New Generation Systems (NGS) Linux Software User's Manual

Version 1.0, September 2019

www.moxa.com/product

New Generation Systems (NGS) Linux Software User's Manual

Copyright Notice

Trademarks

Disclaimer

Information in this document is subject to change without notice and does not represent a commitment on the part of Moxa.

Moxa provides this document as is, without warranty of any kind, either expressed or implied, including, but not limited to, its particular purpose. Moxa reserves the right to make improvements and/or changes to this manual, or to the products and/or the programs described in this manual, at any time.

Information provided in this manual is intended to be accurate and reliable. However, Moxa assumes no responsibility for its use, or for any infringements on the rights of third parties that may result from its use.

This product might include unintentional technical or typographical errors. Changes are periodically made to the information herein to correct such errors, and these changes are incorporated into new editions of the publication.

Technical Support Contact Information

www.moxa.com/support

Moxa Americas
Toll-free: 1-888-669-2872
Tel: +1-714-528-6777
Fax: +1-714-528-6778

Moxa China (Shanghai office)
Toll-free: 800-820-5036
Tel: +86-21-5258-9955
Fax: +86-21-5258-5505

Moxa Europe
Tel: +49-89-3 70 03 99-0
Fax: +49-89-3 70 03 99-99

Moxa Asia-Pacific
Tel: +886-2-8919-1230
Fax: +886-2-8919-1231

Moxa India
Tel: +91-80-4172-9088
Fax: +91-80-4132-1045

Table of Contents

1

Introduction

Thank you for purchasing a Moxa New Generation System (NGS) x86 ready-to-run embedded computer. This manual introduces the software configuration and management of the NGS Linux models, which run the Linux operating system.

Linux is an open, scalable operating system that allows you to build a wide range of innovative, small footprint devices. Software written for desktop PCs can be easily ported to the embedded computer with a GNU cross compiler and a minimum of source code modifications. A typical Linux-based device is designed for a specific use, and is often not connected to other computers, or a number of such devices connect to a centralized, front-end host.

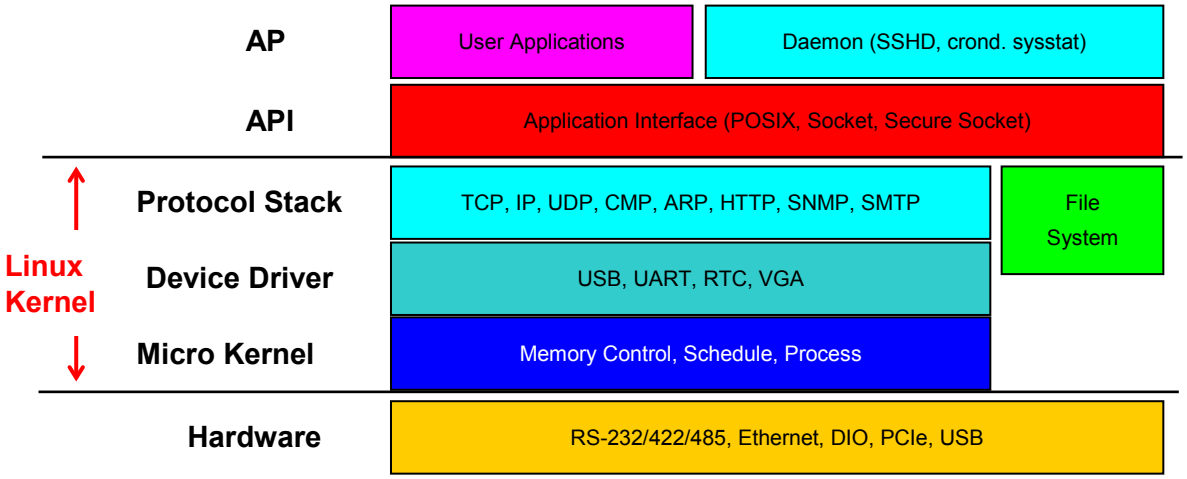
The NGS portfolio consists of the following series/models:

- DA-820C Series

The following topics are covered in this chapter:

Software Specifications

The Linux operating system pre-installed on the NGS embedded computers is the **Debian 9 "Stretch"** distribution. The Debian project involves a worldwide group of volunteers who endeavor to produce an operating system distribution composed entirely of free software. The Debian GNU/Linux follows the standard Linux architecture, making it easy to use programs that meet the POSIX standard. Program porting can be done with the GNU Tool Chain provided by Moxa. In addition to Standard POSIX APIs, device drivers for Moxa UART and other special peripherals are also included. An example software architecture is shown below:



ATTENTION

Refer to http://www.debian.org/ and http://www.gnu.org/ for information and documentation related to Debian GNU/Linux and the free software concept.

ATTENTION

The above software architecture is only an example. Different models or different build revisions of the Linux operating system may include components not shown in the above graphic.

Software Components

The NGS Linux models are preinstalled with the Debian 9 Stretch Linux distribution. For a list of the software components, refer to the chapter for the series.

2

Software Configuration

In this chapter, we explain how to operate a NGS Linux model computer directly from your desktop. There are three ways to connect to the NGS Linux model computer: through a DVI/DP monitor, via with an SSH over the network console from a Windows or Linux machine. This chapter describes basic Linux operating system configurations. Advanced network management and configuration instructions will be described in the next chapter, "Managing Communications."

The following topics are covered in this chapter:

- **Account Management**
- **Setting Up a Desktop Environment**
- **Starting from a VGA Console**
- **Connecting from an SSH Console**
  - Windows Users
  - Linux Users
- **Adjusting the System Time**
  - Setting the Time Manually
  - NTP Client or system-timesyncd Service
  - Manage the Service by systemd Script
- **Cron—Daemon for Executing Scheduled Commands**
- **Inserting a USB Storage Device into the Computer**
- **Checking the Linux Version**
- **Checking Moxa's Control Version**
- **APT—Installing and Removing Package**

Account Management

Connect the NGS to a display and turn on the computer. Enter the following information to log in the computer.

Login: moxa
Password: moxa

For security concern, we already disable the root account. We strongly suggest change the password at the first login. After successfully logging, provide the new password.

```
Using username "moxa".
Linux Moxa 4.9.0-6-amd64 #1 SMP Debian 4.9.88-1 (2018-04-29) x86_64

    ####       ####    ######    ####### ######     ##
     ###      ####   ###    ###    ####   ####     ###
      ###     ###   ###     ###    ###    ##      ###
      ###    ####   ##       ##    ###   #        ####
      ####    # ##  ###       ###    ### ##       ## ##
     ## ##    # ##  ###        ##     ####        #  ##
     ## ###  ## ##  ##         ##     ####        #  ###
     ##  ##  #  ##  ##         ##      ###       #######
     ##  ##  #  ##  ###       ###     #####      #    ##
     ##   ###   ##  ###       ###    ##  ###     #    ###
     ##   ###   ##   ##       ##     ##    ###    ##    ##
     ##   ###   ##    ##     ##     #     ###    #     ##
    ######  #  ######   ########   ####### ###########  ######

For further information check:
http://www.moxa.com/

You have mail.
Last login: Wed Mar  6 00:10:56 2019 from 10.144.54.91
You are using Moxa embedded computer.
Please change the default password in consideration of higher security level or
disable the default user, moxa.
moxa@Moxa:~$
```

When you finish changing the password, remember to type **sudo** each time you want to run commands with the privilege as the root. For example, typing **sudo ifconfig enp0s31f6 192.168.100.100** will allow you to configure the IP address of the LAN 1 port.

```
moxa@Moxa:~$ sudo ifconfig enp0s31f6 192.168.100.100
moxa@Moxa:~$ sudo  ifconfig enp0s31f6
enp0s31f6: flags=4099<UP,BROADCAST,MULTICAST>  mtu 1500
       inet 192.168.100.100  netmask 255.255.255.0  broadcast 192.168.100.255
       ether 00:90:e8:00:d7:38  txqueuelen 1000  (Ethernet)
       RX packets 0  bytes 0 (0.0 B)
       RX errors 0  dropped 0  overruns 0  frame 0
       TX packets 0  bytes 0 (0.0 B)
       TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
       device memory 0xb1300000-b137ffff
```

In addition, use **sudo –i** to login as root to have more privileges.

```
moxa@Moxa:~# sudo -i
[sudo] password for moxa:
root@Moxa:~$
```

For security concern, the system has enforced a limitation on the number of consecutive invalid access attempts by a user during a set time period based on applicable organizational policy. The configuration has applied in /etc/pam.d/common-auth:

The system default configures to lock out the user in 120 seconds for a user over 3 times of the unsuccessful login attempts. Unlock the user after time is up. The configure is located in /etc/pam.d/common-auth

```
...
auth required pam_tally2.so file=/var/log/tallylog deny=3 even_deny_root
unlock_time=120
```

The system default has configured a stronger password strength checking based on minimum length (4-16) and variety of character types by libpam-cracklib module. The configuration is in /etc/pam.d/common-password.

```
...
password       requisite                    pam_cracklib.so retry=3 difok=3
dcredit=-1 lcredit=-1 minlen=12 ocredit=-1 ucredit=-1
password       [success=1 default=ignore]     pam_unix.so obscure use_authtok
try_first_pass sha512
...
```

Setting Up a Desktop Environment

This section introduces the desktop environment setup for NGS series. The NGS Linux operating system default doesn't install a desktop environment. Debian supports all kinds of fully-featured graphical environment, such as, Gnome, KDE, lighter environment like Xfce and LXDE. User can choose to install one of these desktop systems in NGS. You can use these commands to install a desktop:

To install Gnome:

```
moxa@Moxa:~# sudo apt-get install task-gnome-desktop
```

To install KDE:

```
moxa@Moxa:~# sudo apt-get install aptitude tasksel
moxa@Moxa:~# sudo aptitude install ~t^desktop$ ~t^kde-desktop$
```

To install Xfce:

```
moxa@Moxa:~# sudo apt-get install xfce4 xfce4-goodies task-xfce-desktop
```

To install the minimum LXDE:

```
moxa@Moxa:~# sudo apt-get install lxde-core lxde
```

Starting from a VGA Console

Connect the display monitor to the NGS computers connector, and then power it up by connecting it to the power adaptor. It takes approximately 30 to 60 seconds for the system to boot up. Once the system is ready, a login screen will appear on your monitor.

