testing environment

System: Rocky Linux (CentOS 8)
Dahdi-3.4.0
Asterisk-20
Libpri 1.6.1

download

Download the DAHDI source code package from the OpenVox official website
https://www.openvoxtech.com/pub/drivers/dahdi-linux-complete/openvox_dahdi-linux-complete-current.tar.gz
Get Libpri from Digium website:
https://downloads.asterisk.org/pub/telephony/libpri/libpri-1-current.tar.gz
Get Asterisk from Digium website:
http://downloads.asterisk.org/pub/telephony/asterisk/asterisk-20-current.tar.gz
Usually run the following command in the directory/etc/src/to download and decompress DAHDI, Asterisk, and Libpri:

```
#wget   https://www.openvoxtech.com/pub/drivers/dahdi-linux-complete/openvox_dahdi-linux-complete-current.tar.gz
#wget   https://downloads.asterisk.org/pub/telephony/asterisk/asterisk-20-current.tar.gz
#wget   https://downloads.asterisk.org/pub/telephony/libpri/libpri-1-current.tar.gz
# tar -xvzf openvox_dahdi-linux-complete-current.tar.gz
# tar  –xvzf asterisk-20-current.tar.gz
# tar  –xvzf libpri-current.tar.gz
```

install

Dependency package installation
Before installing DAHDI, please check if all dependency packages have been successfully installed. If the dependency packages have not been installed, it will result in subsequent software installations being unable to proceed smoothly

```
#dnf -y install epel-release
#dnf group -y install "Development Tools"
#dnf -y install git wget vim net-tools sqlite-devel psmisc bison ncurses ncurses-devel libtermcap-devel newt-devel libxml2-devel libtiff-devel gtk2-devel libtool libuuid-devel subversion initscripts kernel-devel kernel-devel-$(uname -r) crontabs cronie-anacron libedit libedit-devel   zlib zlib-devel openssl openssl-devel gnutls-devel gcc gcc-c++
```

If no matching kernel level is found in the update source, it is necessary to download the matching RPM package for manual installation, or perform the following specified upgrade to the latest stable kernel version:

```
# dnf install kernel kernel-devel
```

After installation, restart the system and apply the new kernel to continue the subsequent operations on the new kernel.

In the above dependency package detection process, if the dependency package has already been installed, the system will prompt that it has been installed and will not continue to install this package. The user can run the next command to install other packages;

If not installed, it will be automatically installed until the system prompts for successful installation.

Dahdi installation

Convert the path to the directory of the dahdi linux-comple-XX source code package (XX represents the DAHDI version), and run the following command to install DAHDI:

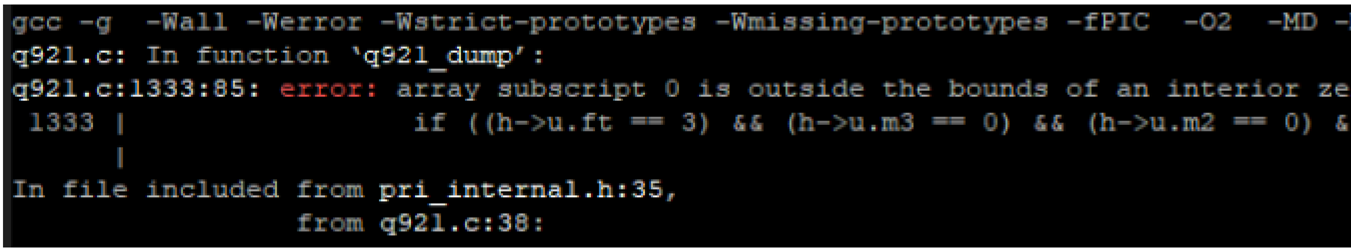
```
# cd /usr/src/dahdi-linux-complete-XX
# make
# make install
# make install-config
#systemctl enable dahdi
```

Libpri installation

Convert the path to the directory where the libpri source code package is located, and run the following command to install Libpri:

