)

-- mISDN/tmp0-u0 is proceeding passing it to SIP/300-00000000

-- mISDN/tmp0-u0 answered SIP/300-00000000

Chapter 6 Reference

http://www.misdn.org/index.php/Main_Page
http://www.asteriskguru.com/
http://www.asterisk.org/downloads
http://www.openippbx.org/index.php?title=Main_Page
http://www.atcom.cn/