To log in, type the login name and password as requested. The default values are both **moxa**.

Login: moxa
Password: moxa

```
Debian GNU/Linux 9 Moxa tty1

Moxa login: moxa
```

```
Password:
Linux Moxa 4.9.0-6-amd64 #1 SMP Debian 4.9.88-1 (2018-04-29) x86_64

   ####       ####    ######    ####### ######     ##
    ###      ####   ###    ###    ####   ####     ###
     ###     ###   ###     ###    ###    ##      ###
     ###    ####   ##       ##    ###   #       ####
     ####    # ##  ###       ###    ### ##       ## ##
    ## ##    # ##  ###        ##     ####        #  ##
    ## ###  ## ##  ##         ##     ####        #  ###
    ##  ##  #  ##  ##         ##     ###       #######
    ##  ##  #  ##  ###       ###     #####      #    ##
    ##   ###   ##  ###       ###    ##  ###     #    ###
    ##   ###   ##   ##       ##     ##    ###    ##    ##
    ##   ###   ##    ##     ##     #     ###    #     ##
   ######  #  ######   ########   ####### ###########  ######

For further information check:
http://www.moxa.com/

Last login: Wed Mar  6 00:10:56 2019 from 10.144.54.91
You are using Moxa embedded computer.
Please change the default password in consideration of higher security level or
disable the default user, moxa.
moxa@Moxa:~$
```

Connecting from an SSH Console

The NGS computers support the SSH console to offer users better network security compared to Telnet. The default IP addresses and netmasks of the network interfaces are as follows:

	Default IP Address	**Netmask**
LAN 1	192.168.3.127	255.255.255.0
LAN 2	192.168.4.127	255.255.255.0
LAN 3	192.168.5.127	255.255.255.0
LAN 4	192.168.6.127	255.255.255.0

Before using the ssh client, you should change the IP address of your development workstation so that the network ports are on the same subnet as the IP address for the LAN port that you will connect to. For example, if you will connect to LAN1, you could set your PC's IP address to 192.168.3.126, and the netmask to 255.255.255.0. If you will connect to LAN2, you could set your PC's IP address to 192.168.4.126, and the netmask to 255.255.255.0.

Use a cross-over Ethernet cable to connect your development workstation directly to the target computer, or use a straight-through Ethernet cable to connect the computer to a LAN hub or switch. Next, use a SSH client on your development workstation to connect to the target computer. After a connection has been established, type the login name and password as requested to log on to the computer. The default values are both **moxa**.

Login: moxa
Password: moxa

ATTENTION

For security concern, the system has applied to automatic logout if the ssh remote console or serial console is inactive within 5 minutes. If you don't need it in developing phase, you can comment out these configures.

The sshd automatic logout is configured in /etc/ssh/sshd_config

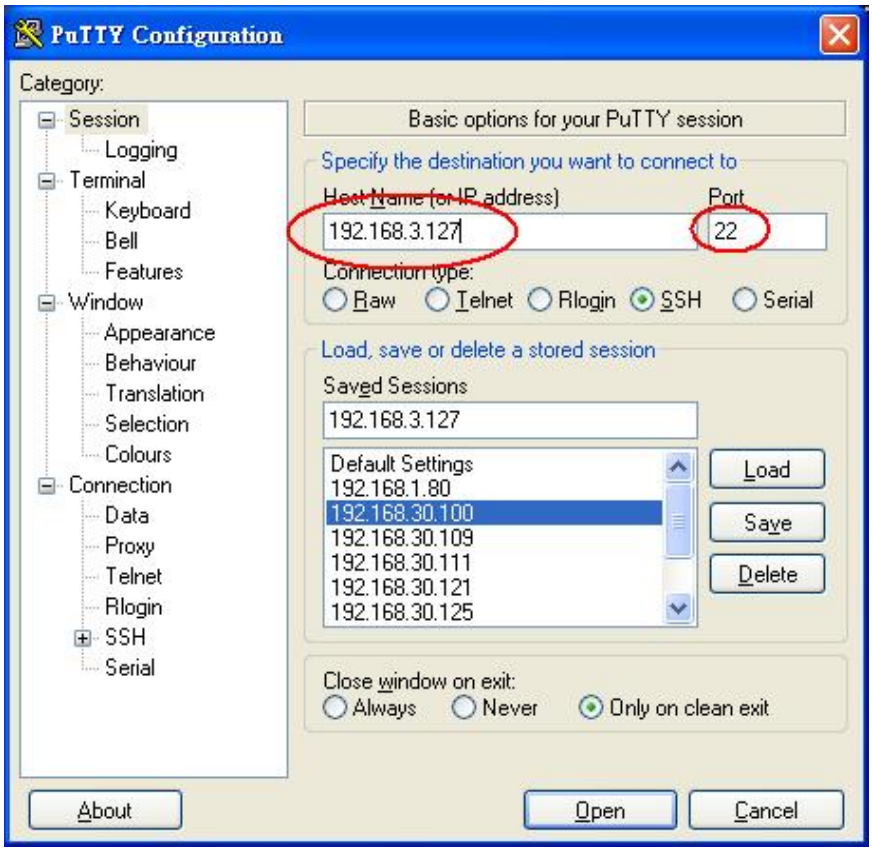
...
LoginGraceTime 120
PermitRootLogin without-password
StrictModes yes

The console automatic logout is configured in /etc/profile.d/moxa.sh

TMOUT=300
Export TMOUT

Windows Users

Click on the link http://www.chiark.greenend.org.uk/~sgtatham/putty/download.html to download **PuTTY** (free software) to set up an SSH console for the NGS in a Windows environment. The following screen shows an example of the configuration that is required.



Linux Users

From a Linux machine, use the **ssh** command to access the NGS device's console utility via SSH.

#ssh moxa@192.168.3.127

Select **yes** to open the connection.

```
[root@Jim_notebook root]# ssh 192.168.3.127
The authenticity of host '192.168.3.127 (192.168.3.127)' can't be established.
RSA key fingerprint is 8b:ee:ff:84:41:25:fc:cd:2a:f2:92:8f:cb:1f:6b:2f.
Are you sure you want to continue connection (yes/no)? yes_
```

Adjusting the System Time

The NGS has two time settings. One is the system time, and the other is provided by an RTC (Real Time Clock) built into the NGS hardware.

Setting the Time Manually

Use the **date** command to query the current system time or to set a new system time. Use **hwclock** to query the current RTC time or to set a new RTC time.

Use the following command to set the system time.

 # date MMDDhhmmYYYY

MM: Month
DD: Date
hhmm: Hour and Minute
YYYY: Year

Use the following command to write the current system time to the RTC.

 # hwclock –w

```
root@Moxa:/home/moxa# date
Wed Mar  6 19:33:51 CST 2019
root@Moxa:/home/moxa# hwclock
2019-03-06 19:33:57.482903+0800
root@Moxa:/home/moxa# date 030619352019.30
Wed Mar  6 19:35:30 CST 2019
root@Moxa:/home/moxa# hwclock -w
root@Moxa:/home/moxa# date; hwclock
Wed Mar  6 19:35:34 CST 2019
2019-03-06 19:35:34.061120+0800
```

NTP Client or system-timesyncd Service

The NGS can use a NTP (Network Time Protocol) client to initialize a time request to a remote NTP server. Use the **`ntpdate`** command to update the system time. Make sure that the device is connected to an Ethernet network before you run the **`ntpdate`** command.

 # ntpdate time.stdtime.gov.tw

 # hwclock –w

Visit http://www.ntp.org for more information about NTP and NTP server addresses.

```
root@Moxa:/home/moxa# ntpdate time.stdtime.gov.tw
 6 Mar 19:36:21 ntpdate[1172]: adjust time server 118.163.81.61 offset -0.000877
sec
root@Moxa:/home/moxa# hwclock -w
root@Moxa:/home/moxa# date; hwclock
Wed Mar  6 19:36:50 CST 2019
2019-03-06 19:36:50.154796+0800
```

The NGS has a built-in system-timesyncd that is used for Network Time Synchronization. This service default is enabled.

```
root@Moxa:/home/moxa# systemctl status systemd-timesyncd
● systemd-timesyncd.service - Network Time Synchronization
   Loaded: loaded (/lib/systemd/system/systemd-timesyncd.service; enabled; vendor
preset: enabled)
  Drop-In: /lib/systemd/system/systemd-timesyncd.service.d
           └─disable-with-time-daemon.conf
   Active: active (running) since Wed 2019-03-06 19:30:32 CST; 7min ago
     Docs: man:systemd-timesyncd.service(8)
 Main PID: 274 (systemd-timesyn)
   Status: "Synchronized to time server 103.18.128.60:123
(2.debian.pool.ntp.org)."
    Tasks: 2 (limit: 4915)
   CGroup: /system.slice/systemd-timesyncd.service
           └─274 /lib/systemd/systemd-timesyncd

Mar 06 19:30:31 Moxa systemd[1]: Starting Network Time Synchronization...
Mar 06 19:30:32 Moxa systemd[1]: Started Network Time Synchronization.
Mar 06 19:31:02 Moxa systemd-timesyncd[274]: Synchronized to time server
103.18.128.60:123 (2.debian.pool.ntp.org).
```

ATTENTION

Before using the NTP client utility, check your IP address and network settings (gateway and DNS) to make sure an Internet connection is available.

Manage the Service by systemd Script

Linux services can be started or stopped using system script. If you want to start up some service, you can use systemctl to enable or disable the service.