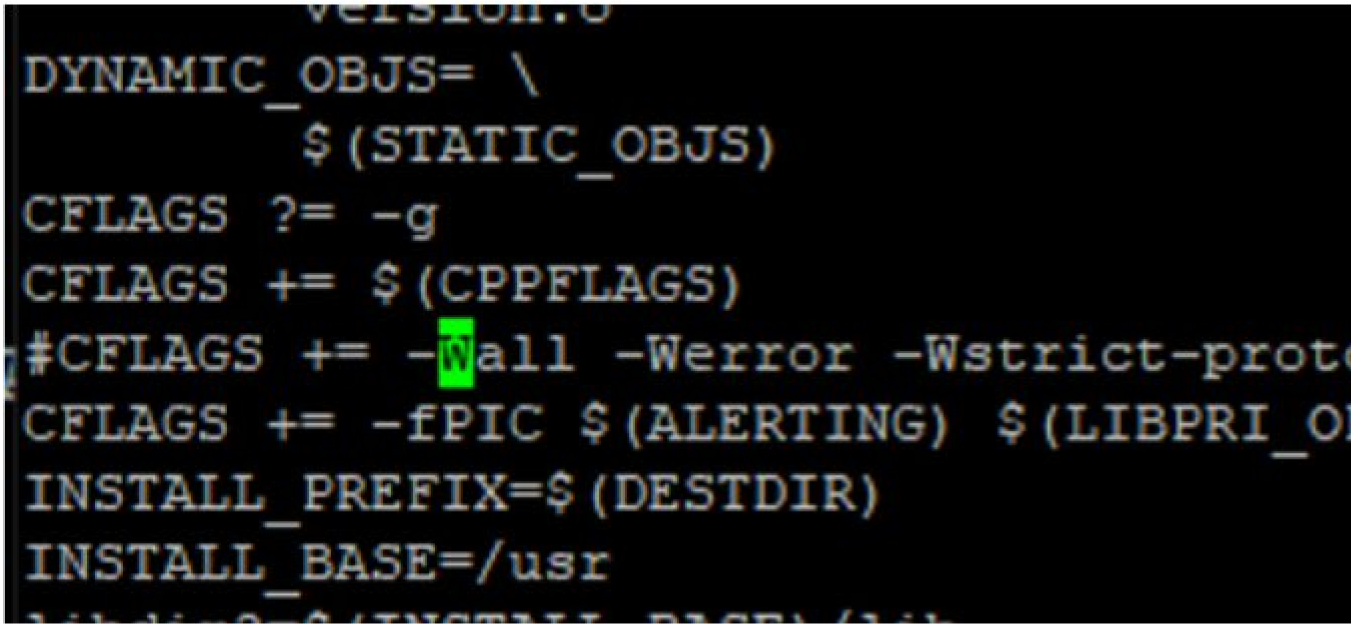
```
# cd libpri-XX
# make
# make install
```

Compiling encountered the following issues



```
gcc -g  -Wall -Werror -Wstrict-prototypes -Wmissing-prototypes -fPIC  -O2  -MD -
q921.c: In function ‘q921_dump’:
q921.c:1333:85: error: array subscript 0 is outside the bounds of an interior ze
 1333 |                 if ((h->u.ft == 3) && (h->u.m3 == 0) && (h->u.m2 == 0) &
      |
In file included from pri_internal.h:35,
                 from q921.c:38:
```

Solution: Annotate the - Wall line in the libpri-1.6.0/Makefile file file



```
        version.o
DYNAMIC_OBJS= \
        $(STATIC_OBJS)
CFLAGS ?= -g
CFLAGS += $(CPPFLAGS)
#CFLAGS += -Wall -Werror -Wstrict-prot
CFLAGS += -fPIC $(ALERTING) $(LIBPRI_O
INSTALL_PREFIX=$(DESTDIR)
INSTALL_BASE=/usr
```

Asterisk installation

Convert the path to the Asterisk source code package directory (XX represents Asterisk version), and run the following command to install Asterisk:

```
# cd asterisk-20.xx
#contrib/scripts/install_prereq install
# . /configure --libdir=/usr/lib64 --with-jansson-bundled=yes
# make
# make install
# make samples
#systemctl enable asterisk
```

configuration

Load Driver

After compilation, please run the following instructions to load the driver according to the corresponding board model:

The corresponding driver for the d130 board is opvxd115, d230, d430, d830, and d1630, and the corresponding driver for d1630 is wct4xxp