You can follow this example to add or remove your service in the system. First, you should write a system service unit. This example creates a systemd service unit at /etc/systemd/system/networking-check.service

```
[Unit]
After=snmpd.service

[Service]
ExecStart=/usr/local/bin/networking-check.sh

[Install]
WantedBy=default.target
```

- After: Instructs systemd on when the script should be run. In our case the script will run after snmpd.service has started.
- ExecStart: This field provides a full path the actual script to be execute
- WantedBy: Into what boot target the systemd unit should be installed

This is minimum example of a system script. More information please check systemd.serviceNext create /usr/local/bin/networking-check.sh script to check the network status. This example will ping a global DNS server to check if network is available and write the results in /var/log/networking-check.log.

```
moxa@Moxa:~# sudo vi /usr/local/bin/networking-check.sh
#!/bin/sh

while [ 1 ]; do
    date >> /var/log/networking-check.log
    ping -q -w 1 8.8.8.8
    if [ $? -eq  0 ]; then
        echo "Network is available" >> /var/log/networking-check.log
    else
        echo "Network is not available" >> /var/log/networking-check.log
    fi
    sleep 1
done
```

Before we launch this service, we need to make this script executable:

```
root@Moxa:~# chmod a+x /usr/local/bin/networking-check.sh
```

Then we can start the networking-check service by this command

```
root@Moxa:~# systemctl start networking-check
```

The networking-check.sh should launch in the background.

```
root@Moxa:~# ps aux|grep networking-check
root     2260  0.0  0.0   4288  1500 ?       Ss   14:49   0:00 /bin/sh
/usr/local/bin/networking-check.sh
root     2276  0.0  0.0  12784   980 pts/0    S+   14:49   0:00 grep networking-
check
```

/var/log/networking-check.log should be created.

```
root@Moxa:~# cat /var/log/networking-check.log
Wed Mar 14 14:49:09 EDT 2018
Network is available
...
```

Remember use this command to stop this service to prevent the log of this example occupied too much disk space.

```
root@Moxa:~# systemctl stop networking-check
```

Finally, you can enable this service at boot time by this command and reboot the system.

```
root@Moxa:~# systemctl enable networking-check
root@Moxa:~# reboot
```

To disable this service by the systemctl disable command.

```
root@Moxa:~# systemctl disable networking-check
```

Cron—Daemon for Executing Scheduled Commands

The Cron daemon will search **/etc/crontab** for crontab files.

Cron wakes up every minute and checks each command to see if it should be run at that time. When executing commands, output is mailed to the owner of the **crontab** (or to the user named in the MAILTO environment variable in the **crontab**, if such a user exists).

Modify the file **/etc/crontab** to set up your scheduled applications. **Crontab** files have the following format:

Mm	h	Dom	mon	Dow	user	command
minute	hour	Date	month	Week	user	command
0-59	0-23	1-31	1-12	0-6 (0 is Sunday)		

For example, issue the following command if you want to launch a program at 8:00 every day:

```
#minute hour  date  month  week  user   command
*        8     *     *      *     root   /path/to/your/program
```

The following example demonstrates how to use **Cron** to update the system time and RTC time every day at 8:00.

1. Write a shell script named fixtime.sh and save it to **/home/**.

   #!/bin/sh
   ntpdate time.stdtime.gov.tw
   hwclock –w
   exit 0

2. Change the mode of **fixtime.sh**

   # chmod 755 fixtime.sh

3. Modify the **/etc/crontab** file to run fixtime.sh at 8:00 every day.

   Add the following line to the end of crontab:

   *** 8 * * *root /home/fixtime.sh**

Inserting a USB Storage Device into the Computer

This system doesn't support auto mounting a USB storage devices automatically. In Linux system, you should mount it manually. Before mounting the USB storage, you should check the USB storage name by dmesg.

```
root@Moxa:~# dmesg
...
[  564.751226] sd 6:0:0:0: Attached scsi generic sg1 type 0
[  564.752400] sd 6:0:0:0: [sdb] 3973118 512-byte logical blocks: (2.03 GB/1.89
GiB)
[  564.753008] sd 6:0:0:0: [sdb] Write Protect is off
[  564.753013] sd 6:0:0:0: [sdb] Mode Sense: 03 00 00 00
[  564.753674] sd 6:0:0:0: [sdb] No Caching mode page found
[  564.753797] sd 6:0:0:0: [sdb] Assuming drive cache: write through
[  564.759333]  sdb: sdb1
[  564.762273] sd 6:0:0:0: [sdb] Attached SCSI removable disk
```

Or check /proc/partitions

```
root@Moxa:~# cat /proc/partitions
major minor  #blocks  name
   8       0    7824600 sda
   8       1    7823576 sda1
   8      16    1986559 sdb
   8      17    1985535 sdb1
```

Mount the USB storage partition 1, /dev/sdb1, on /mnt.

```
root:~# mount -t vfat /dev/sdb1 /mnt
```

Mount the USB storage partition 1, /dev/sdb1, on /mnt.

```
root:~# mount
...
/dev/sdb1 on /mnt type vfat
(rw,relatime,fmask=0022,dmask=0022,codepage=437,iocharset=ascii,shortname=mixed,utf8,errors=
ro)
```

If you want to automatic mount the USB storage at boot time, you can add it in /etc/fstab

```
...
LABEL=root   /            ext4    noatime,errors=remount-ro 0      1
#usbfs         /proc/bus/usb   usbfs   defaults 0      0
/dev/sdb1      /mnt    vfat    defaults       0      0
```

ATTENTION

Remember to type the command # **sync** before you disconnect the USB storage device to avoid losing data.

Checking the Linux Version

The program **uname**, which stands for "UNIX Name" and is part of the UNIX operating system, prints the name, version, and other details about the operating system running on the computer. Use the -**a** option to generate a response similar to the one shown below:

```
moxa@Moxa:~$ uname -a
Linux Moxa 4.9.0-6-amd64 #1 SMP Debian 4.9.88-1 (2018-04-29) x86_64 GNU/Linux
```

Checking Moxa's Control Version

The program **kversion** determines the Linux system released version controlled by Moxa. Use the -**a** option to check the built date. The built date format is YYYYMMDDHHmm. You can use kversion to check the released image version in troubleshooting. This example shows the built date, 201901091804, means it was built at 2019/01/09 18:04.

According to EPPROM information on IO board, use the -s option to show the PCBA serial number, and -t option to show PCBA type number(00-CPU Board, 01-Carrier Board, 02-Riser card), and -v option to show PCBA version.

For DA-820C's example:

```
root@Moxa:/home/moxa# kversion -h
Usage:
      kversion [OPTIONS]
Options:
      -a, --all             show firmware version and build date
      -s, --serial          show PCBA serial number
      -t, --type            show PCBA type number(00-CPU Board, 01-Carrier Board,
02-Riser card)
      -v, --version         show PCBA version
      -h, --help            show this help page
root@Moxa:/home/moxa# kversion -a
DA-820C firmware version v1.0.0 build 201901091804
root@Moxa:/home/moxa# kversion -s
```

```
0
root@Moxa:/home/moxa# kversion -t
01
root@Moxa:/home/moxa# kversion -v
0
```

APT—Installing and Removing Package

APT is the Debian tool used to install and remove packages. Before installing a package, you need to configure the apt source file, **/etc/apt/sources.list**.

Use vi editor to configure **/etc/apt/sources.list**.

```
deb mirror://debian.moxa.com/debian/mirrors stretch main contrib non-free

deb  http://deb.debian.org/debian stretch main contrib non-free
#deb-src  http://deb.debian.org/debian stretch main contrib non-free

deb  http://deb.debian.org/debian stretch-updates main contrib non-free
#deb-src  http://deb.debian.org/debian stretch-updates main contrib non-free

deb  http://deb.debian.org/debian stretch-backports main contrib non-free
#deb-src  http://deb.debian.org/debian stretch-backports main contrib non-free

deb  http://security.debian.org/ stretch/updates main contrib non-free
#deb-src  http://security.debian.org/ stretch/updates main contrib non-free
```

1. Update the source list after you configure it.

   ```
   moxa@MOXA:~# sudo apt-get update
   ```

2. Once you indicate which package you want to install (**ipsec-tools**, for example), type:

   ```
   moxa@MOXA:~# sudo apt-get install ipsec-tools
   ```

3. Use one of the following commands to remove a package:

   a. For a simple package removal:

      ```
      moxa@MOXA:~# sudo apt-get remove ipsec-tools
      ```

   b. For a complete package removal:

      ```
      moxa@MOXA:~# sudo apt-get remove ipsec-tools --purge
      ```

ATTENTION

You can free up the cache space with the command **# apt-get clean**.

```
moxa@MOXA:~# apt-get clean
```

3

Managing Communications

The NGS ready-to-run embedded computer is a network-centric platform designed to serve as a front-end for data acquisition and industrial control applications. This chapter describes how to configure the various communication functions supported by the Linux operating system.

The following topics are covered in this chapter:

- **Changing the interfaces Configuration File**
  - Adjusting IP Addresses with "ifconfig"
- **DNS Client**
  - /etc/hostname
  - /etc/resolv.conf
  - /etc/nsswitch.conf
- **Configuring Ethernet Bonding**
- **IPTABLES**
  - IPTABLES Hierarchy
  - IPTABLES Modules
  - Observe and Erase Chain Rules
  - Define Policy for Chain Rules
  - Append or Delete Rules
- **NAT (Network Address Translation)**
  - NAT Example
  - Enabling NAT at Bootup
- **PPP (Point-to-point Protocol)**
  - Connecting to a PPP Server over a Simple Dial-up Connection
  - Connecting to a PPP Server over a Hard-wired Link
  - Checking the Connection
  - Setting up a Machine for Incoming PPP Connections
- **PPPoE**
- **NFS (Network File System) Client**
- **SNMP (Simple Network Management Protocol)**
- **OpenVPN**
  - Install the OpenVPN
  - Ethernet Bridging for Private Networks on Different Subnets
  - Ethernet Bridging for Private Networks on the Same Subnet
  - Routed IP

Changing the interfaces Configuration File

1. Type cd /etc/network to change directories.

```
root@Moxa:~# cd /etc/network
```

2. Type **vi interfaces** to edit the network configuration file with **vi** editor. You can configure NGS computer Ethernet ports for static or dynamic (DHCP) IP addresses.

```
root@Moxa:~#/etc/network# vi interfaces
```

Static IP Address

As shown in the following example, the default static IP addresses can be modified.

```
# The loopback network interface
auto lo
iface lo inet loopback

# The primary network interface
auto enp0s31f6
allow-hotplug enp0s31f6
iface enp0s31f6 inet static
       address 192.168.3.127
       netmask 255.255.255.0
       broadcast 192.168.3.255

auto enp9s0
allow-hotplug enp9s0
iface enp9s0 inet static
       address 192.168.4.127
       netmask 255.255.255.0
       broadcast 192.168.4.255

auto enp10s0
allow-hotplug enp10s0
iface enp10s0 inet static
       address 192.168.5.127
       netmask 255.255.255.0
       broadcast 192.168.5.255

auto enp11s0
allow-hotplug enp11s0
iface enp11s0 inet static
       address 192.168.6.127
       netmask 255.255.255.0
       broadcast 192.168.6.255
```

Dynamic IP Address using DHCP

To configure one or both LAN ports to request an IP address dynamically, replace **static** with **dhcp** and then delete the rest of the lines.

```
# The primary network interface
auto enp0s31f6
iface enp0s31f6 inet dhcp
```

After modifying the boot settings of the LAN interface, issue the following command to activate the LAN settings immediately.

 # /etc/init.d/networking restart

```
moxa@Moxa:~# sudo service networking restart
```

Adjusting IP Addresses with "ifconfig"

IP settings can be adjusted during run-time, but the new settings will not be saved to the flash ROM without modifying the file **/etc/network/interfaces**. For example, type the command **# ifconfig enp0s31f6 192.168.1.1** to change the IP address of LAN1 to 192.168.1.1.

```
moxa@Moxa:~# sudo ifconfig enp0s31f6 192.168.1.1
```

DNS Client

The NGS computer supports DNS client (but not DNS server). To set up DNS client, you need to edit three configuration files: **/etc/hostname, /etc/resolv.conf,** and **/etc/nsswitch.conf**.

/etc/hostname

1. Edit /etc/hostname:

```
moxa@Moxa:~# sudo vi /etc/hostname
MOXA
```

2. Rboot the hostname.

```
moxa@Moxa:~# sudo reboot
```

3. Check the new hostname.

```
moxa@Moxa:~# sudo hostname
```

/etc/resolv.conf

This is the most important file that you need to edit when using DNS. For example, before using **# ntpdate time.stdtime.gov.tw** to update the system time, you will need to add the DNS server address to the file. Ask your network administrator which DNS server address you should use. The DNS server's IP address is specified with the **nameserver** command. For example, add the following line to /etc/resolv.conf (assuming the DNS server's IP address is 8.8.8.8): **nameserver 8.8.8.8**

```
moxa@Moxa:/etc# cat resolv.conf
#
# resolv.conf  This file is the resolver configuration file
# See resolver(5).
#
#nameserver 192.168.1.16

nameserver 8.8.8.8
nameserver 8.8.8.4
nameserver 168.95.1.1
```

/etc/nsswitch.conf

This file defines the sequence of files, **/etc/hosts** or **/etc/resolv.conf**, to be read to resolve the IP address. The **hosts** line in **/etc/nsswitch.conf** means use **/etc/host** first and DNS service to resolve the address.

```
# /etc/nsswitch.conf
#
# Example configuration of GNU Name Service Switch functionality.
# If you have the `glibc-doc-reference' and `info' packages installed, try:
# `info libc "Name Service Switch"' for information about this file.

passwd:         compat
group:          compat
shadow:         compat

hosts:          files dns
networks:       files

protocols:      db files
services:       db files
ethers:         db files
rpc:            db files

netgroup:       nis
```

Configuring Ethernet Bonding

The Linux bonding driver provides a method for aggregating multiple network interfaces into a single logical "bonded" interface. To use the bonding feature, you have to load the bonding driver with mode setting. Then use ifenslave to add the Ethernet interface into bond0 interface. Here is the script bonded enp9s0 and enp10s0 together, you can put it in **/etc/init.d/bonding.sh**.

```
#! /bin/bash

NAME=bonding
PATH=/bin:/usr/bin:/sbin:/usr/sbin
BONDING_IP=192.168.3.127

case "$1" in
  start)
    # to set ethX interfaces as slave the bond0 must have an ip
    echo "Starting bonding service: $NAME."
    modprobe bonding mode=1 miimon=100      # load bonding module

    ifdown enp10s0              # putting down LAN3
    ifdown enp9s0               # putting down LAN2

    ifconfig bond0 $BONDING_IP netmask 255.255.255.0 up # set ip address

    ifenslave bond0 enp10s0         # set LAN3 in slave for bond0
    ifenslave bond0 enp9s0          # set LAN2 in slave for bond0
    ;;

  stop)
    echo "Stopping bonding service: $NAME"
    ifenslave -d bond0 enp10s0          # release LAN3 from bond0
    ifenslave -d bond0 enp9s0           # release LAN2 from bond0

    ifconfig bond0 down             # putting down bond0
```

```
    modprobe -r bonding             # unload bonding module

    ifup enp10s0
    ifup enp9s0
    ;;

  restart)
    $0 stop
    $0 start $BONDING_IP
    ;;

  *)
    echo "Usage: systemctl {start|stop|restart} $NAME "
    exit 1
    ;;
esac

exit 0
```

Creates a systemd service unit at /etc/systemd/system/bonding.service for bonding the Ethernet service

```
[Unit]
Description=Bonding service

[Service]
Type=oneshot
ExecStart=/sbin/bonding.sh start
ExecStop=/sbin/bonding.sh stop
RemainAfterExit=yes

[Install]
WantedBy=default.target
```

To install it, use the following command.

```
moxa@Moxa:~# sudo systemctl enable bonding
```

To uninstall it, use this command.

```
moxa@Moxa:~# sudo systemctl disable bonding
```

IPTABLES

IPTABLES is an administrative tool for setting up, maintaining, and inspecting the Linux kernel's IP packet filter rule tables. Several different tables are defined, with each table containing built-in chains and user-defined chains.

Each chain is a list of rules that apply to a certain type of packet. Each rule specifies what to do with a matching packet. A rule (such as a jump to a user-defined chain in the same table) is called a **target**.

Linux supports three types of IPTABLES: Filter tables, NAT tables, and Mangle tables.

Filter Table—includes three chains:

- **INPUT chain**
- **OUTPUT chain**
- **FORWARD chain**

NAT Table—includes three chains:

- **PREROUTING chain**—transfers the destination IP address (DNAT).
- **POSTROUTING chain**—works after the routing process and before the Ethernet device process to transfer the source IP address (SNAT).
- **OUTPUT chain**—produces local packets.

Sub-tables

- **Source NAT (SNAT)**—changes the first source IP address of the packet.
- **Destination NAT (DNAT)**—changes the first destination IP address of the packet.
- **MASQUERADE**—a special form for SNAT. If one host can connect to the Internet, then the other computers that connect to this host can connect to the Internet when the computer does not have an actual IP address.
- **REDIRECT**—a special form of DNAT that re-sends packets to a local host independent of the destination IP address.

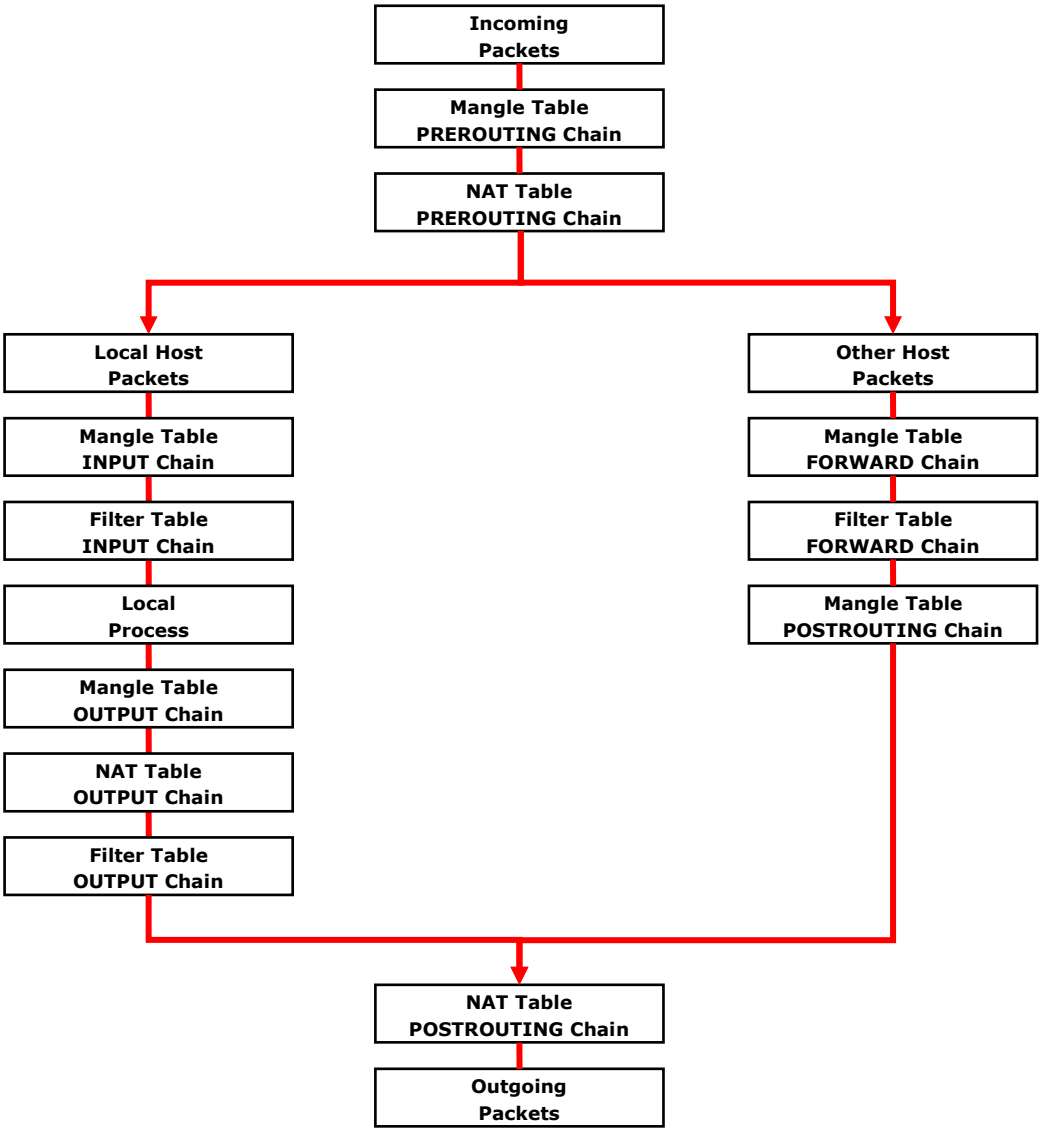
Mangle Table—includes two chains

- **PREROUTING chain**—pre-processes packets before the routing process.
- **OUTPUT chain**—processes packets after the routing process.