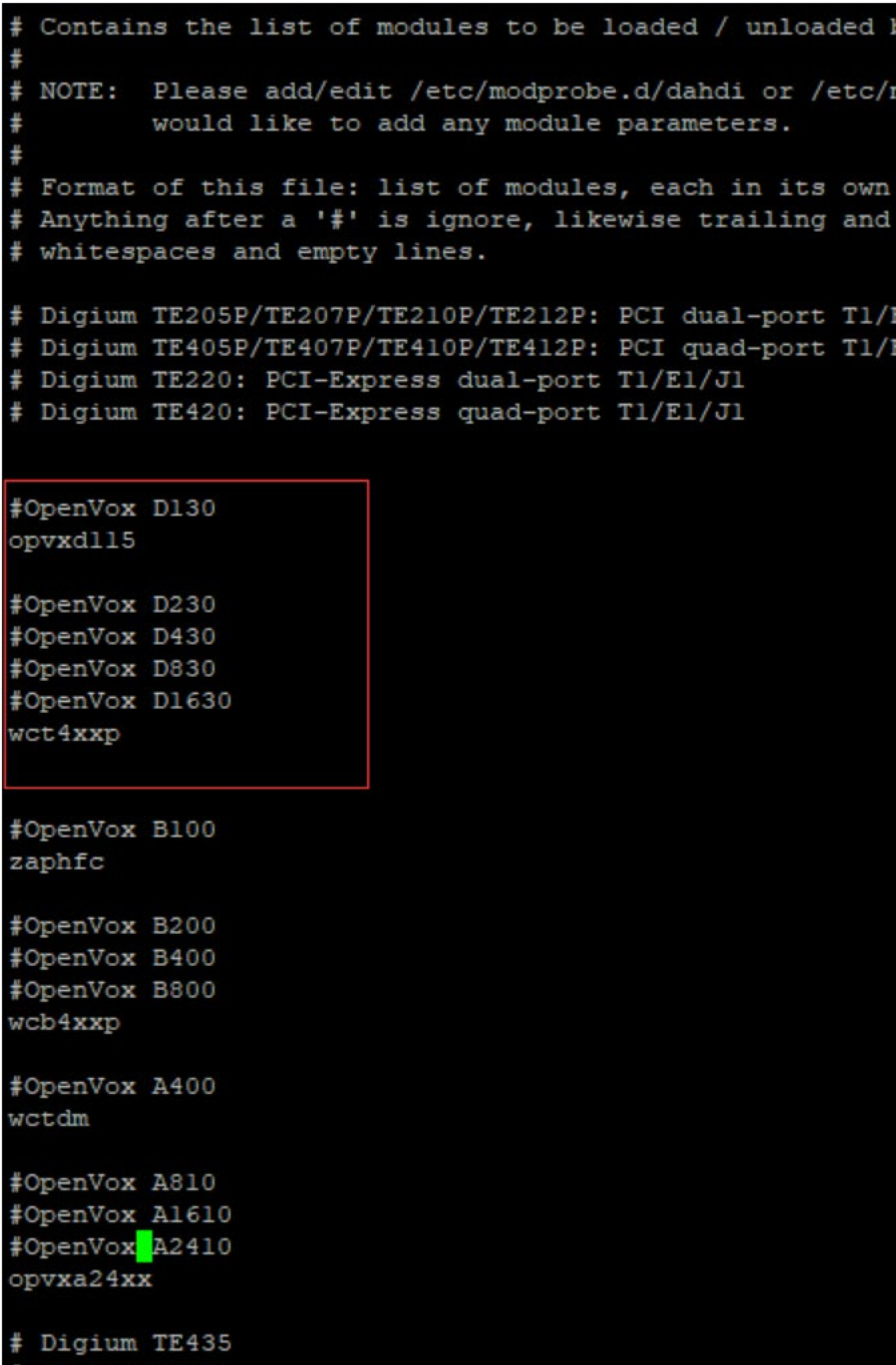
```
# modprobe dahdi
# modprobe opvxd115（D130）
# modprobe wct4xxp (D230,D430,D830,D1630)
# dahdi_genconf -vvv
```

Under normal circumstances, after executing the command "dahdi_gengconf", the system will automatically generate two files:/etc/dahdi/system.exe and/etc/asterisk/dahdi-channels.exe. Check if the generated configuration file meets your requirements, or you can manually modify the relevant parameters. It is worth noting that it is confirmed that dahdi-channels.conf is included in chan_dahdi.conf. If not, please run the command:

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

Add corresponding board drivers to/etc/dahdi/modules to achieve automatic loading of board drivers upon startup