Mangle tables can have one of three extensions—TTL, MARK, TOS.

IPTABLES Hierarchy

The following figure shows the IPTABLES hierarchy.



IPTABLES Modules

The iptables supports the following sub-modules. Be sure to use the module that matches your application.

arptable_filter.ko	arp_tables.ko	arpt_mangle.ko	ip_conntrack_amanda.ko
ip_conntrack_ftp.ko	ip_conntrack_h323.ko	ip_conntrack_irc.ko	ip_conntrack.ko
ip_conntrack_netbios_ns.ko	ip_conntrack_netlink.ko	ip_conntrack_pptp.ko	ip_conntrack_proto_sctp.ko
ip_conntrack_sip.ko	ip_conntrack_tftp.ko	ip_nat_amanda.ko	ip_nat_ftp.ko
ip_nat_h323.ko	ip_nat_irc.ko	ip_nat.ko	ip_nat_pptp.ko
ip_nat_sip.ko	ip_nat_snmp_basic.ko	ip_nat_tftp.ko	ip_queue.ko
iptable_filter.ko	iptable_mangle.ko	iptable_nat.ko	iptable_raw.ko
ip_tables.ko	ipt_addrtype.ko	ipt_ah.ko	ipt_CLUSTERIP.ko
ipt_dscp.ko	ipt_DSCP.ko	ipt_ecn.ko	ipt_ECN.ko
ipt_hashlimit.ko	ipt_iprange.ko	ipt_LOG.ko	ipt_MASQUERADE.ko
ipt_NETMAP.ko	ipt_owner.ko	ipt_recent.ko	ipt_REDIRECT.ko
ipt_REJECT.ko	ipt_SAME.ko	ipt_TCPMSS.ko	ipt_tos.ko
ipt_TOS.ko	ipt_ttl.ko	ipt_TTL.ko	ipt_ULOG.ko

The basic syntax to enable and load an IPTABLES module is as follows:

```
# lsmod
# modprobe ip_tables
# modprobe iptable_filter
# modprobe iptable_mangle
# modprobe iptable_nat
```

Use **lsmod** to check if the **ip_tables** module has already been loaded in the NGS computer. Use **modprobe** to insert and enable the module.

Use **iptables**, **iptables-restore**, and **iptables-save** to maintain the database.

ATTENTION

IPTABLES plays the role of packet filtering or NAT. Be careful when setting up the IPTABLES rules. If the rules are not correct, remote hosts that connect via a LAN or PPP may be denied. We recommend using the VGA console to set up the IPTABLES. Click on the following links for more information about IPTABLES.

http://www.linuxguruz.com/iptables/
http://www.netfilter.org/documentation/HOWTO//packet-filtering-HOWTO.html

Since the IPTABLES command is very complex, to illustrate the IPTABLES syntax we have divided our discussion of the various rules into three categories: **Observe and erase chain rules**, **Define policy rules**, and **Append or delete rules**.

Observe and Erase Chain Rules

Usage:

 # iptables [-t tables] [-L] [-n]

-t tables: Table to manipulate (default: 'filter'); example: nat or filter.
-L [chain]: List all rules in selected chains. If no chain is selected, all chains are listed.
-n: Numeric output of addresses and ports.

 # iptables [-t tables] [-FXZ]

-F: Flush the selected chain (all the chains in the table if none is listed).
-X: Delete the specified user-defined chain.
-Z: Set the packet and byte counters in all chains to zero.

Example:

 # iptables -L -n

In this example, since we do not use the -t parameter, the system uses the default "filter" table. Three chains are included: INPUT, OUTPUT, and FORWARD. INPUT chains are accepted automatically, and all connections are accepted without being filtered.

 # iptables -F
 # iptables -X
 # iptables -Z

Define Policy for Chain Rules

Usage:

 # iptables [-t tables] [-P] [INPUT, OUTPUT, FORWARD, PREROUTING, OUTPUT, POSTROUTING] [ACCEPT, DROP]

-P: Set the policy for the chain to the given target.
INPUT: For packets coming into the device.
OUTPUT: For locally-generated packets.
FORWARD: For packets routed out through the device.
PREROUTING: To alter packets as soon as they come in.
POSTROUTING: To alter packets as they are about to be sent out.

Example:

#iptables -P INPUT DROP
#iptables -P OUTPUT ACCEPT
#iptables -P FORWARD ACCEPT
#iptables -t nat -P PREROUTING ACCEPT
#iptables -t nat -P OUTPUT ACCEPT
#iptables -t nat -P POSTROUTING ACCEPT

In this example, the policy accepts outgoing packets and denies incoming packets.

Append or Delete Rules

Usage:

 # iptables [-t table] [-AI] [INPUT, OUTPUT, FORWARD] [-io interface] [-p tcp, udp, icmp, all] [-s IP/network] [--sport ports] [-d IP/network] [--dport ports] -j [ACCEPT. DROP]

-A: Append one or more rules to the end of the selected chain.
-I: Insert one or more rules in the selected chain as the given rule number.
-i: Name of an interface via which a packet is going to be received.
-o: Name of an interface via which a packet is going to be sent.
-p: The protocol of the rule or of the packet to check.
-s: Source address (network name, host name, network IP address, or plain IP address).
--sport: Source port number.
-d: Destination address.
--dport: Destination port number.
-j: Jump target. Specifies the target of the rules; i.e., how to handle matched packets.

For example, ACCEPT the packet, DROP the packet, or LOG the packet.

Examples:

Example 1: Accept all packets from the lo interface.

 # iptables -A INPUT -i lo -j ACCEPT

Example 2: Accept TCP packets from 192.168.0.1.

 # iptables -A INPUT -i enp0s31f6 -p tcp -s 192.168.0.1 -j ACCEPT

Example 3: Accept TCP packets from Class C network 192.168.1.0/24.

 # iptables -A INPUT -i enp0s31f6 -p tcp -s 192.168.1.0/24 -j ACCEPT

Example 4: Drop TCP packets from 192.168.1.25.

 # iptables -A INPUT -i enp0s31f6 -p tcp -s 192.168.1.25 -j DROP

Example 5: Drop TCP packets addressed for port 21.

 # iptables -A INPUT -i enp0s31f6 -p tcp --dport 21 -j DROP

Example 6: Accept TCP packets from 192.168.0.24 to NGS computer's port 137, 138, 139

 # iptables -A INPUT -i enp0s31f6 -p tcp -s 192.168.0.24 --dport 137:139 -j ACCEPT

Example 7: Log TCP packets that visit NGS computer's port 25.

 # iptables -A INPUT -i enp0s31f6 -p tcp --dport 25 -j LOG

Example 8: Drop all packets from MAC address 01:02:03:04:05:06.

 # iptables -A INPUT -i enp0s31f6 -p all -m mac --mac-source 01:02:03:04:05:06 -j DROP

ATTENTION

In Example 8, remember to issue the command **# modprobe ipt_mac** first to load the module ipt_mac.

NAT (Network Address Translation)

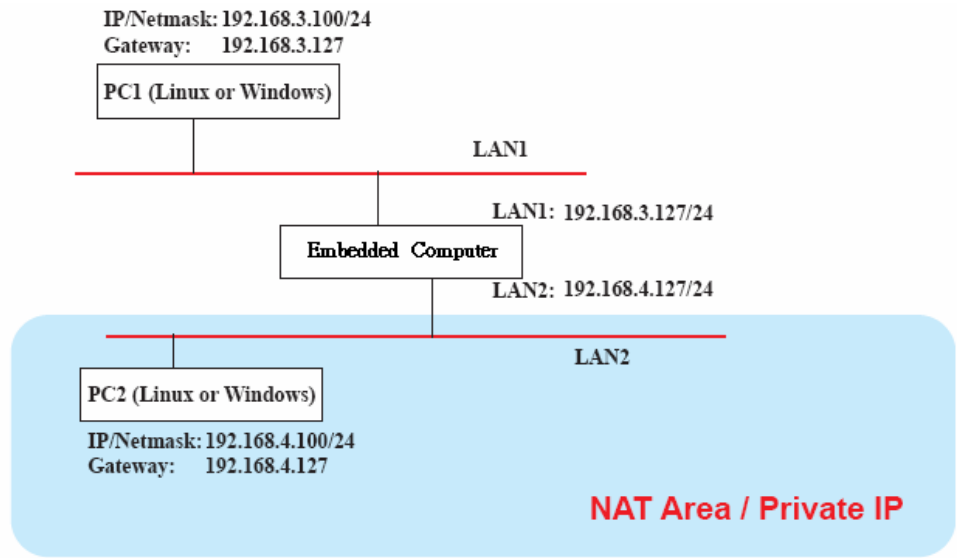
The NAT (Network Address Translation) protocol translates IP addresses used on one network into IP addresses used on a connecting network. One network is designated the inside network and the other is the outside network. Typically, the NGS computer connects several devices on a network and maps local inside network addresses to one or more global outside IP addresses, and un-maps the global IP addresses on incoming packets back into local IP addresses.

ATTENTION

Click the following link for more information on NAT:
http://www.netfilter.org/documentation/HOWTO//packet-filtering-HOWTO.html

NAT Example

The IP address of all packets leaving LAN1 are changed to **192.168.3.127** (you will need to load the module **ipt_MASQUERADE**):



Enabling NAT at Bootup

In most real world situations, you will want to use a simple shell script to enable NAT when the NGS computer boots up. The following script is an example.

```
#!/bin/bash
# If you put this shell script in the /home/nat.sh
# Remember to chmod 744 /home/nat.sh
# Edit the rc.local file to make this shell startup automatically.
# vi /etc/rc.local
# Add a line in the end of rc.local /home/nat.sh
EXIF= "enp0s31f6"  #This is an external interface for setting up a valid IP
address.
EXNET= "192.168.4.0/24"  #This is an internal network address.
# Step 1. Insert modules.
# Here 2> /dev/null means the standard error messages will be dump to null
device.
modprobe ip_tables  2> /dev/null
modprobe ip_nat_ftp  2> /dev/null
modprobe ip_nat_irc  2> /dev/null
modprobe ip_conntrack  2> /dev/null
modprobe ip_conntrack_ftp  2> /dev/null
modprobe ip_conntrack_irc  2> /dev/null
# Step 2. Define variables, enable routing and erase default rules.
PATH=/bin:/sbin:/usr/bin:/usr/sbin:/usr/local/bin:/usr/local/sbin
export PATH
echo "1" > /proc/sys/net/ipv4/ip_forward
/sbin/iptables -F
/sbin/iptables -X
/sbin/iptables -Z
/sbin/iptables -F -t nat
/sbin/iptables -X -t nat
/sbin/iptables -Z -t nat
/sbin/iptables -P INPUT ACCEPT
/sbin/iptables -P OUTPUT ACCEPT
/sbin/iptables -P FORWARD ACCEPT
/sbin/iptables -t nat -P PREROUTING ACCEPT
/sbin/iptables -t nat -P POSTROUTING ACCEPT
/sbin/iptables -t nat -P OUTPUT ACCEPT
# Step 3. Enable IP masquerade.
#ehco 1 > /proc/sys/net/ipv4/ip_forward#modprobe ipt_MASQUERADE#iptables –t nat –
A POSTROUTING –o enp0s31f6 –j MASQUERADE
```

PPP (Point-to-point Protocol)

PPP (Point to Point Protocol) is used to run IP (Internet Protocol) and other network protocols over a serial link. PPP can be used for direct serial connections (using a null-modem cable) over a Telnet link, and links established using a modem over a telephone line.

Modem/PPP access is almost identical to connecting directly to a network through the embedded computer Ethernet port. Since PPP is a peer-to-peer system, the Linux operating system can use PPP to link two networks (or a local network to the Internet) to create a Wide Area Network (WAN).

ATTENTION

Click on the following links for more information about PPP:

http://tldp.org/HOWTO/PPP-HOWTO/index.html
http://axion.physics.ubc.ca/ppp-linux.html

Connecting to a PPP Server over a Simple Dial-up Connection

The following command is used to connect to a PPP server by modem. Use this command for old ppp servers that prompt for a login name (replace "username" with the correct name) and password (replace "password" with the correct password).

```
#pppd connect 'chat -v "" ATDT5551212 CONNECT ""' ogin: username word: password'
/dev/ ttyM0 115200 debug crtscts modem defaultroute 192.1.1.17
```

Note that "debug crtscts" and "defaultroute 192.1.1.17" are optional.

If the PPP server does not prompt for the username and password, the command should be entered as follows (replace "username" with the correct username and replace "password" with the correct password):

```
#pppd connect 'chat -v "" ATDT5551212 CONNECT ""' user username password password
/dev/ ttyM0 115200 crtscts modem
```

The pppd options are described below:

connect 'chat etc...' This option gives the command to contact the PPP server. The **chat** program is used to dial a remote computer. The entire command is enclosed in single quotes because pppd expects a one-word argument for the **connect** option. The options for **chat** are given below:

-v verbose mode; log what we do to syslog

" " Double quotes—don't wait for a prompt, but instead do ... (note that you must include a space after the second quotation mark)

ATDT5551212 Dial the modem, and then ...

CONNECT Wait for an answer.

" " Send a return (null text followed by the usual return)

login: username, password: password
Log in with username and password.

Note: Refer to the chat man page, chat.8, for more information about the **chat** utility.

/dev/ Specify the callout serial port.

115200 The baud rate.

debug Log status in syslog.

crtscts Use hardware flow control between the computer and modem (at baudrate of 115200 this is a must).

modem Indicates that this is a modem device; pppd will hang up the phone before and after making the call.

defaultroute Once the PPP link is established, make it the default route; if you have a PPP link to the Internet, this is probably what you want.

192.1.1.17 This is a degenerate case of a general option of the form x.x.x.x:y.y.y.y. Here x.x.x.x is the local IP address and y.y.y.y is the IP address of the remote end of the PPP connection. If this option is not specified, or if just one side is specified, then

x.x.x.x defaults to the IP address associated with the local machine's hostname (located in /**etc/hosts**), and y.y.y.y is determined by the remote machine.

Connecting to a PPP Server over a Hard-wired Link

If a username and password are not required, use the following command (note that **noipdefault** is optional):

#pppd connect 'chat –v" " " " ' noipdefault /dev/ttyM0 19200 crtscts

If a username and password is required, use the following command (note that **noipdefault** is optional, and the username and password are both "root"):

#pppd connect 'chat –v" " " " ' user root password root noipdefault /dev/ttyM0 19200 crtscts

Checking the Connection

Once you have set up a PPP connection, there are some steps you can take to test the connection. First, type:
 # /sbin/ifconfig

Depending on your distribution, the command might be located elsewhere. After executing the command, you should be able to see all of the network interfaces that are UP.

ppp0 should be one of the network interfaces. You should recognize the first IP address as the IP address of the computer, and **P-t-P address** is the IP address of the server. The output should be similar to the following:

```
lo      Link encap Local Loopback
        inet addr 127.0.0.1   Bcast 127.255.255.255 Mask 255.0.0.0
        UP LOOPBACK RUNNING   MTU 2000   Metric 1
        RX packets 0 errors 0 dropped 0 overrun 0

ppp0    Link encap Point-to-Point Protocol
        inet addr 192.76.32.3   P-t-P 129.67.1.165 Mask 255.255.255.0
        UP POINTOPOINT RUNNING   MTU 1500   Metric 1
        RX packets 33 errors 0 dropped 0 overrun 0
        TX packets 42 errors 0 dropped 0 overrun 0
```

Now, type:
 # ping z.z.z.z

where z.z.z.z is the address of your name server. The output should be similar to the following:

```
MOXA:~# ping 129.67.1.165
PING 129.67.1.165 (129.67.1.165): 56 data bytes
64 bytes from 129.67.1.165: icmp_seq=0 ttl=225 time=268 ms
64 bytes from 129.67.1.165: icmp_seq=1 ttl=225 time=247 ms
64 bytes from 129.67.1.165: icmp_seq=2 ttl=225 time=266 ms
^C
--- 129.67.1.165 ping statistics ---
3 packets transmitted, 3 packets received, 0% packet loss
round-trip min/avg/max = 247/260/268 ms
MOXA:~#
```

Try typing:
 # netstat -nr

You should see three routes similar to the following:

```
Kernel routing table
Destination   Gateway   Genmask         Flags   Metric  Ref Use     iface
129.67.1.165  0.0.0.0   255.255.255.255 UH      0       0   6       ppp0
127.0.0.0     0.0.0.0   255.0.0.0       U       0       0   0       lo
0.0.0.0       129.67.1.165  0.0.0.0     UG      0       0   6298    ppp0
```

If your output looks similar but does not have the "destination 0.0.0.0" line (which refers to the default route used for connections), you may have run pppd without the **defaultroute** option. At this point, you can try using Telnet, ftp, or finger, bearing in mind that you will have to use numeric IP addresses unless you have configured **/etc/resolv.conf** correctly.

Setting up a Machine for Incoming PPP Connections

Method 1: pppd dial-in with pppd commands

This first example applies to using a modem, and requiring authorization with a username and password.

#pppd /dev/ttyM0 115200 crtscts modem 192.168.16.1:192.168.16.2 login auth

You should also add the following line to the file **/etc/ppp/pap-secrets**:

*** * "" ***

The first star (*) lets everyone login. The second star (*) lets every host connect. The pair of double quotation marks ("") indicates that the file **/etc/passwd** can be used to check the password. The last star (*) is to let any IP connect.

The following example does not check the username and password:

 # pppd/dev/ttyM0 115200 crtscts modem 192.168.16.1:192.168.16.2

Method 2: pppd dial-in with pppd script

Configure a dial-in script **/etc/ppp/peer/dialin**

```
# You usually need this if there is no PAP authentication
noauth
#auth
#login

# The chat script (be sure to edit that file, too!)
init "/usr/sbin/chat -v -f /etc/ppp/ppp-ttyMUE0.chat"

# Set up routing to go through this PPP link
defaultroute

# Default modem (you better replace this with /dev/ttySx!)
/dev/ttyM0

# Speed
115200

# Keep modem up even if connection fails
persist
crtscts
modem
192.168.16.1:192.168.16.2
debug
-detach
```

Configure the chat script **/etc/ppp/ppp-ttyM0.chat**

```
SAY     'Auto Answer ON\n'
''      ATS0=1
```

Start the **pppd** dial-in service.

```
# pppd call dialin
```

PPPoE

Use the following procedure to configure PPPoE:

1. Connect the NGS computer's LAN port to an ADSL modem with a cross-over cable, HUB, or switch.
2. Log in to the NGS computer as the root user.
3. Edit the file **/etc/ppp/chap-secrets** and add the following:
   "username@hinet.net" * "password" *

```
# Secrets for authentication using CHAP
# client        server  secret                  IP addresses

# PPPOE example, if you want to use it, you need to unmark it and modify it
"username@hinet.net"   *      "password"      *
```

   username@hinet.net is the username obtained from the ISP to log in to the ISP account. **password** is the corresponding password for the account.

4. Edit the file **/etc/ppp/pap-secrets** and add the following:
   "username@hinet.net" * "password" *

```
# ATTENTION: The definitions here can allow users to login without a
# password if you don't use the login option of pppd! The mgetty Debian
# package already provides this option; make sure you don't change that.

# INBOUND connections

# Every regular user can use PPP and has to use passwords from /etc/passwd
*       hostname        ""      *
"username@hinet.net"    *       "password"      *

# UserIDs that cannot use PPP at all. Check your /etc/passwd and add any
# other accounts that should not be able to use pppd!
guest   hostname        "*"     -
master  hostname        "*"     -
root    hostname        "*"     -
support hostname        "*"     -
stats   hostname        "*"     -

# OUTBOUND connections
```

   username@hinet.net is the username obtained from the ISP to log in to the ISP account. **password** is the corresponding password for the account.