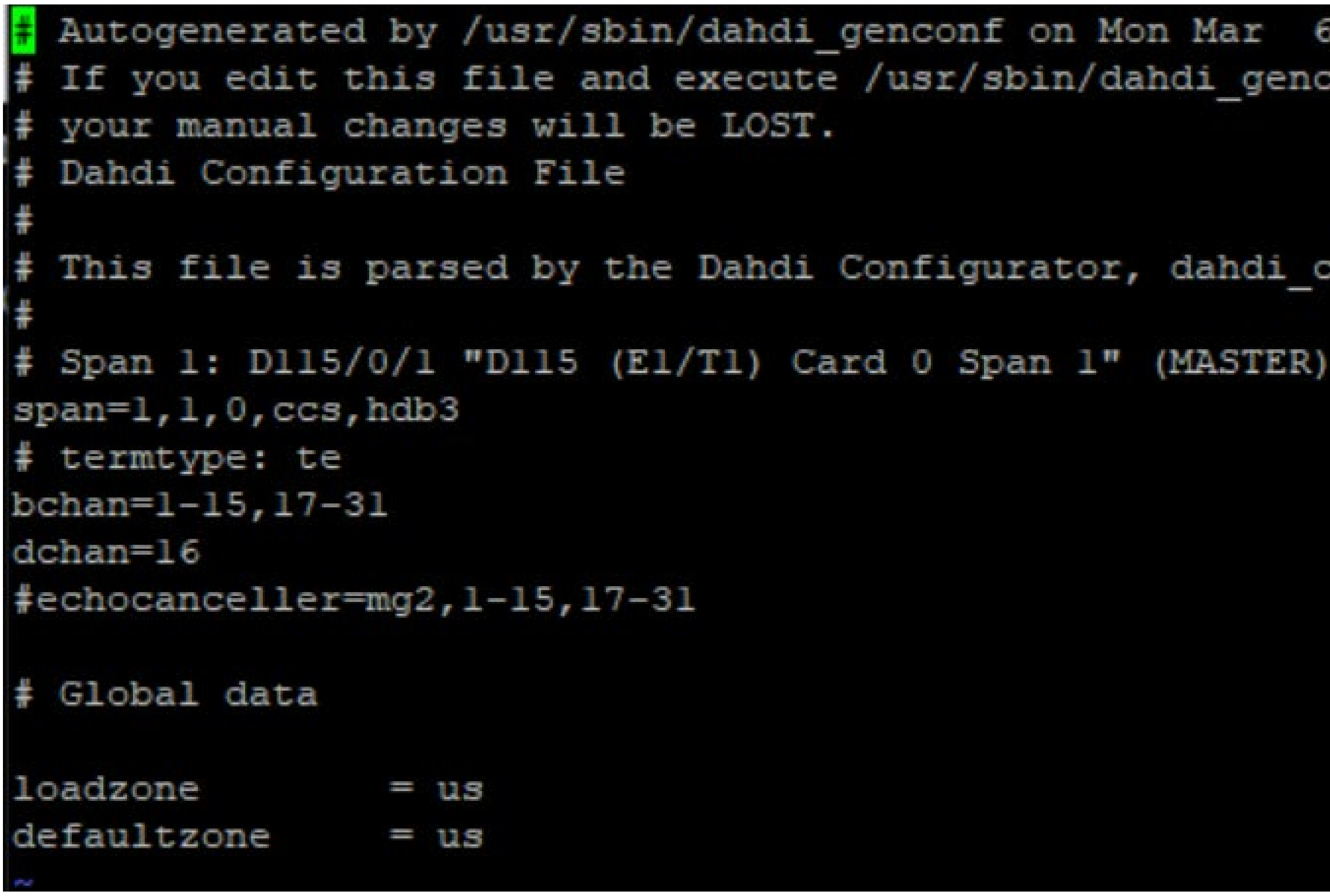
```
#vi /etc/dahdi/modules
```



```
# Contains the list of modules to be loaded / unloaded
#
# NOTE:  Please add/edit /etc/modprobe.d/dahdi or /etc/
#        would like to add any module parameters.
#
# Format of this file: list of modules, each in its own
# Anything after a '#' is ignore, likewise trailing and
# whitespaces and empty lines.

# Digium TE205P/TE207P/TE210P/TE212P: PCI dual-port T1/
# Digium TE405P/TE407P/TE410P/TE412P: PCI quad-port T1/
# Digium TE220: PCI-Express dual-port T1/E1/J1
# Digium TE420: PCI-Express quad-port T1/E1/J1


#OpenVox D130
opvxd115

#OpenVox D230
#OpenVox D430
#OpenVox D830
#OpenVox D1630
wct4xxp


#OpenVox B100
zaphfc

#OpenVox B200
#OpenVox B400
#OpenVox B800
wcb4xxp

#OpenVox A400
wctdm

#OpenVox A810
#OpenVox A1610
#OpenVox A2410
opvxa24xx

# Digium TE435
```

Using D130 board as an example to configure

The following shows a portion of the basic channel configuration file/etc/dahdi/systemconf:



```
# Autogenerated by /usr/sbin/dahdi_genconf on Mon Mar  6
# If you edit this file and execute /usr/sbin/dahdi_genc
# your manual changes will be LOST.
# Dahdi Configuration File
#
# This file is parsed by the Dahdi Configurator, dahdi_c
#
# Span 1: D115/0/1 "D115 (E1/T1) Card 0 Span 1" (MASTER)
span=1,1,0,ccs,hdb3
# termtype: te
bchan=1-15,17-31
dchan=16
#echocanceller=mg2,1-15,17-31

# Global data

loadzone        = us
defaultzone     = us
~
```

Annotate the echocanceler line

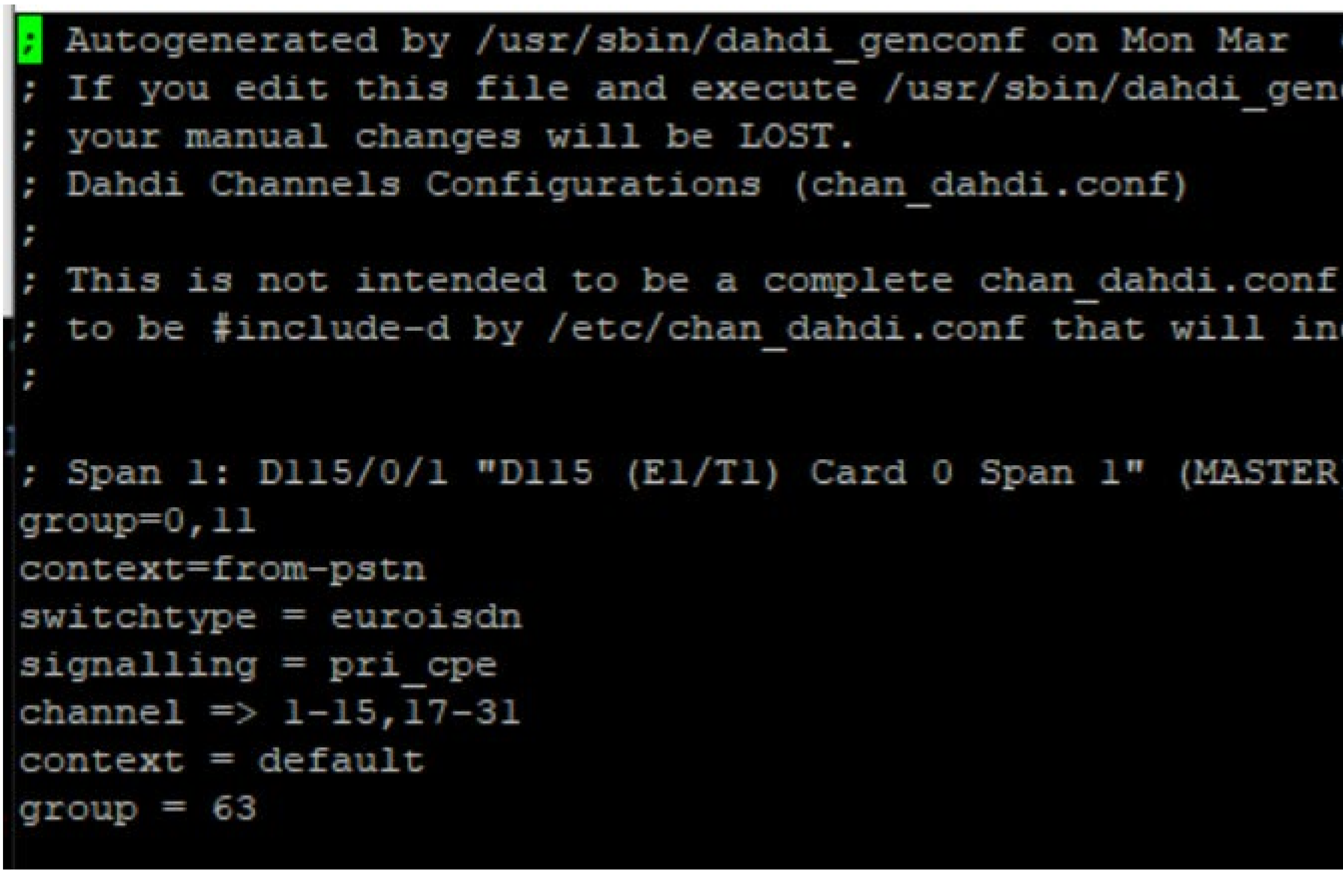
Determine whether crc4 verification needs to be enabled