5. Edit the file **/etc/ppp/options** and add the following line:
   plugin rp-pppoe

```
# received.  Note: it is not advisable to use this option with the persist
# option without the demand option.  If the active-filter option is given,
# data packets which are rejected by the specified activity filter also
# count as the link being idle.
#idle <n>
```

```
# Specifies how many seconds to wait before re-initiating the link after
# it terminates.  This option only has any effect if the persist or demand
# option is used.  The holdoff period is not applied if the link was
# terminated because it was idle.
#holdoff <n>

# Wait for up n milliseconds after the connect script finishes for a valid
# PPP packet from the peer.  At the end of this time, or when a valid PPP
# packet is received from the peer, pppd will commence negotiation by
# sending its first LCP packet.  The default value is 1000 (1 second).
# This wait period only applies if the connect or pty option is used.
#connect-delay <n>

# Load the pppoe plugin
plugin rp-pppoe.so

# ---<End of File>---
```

6. If you use LAN1 to connect to the ADSL modem, add the file **/etc/ppp/options.enp0s31f6**, if you use LAN2 to connect to the ADSL modem, add **/etc/ppp/options.enp9s0**, etc.

```
name username@hinet.net
mtu 1492
mru 1492
defaultroute
noipdefault
~
~
 "/etc/ppp/options.enp0s31f6" 5 lines, 67 characters
```

   Type your username (the one you set in the **/etc/ppp/pap-secrets and /etc/ppp/chap-secrets** files) after the **name** option. You may add other options as needed.

7. Set up DNS.

   If you are using DNS servers supplied by your ISP, edit the file **/etc/resolv.conf** by adding the following lines of code:

   nameserver ip_addr_of_first_dns_server
   nameserver ip_addr_of_second_dns_server

   For example:

   nameserver 168.95.1.1
   nameserver 139.175.10.20

```
moxa@Moxa:/etc# cat resolv.conf
#
# resolv.conf  This file is the resolver configuration file
# See resolver(5).
#
nameserver 168.95.1.1
nameserver 139.175.10.20
```

   Use the following command to create a **pppoe** connection:
   #pppd enp0s31f6

8. The ADSL modem is connected to the **LAN1** port, which is named **enp0s31f6**. If the ADSL modem is connected to **LAN2**, use **enp9s0**, etc.
9. Type **#ifconfig ppp0** to check if the connection is OK. If the connection is OK, you should see the IP address of ppp0. Use **#ping** to test the IP address.

```
ppp0    Link encap Point-to-Point Protocol
        inet addr 192.76.32.3   P-t-P 129.67.1.165 Mask 255.255.255.0
        UP POINTOPOINT RUNNING   MTU 1500   Metric 1
        RX packets 33 errors 0 dropped 0 overrun 0
        TX packets 42 errors 0 dropped 0 overrun 0
```

10. If you want to disconnect the connection, use the kill command to kill the **pppd** process.

NFS (Network File System) Client

The Network File System (NFS) is used to mount a disk partition on a remote machine (as if it were on a local hard drive), allowing fast and seamless sharing of files across a network. NFS allows users to develop applications for the NGS computer without worrying about the amount of disk space that will be available. The NGS computer only supports NFS client protocol.

ATTENTION

Click on the following links for more information about NFS.

http://www.ietf.org/rfc/rfc1213.txt
http://www.faqs.org/rfcs/rfc1317.html

The following procedures illustrate how to mount a remote NFS Server.

1. Scan the NFS Server's shared directory:
   #showmount -e HOST
   showmount: Shows the mount information of an NFS Server
   -e: Shows the NFS Server's export list.
   HOST: IP address or DNS address
2. Establish a mount point on the NFS Client site:
   #mkdir -p /home/nfs/public
3. Mount the remote directory to a local directory:
   # mount -t nfs -o nolock 192.168.3.100:/home/public /home/nfs/public
   (This is where 192.168.3.100 is the example IP address of the NFS server.)

SNMP (Simple Network Management Protocol)

The NGS computer comes with the SNMP v2c (Simple Network Management Protocol) software package. The snmpd service default is disabled. You can enable it by this command.

```
moxa@Moxa:~# sudo systemctl enable snmpd
```

The snmpd configure is located, /etc/snmp/snmpd.conf. If you want to support the SNMP service for all listening interface, you should remove the '#' and restart the snmpd service.

```
# /etc/snmp/snmpd.conf
# ...
#  Listen for connections from the local system only
# agentAddress  udp:127.0.0.1:161
#  Listen for connections on all interfaces (both IPv4 *and* IPv6)
# For security concern, we comment out this line. If you want to support SNMP on
all Ethernet Interfaces, please remove the '#' and restart the snmpd service.
# agentAddress udp:161,udp6:[::1]:161
```

Then restart the snmpd service.

```
moxa@Moxa:~# sudo systemctl restart snmpd
```

The following example shows an SNMP agent responding to a query from the SNMP browser on the host site:

```
root@Moxa:/home/moxa# snmpwalk -c public -v2c 127.0.0.1
iso.3.6.1.2.1.1.1.0 = STRING: "Linux Moxa 4.9.0-6-amd64 #1 SMP Debian 4.9.88-1
(2018-04-29) x86_64"
iso.3.6.1.2.1.1.2.0 = OID: iso.3.6.1.4.1.8072.3.2.10
iso.3.6.1.2.1.1.3.0 = Timeticks: (21934) 0:03:39.34
iso.3.6.1.2.1.1.4.0 = STRING: "Me <me@example.org>"
iso.3.6.1.2.1.1.5.0 = STRING: "Moxa"
iso.3.6.1.2.1.1.6.0 = STRING: "Sitting on the Dock of the Bay"
iso.3.6.1.2.1.1.7.0 = INTEGER: 72
iso.3.6.1.2.1.1.8.0 = Timeticks: (1) 0:00:00.01
iso.3.6.1.2.1.1.9.1.2.1 = OID: iso.3.6.1.6.3.11.3.1.1
iso.3.6.1.2.1.1.9.1.2.2 = OID: iso.3.6.1.6.3.15.2.1.1
iso.3.6.1.2.1.1.9.1.2.3 = OID: iso.3.6.1.6.3.10.3.1.1
iso.3.6.1.2.1.1.9.1.2.4 = OID: iso.3.6.1.6.3.1
iso.3.6.1.2.1.1.9.1.2.5 = OID: iso.3.6.1.6.3.16.2.2.1
iso.3.6.1.2.1.1.9.1.2.6 = OID: iso.3.6.1.2.1.49
iso.3.6.1.2.1.1.9.1.2.7 = OID: iso.3.6.1.2.1.4
iso.3.6.1.2.1.1.9.1.2.8 = OID: iso.3.6.1.2.1.50
iso.3.6.1.2.1.1.9.1.2.9 = OID: iso.3.6.1.6.3.13.3.1.3
iso.3.6.1.2.1.1.9.1.2.10 = OID: iso.3.6.1.2.1.92
iso.3.6.1.2.1.1.9.1.3.1 = STRING: "The MIB for Message Processing and
Dispatching."
iso.3.6.1.2.1.1.9.1.3.2 = STRING: "The management information definitions for the
SNMP User-based Security Model."
iso.3.6.1.2.1.1.9.1.3.3 = STRING: "The SNMP Management Architecture MIB."
iso.3.6.1.2.1.1.9.1.3.4 = STRING: "The MIB module for SNMPv2 entities"
iso.3.6.1.2.1.1.9.1.3.5 = STRING: "View-based Access Control Model for SNMP."
iso.3.6.1.2.1.1.9.1.3.6 = STRING: "The MIB module for managing TCP
implementations"
iso.3.6.1.2.1.1.9.1.3.7 = STRING: "The MIB module for managing IP and ICMP
implementations"
iso.3.6.1.2.1.1.9.1.3.8 = STRING: "The MIB module for managing UDP
implementations"
iso.3.6.1.2.1.1.9.1.3.9 = STRING: "The MIB modules for managing SNMP
Notification, plus filtering."
iso.3.6.1.2.1.1.9.1.3.10 = STRING: "The MIB module for logging SNMP
Notifications."
iso.3.6.1.2.1.1.9.1.4.1 = Timeticks: (0) 0:00:00.00
iso.3.6.1.2.1.1.9.1.4.2 = Timeticks: (0) 0:00:00.00
iso.3.6.1.2.1.1.9.1.4.3 = Timeticks: (0) 0:00:00.00
iso.3.6.1.2.1.1.9.1.4.4 = Timeticks: (0) 0:00:00.00
iso.3.6.1.2.1.1.9.1.4.5 = Timeticks: (0) 0:00:00.00
iso.3.6.1.2.1.1.9.1.4.6 = Timeticks: (0) 0:00:00.00
iso.3.6.1.2.1.1.9.1.4.7 = Timeticks: (0) 0:00:00.00
iso.3.6.1.2.1.1.9.1.4.8 = Timeticks: (0) 0:00:00.00
iso.3.6.1.2.1.1.9.1.4.9 = Timeticks: (0) 0:00:00.00
iso.3.6.1.2.1.1.9.1.4.10 = Timeticks: (1) 0:00:00.01
iso.3.6.1.2.1.25.1.1.0 = Timeticks: (1105369) 3:04:13.69
iso.3.6.1.2.1.25.1.2.0 = Hex-STRING: 07 E3 03 07 0E 02 10 00 2B 08 00
iso.3.6.1.2.1.25.1.3.0 = INTEGER: 393216
iso.3.6.1.2.1.25.1.4.0 = STRING: "BOOT_IMAGE=/boot/vmlinuz-4.9.0-6-amd64
root=UUID=babacb3b-a96c-449b-89e0-5505c43a1a40 ro quiet
```

```
"
iso.3.6.1.2.1.25.1.5.0 = Gauge32: 1
iso.3.6.1.2.1.25.1.6.0 = Gauge32: 102
iso.3.6.1.2.1.25.1.7.0 = INTEGER: 0
iso.3.6.1.2.1.25.1.7.0 = No more variables left in this MIB View (It is past the
end of the MIB tree)
```

OpenVPN

OpenVPN provides two types of tunnels for users to implement VPNS: **Routed IP Tunnels** and **Bridged Ethernet Tunnels**.