If the E1 operator has not enabled crc4 verification, crc4 needs to be removed and modified to

span=1,1,0,ccs,hdb3

In China, operators generally do not enable crc4

A part of the file/etc/asterisk/dahdi-channels.exe is shown in the figure:

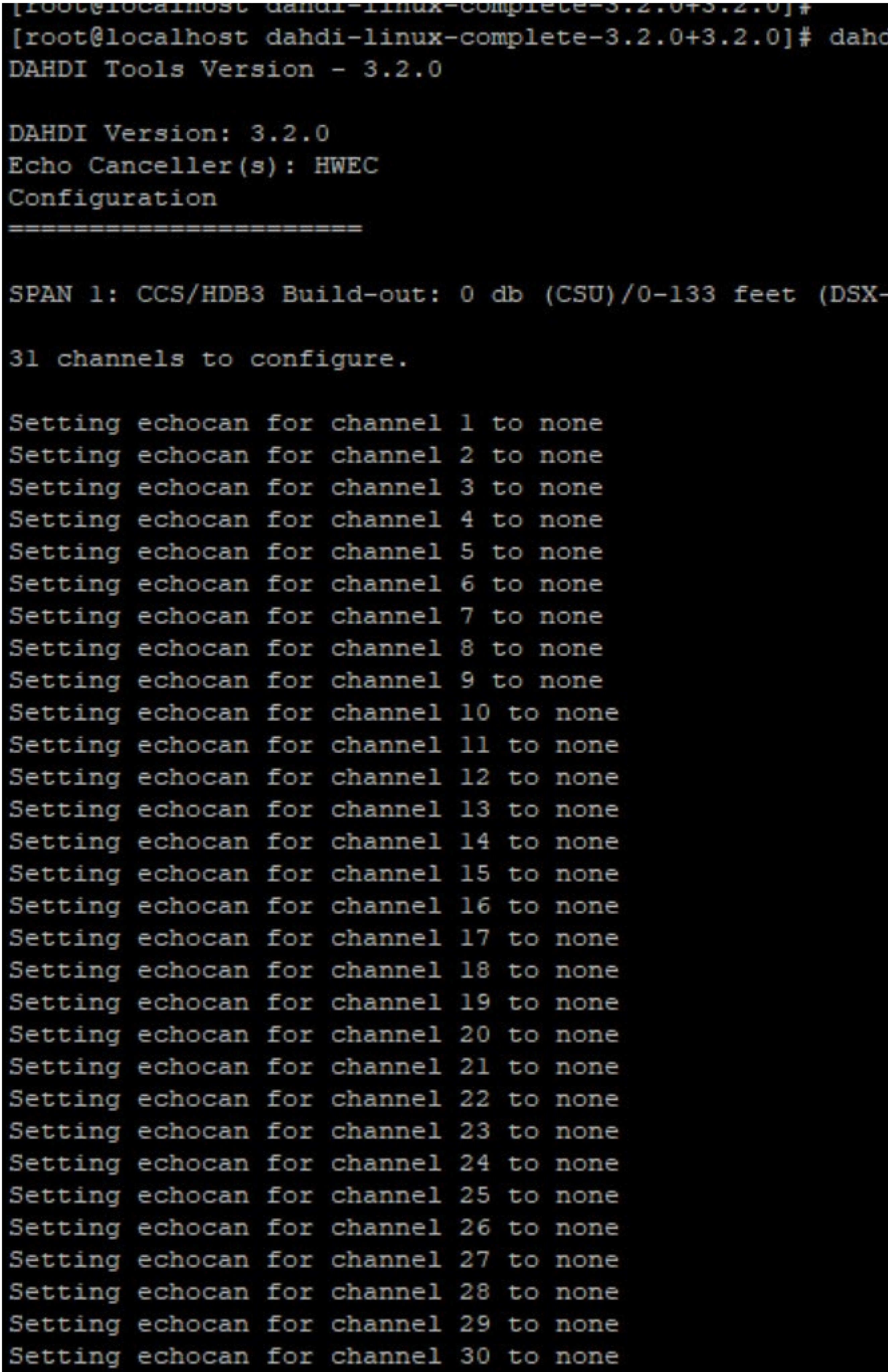


```
; Autogenerated by /usr/sbin/dahdi_genconf on Mon Mar  
; If you edit this file and execute /usr/sbin/dahdi_gen
; your manual changes will be LOST.
; Dahdi Channels Configurations (chan_dahdi.conf)
;
; This is not intended to be a complete chan_dahdi.conf
; to be #include-d by /etc/chan_dahdi.conf that will in
;

; Span 1: D115/0/1 "D115 (E1/T1) Card 0 Span 1" (MASTER
group=0,11
context=from-pstn
switchtype = euroisdn
signalling = pri_cpe
channel => 1-15,17-31
context = default
group = 63
```

After confirming the system. conf and dahdi channels. conf files, execute the following command:

```
# dahdi_cfg  –vvvvvv
```

This command will read and load parameters from the file system.exe into the hardware. Part of the output result of the instruction execution is displayed as shown below.

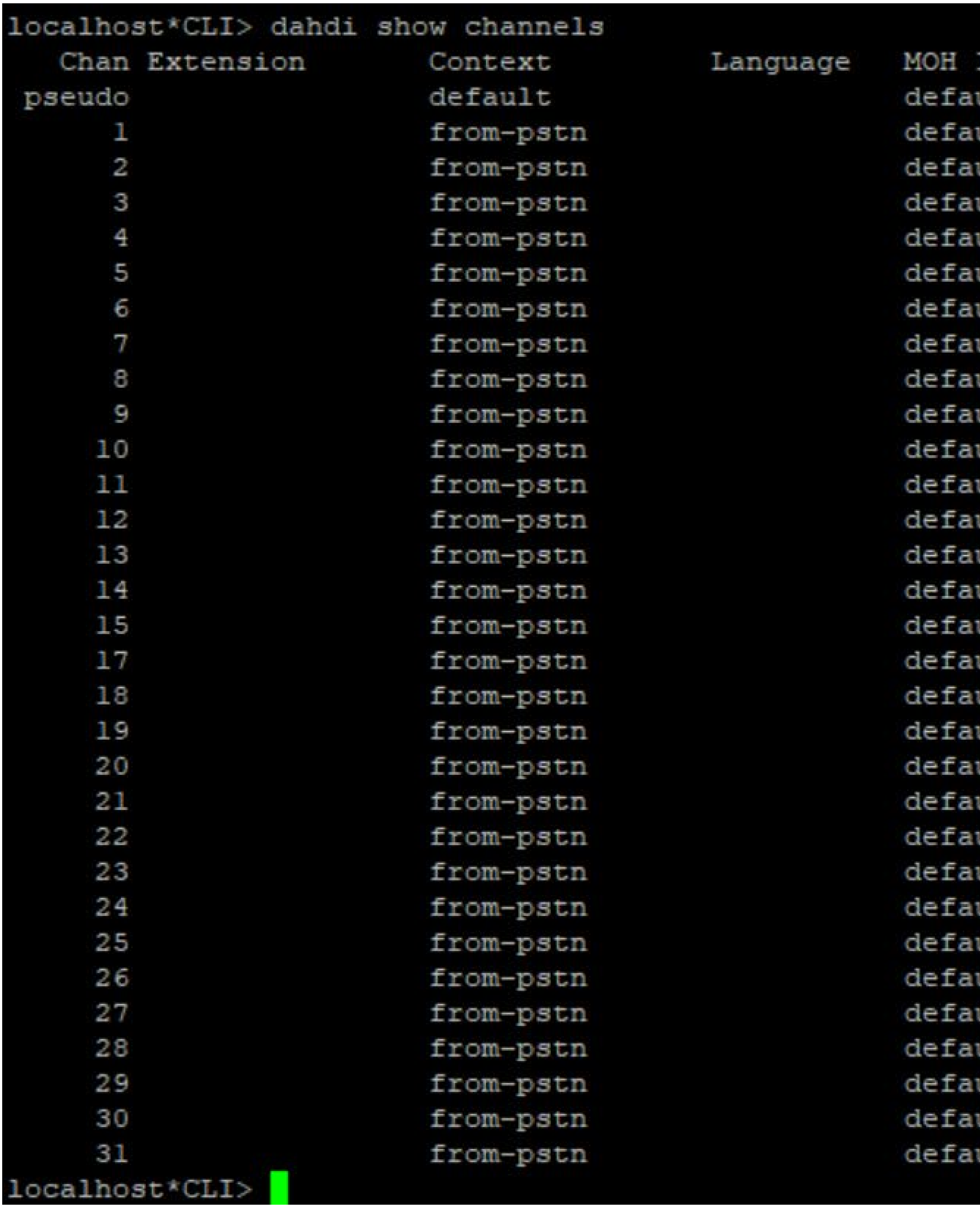


```
[root@localhost dahdi-linux-complete-3.2.0+3.2.0]#
[root@localhost dahdi-linux-complete-3.2.0+3.2.0]# dahd
DAHDI Tools Version - 3.2.0

DAHDI Version: 3.2.0
Echo Canceller(s): HWEC
Configuration
======================

SPAN 1: CCS/HDB3 Build-out: 0 db (CSU)/0-133 feet (DSX-

31 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
Setting echocan for channel 17 to none
Setting echocan for channel 18 to none
Setting echocan for channel 19 to none
Setting echocan for channel 20 to none
Setting echocan for channel 21 to none
Setting echocan for channel 22 to none
Setting echocan for channel 23 to none
Setting echocan for channel 24 to none
Setting echocan for channel 25 to none
Setting echocan for channel 26 to none
Setting echocan for channel 27 to none
Setting echocan for channel 28 to none
Setting echocan for channel 29 to none
Setting echocan for channel 30 to none
```

Start Asterisk

# asterisk -vvvvvgc

If Asterisk is already running, run 'asterisk-r' instead. In the CLI interface, please run:



```
localhost*CLI> dahdi show channels
   Chan Extension        Context         Language   MOH
 pseudo                  default                    defa
      1                  from-pstn                  defa
      2                  from-pstn                  defa
      3                  from-pstn                  defa
      4                  from-pstn                  defa
      5                  from-pstn                  defa
      6                  from-pstn                  defa
      7                  from-pstn                  defa
      8                  from-pstn                  defa
      9                  from-pstn                  defa
     10                  from-pstn                  defa
     11                  from-pstn                  defa
     12                  from-pstn                  defa
     13                  from-pstn                  defa
     14                  from-pstn                  defa
     15                  from-pstn                  defa
     17                  from-pstn                  defa
     18                  from-pstn                  defa
     19                  from-pstn                  defa
     20                  from-pstn                  defa
     21                  from-pstn                  defa
     22                  from-pstn                  defa
     23                  from-pstn                  defa
     24                  from-pstn                  defa
     25                  from-pstn                  defa
     26                  from-pstn                  defa
     27                  from-pstn                  defa
     28                  from-pstn                  defa
     29                  from-pstn                  defa
     30                  from-pstn                  defa
     31                  from-pstn                  defa
localhost*CLI>
```

If all channels display correctly, it means that DAHDI has been successfully loaded into Asterisk.

Next, please set the dialing plan.

Write a dialing plan

Please write a dial plan in the extensions.onf file. The following diagram illustrates a simple inbound and outbound plan:

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

When a call comes in from the E1 card, cc welcome voice will be played

```
[from-pstn]
exten => _X.,1,Answer()      ; answer the inbound call
exten => _X.,n,Playback(cc_welcome)
exten => _X.,n,Hangup()
```

When the extension is set to 'from internet', outgoing calls will be sent from the E1 card dahdi/g0, with ${INTERN} being the called number

```
[from-internal]
exten => _X.,1,Dial(dahdi/g0/${EXTEN})
exten => _X.,n,Hangup()
```

After setting up the dial plan, please run "asterisk-r" and execute the command "dialplan reload" in the CLI interface, and then you can dial