An Ethernet bridge is used to connect different Ethernet networks together. The Ethernets are bundled into one bigger, "logical" Ethernet. Each Ethernet corresponds to one physical interface (or port) that is connected to the bridge.

On each OpenVPN machine, you should carry out configurations in the **/etc/openvpn** directory, where script files and key files reside. Once established, all operations will be performed in that directory.

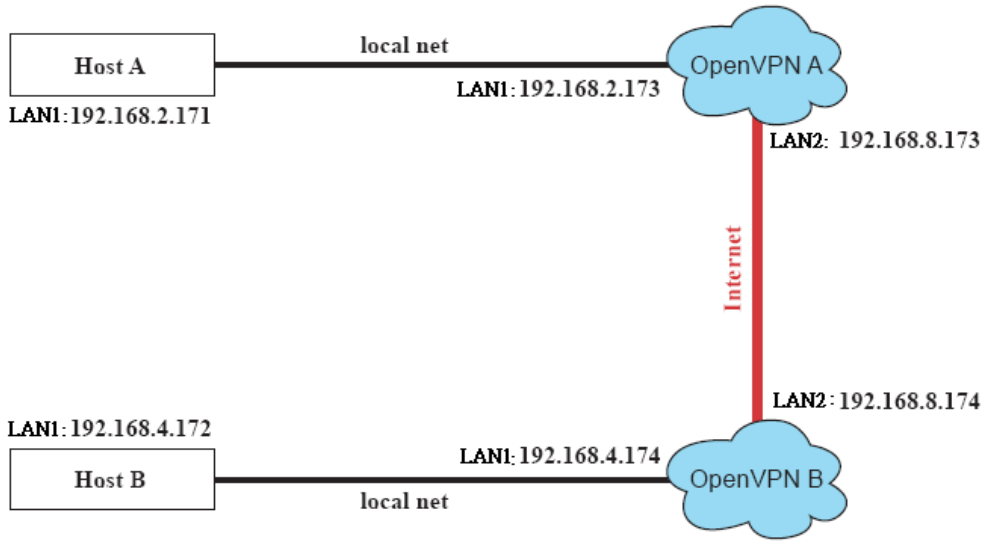
Install the OpenVPN

OpenVPN is a community VPN software. You can install it by this command.

```
moxa@Moxa:~$ sudo apt-get install openvpn
```

Ethernet Bridging for Private Networks on Different Subnets

1. Set up four machines, as shown in the following diagram.



   Host A represents the machine that belongs to OpenVPN A, and Host B represents the machine that belongs to OpenVPN B. The two remote subnets are configured for a different range of IP addresses. When this configuration is moved to a public network, the external interfaces of the OpenVPN machines should be configured for static IPs, or connected to another device (such as a firewall or DSL box) first.

2. Generate a preset shared key by typing the following command:
   # openvpn --genkey --secret secrouter.key

3. Copy the file that is generated to the OpenVPN machine:
   # scp /etc/openvpn/secrouter.key 192.168.8.174:/etc/openvpn

ATTENTION

A preshared key is located at **/etc/openvpn/secrouter.key**. You can use it for testing purposes. We suggest creating a new key for non-testing purposes.

4. On machine OpenVPN A, modify the remote address in configuration file /etc/openvpn/tap0-br.conf.

```
# point to the peer
remote 192.168.8.174
dev tap0
port 1194
secret /etc/openvpn/secrouter.key
cipher DES-EDE3-CBC
auth MD5
tun-mtu 1500
tun-mtu-extra 64
ping 40
up /etc/openvpn/tap0-br.sh
#comp-lzo
```

Next, modify the routing table in /etc/openvpn/tap0-br.sh script.

```
#---------------------------Start----------------------------
#!/bin/sh
# value after "-net" is the subnet behind the remote peer
route add -net 192.168.4.0 netmask 255.255.255.0 dev br0
#-----------------------------end----------------------------
```

And then configure the bridge interface in /etc/openvpn/bridge.

```
#!/bin/bash
# Create global variables
# Define Bridge Interface
br="br0"
# Define list of TAP interfaces to be bridged,
# for example tap="tap0 tap1 tap2".
tap="tap0"
# Define physical ethernet interface to be bridged
# with TAP interface(s) above.
eth="enp9s0"
eth_ip="192.168.8.173"
eth_netmask="255.255.255.0"
eth_broadcast="192.168.8.255"
#gw="192.168.8.174"
...
```

Start the bridge script file to configure the bridge interface:
 # /etc/openvpn/bridge restart

On machine OpenVPN B, modify the remote address in configuration file **#/etc/openvpn/tap0-br.conf.**

```
# point to the peer
remote 192.168.8.173
dev tap0
secret /etc/openvpn/secrouter.key
cipher DES-EDE3-CBC
auth MD5
tun-mtu 1500
tun-mtu-extra 64
ping 40
up /etc/openvpn/tap0-br.sh
#comp-lzo
```

5. Next modify the routing table in **/etc/openvpn/tap0-br.sh script** file.

```
#-----------------------------------Start-----------------------------
#!/bin/sh
# value after "-net" is the subnet behind the remote peer
route add -net 192.168.2.0 netmask 255.255.255.0 dev br0
#---------------------------------- end ----------------------------
```

6. Then configure the bridge interface in **/etc/openvpn/bridge**.

```
#!/bin/bash
# Create global variables
# Define Bridge Interface
br="br0"
# Define list of TAP interfaces to be bridged,
# for example tap="tap0 tap1 tap2".
tap="tap0"
# Define physical ethernet interface to be bridged
# with TAP interface(s) above.
eth="enp9s0"
eth_ip="192.168.8.174"
eth_netmask="255.255.255.0"
eth_broadcast="192.168.8.255"
#gw="192.168.8.173"
...
```

7. Start the bridge script file to configure the bridge interface.

   # /etc/openvpn/bridge restart

ATTENTION

Select cipher and authentication algorithms by specifying cipher and auth. To see which algorithms are available, type:

openvpn --show-ciphers
openvpn --show-auths

8. Start both OpenVPN peers on machine OpenVPN A and OpenVPN B.

   # openvpn --config /etc/openvpn/tap0-br.conf&

   If you see the line **Peer Connection Initiated with 192.168.8.173:5000** on each machine, the connection between OpenVPN machines has been established successfully on UDP port 5000.

ATTENTION

You can create link symbols to start the OpenVPN service at boot time:

 # ln -sf /etc/init.d/openvpn /etc/rc2.d/S16openvpn

To stop the service, you should create these links:

 # ln -sf /etc/init.d/openvpn /etc/rc0.d/K80openvpn
 # ln -sf /etc/init.d/openvpn /etc/rc6.d/K80openvpn

9. On each OpenVPN machine, check the routing table by typing the command **# route**

```
Destination     Gateway Genmsk          Flags   Metric  Ref Use Iface
192.168.5.0     0.0.0.0 255.255.255.0   U       0       0   0   enp10s0
192.168.4.0     0.0.0.0 255.255.255.0   U       0       0   0   br0
192.168.3.0     0.0.0.0 255.255.255.0   U       0       0   0   enp0s31f6
```

```
192.168.30.0    0.0.0.0 255.255.255.0   U       0       0    0   enp11s0
192.168.8.0     0.0.0.0 255.255.255.0   U       0       0    0   br0
```

Interface **enp9s0** and device **tap0** both connect to the bridging interface, and the virtual device **tun** sits on top of **tap0**. This ensures that all traffic coming to this bridge from internal networks connected to interface enp9s0 write to the TAP/TUN device that the OpenVPN program monitors. Once the OpenVPN program detects traffic on the virtual device, it sends the traffic to its peer.

10. To create an indirect connection to Host B from Host A, you need to add the following routing item:
    # route add –net 192.168.4.0 netmask 255.255.255.0 dev enp0s31f6

    To create an indirect connection to Host A from Host B, you need to add the following routing item:
    # route add –net 192.168.2.0 netmask 255.255.255.0 dev enp0s31f6

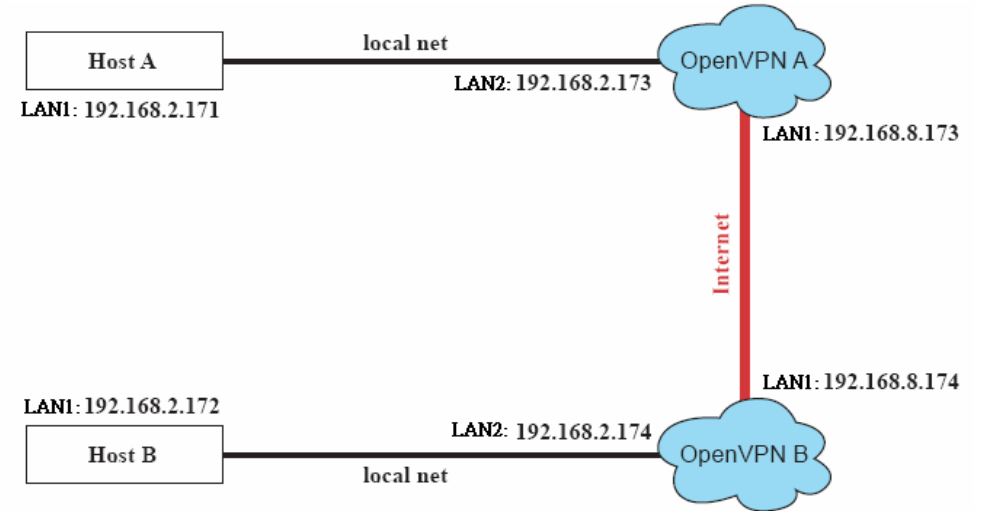
    Now ping Host B from Host A by typing:
    # ping 192.168.4.174

    A successful ping indicates that you have created a VPN system that only allows authorized users from one internal network to access users at the remote site. For this system, all data is transmitted by UDP packets on port 5000 between OpenVPN peers.

11. To shut down OpenVPN programs, type the command:
    # killall -TERM openvpn

Ethernet Bridging for Private Networks on the Same Subnet

1. Set up four machines, as shown in the following diagram.

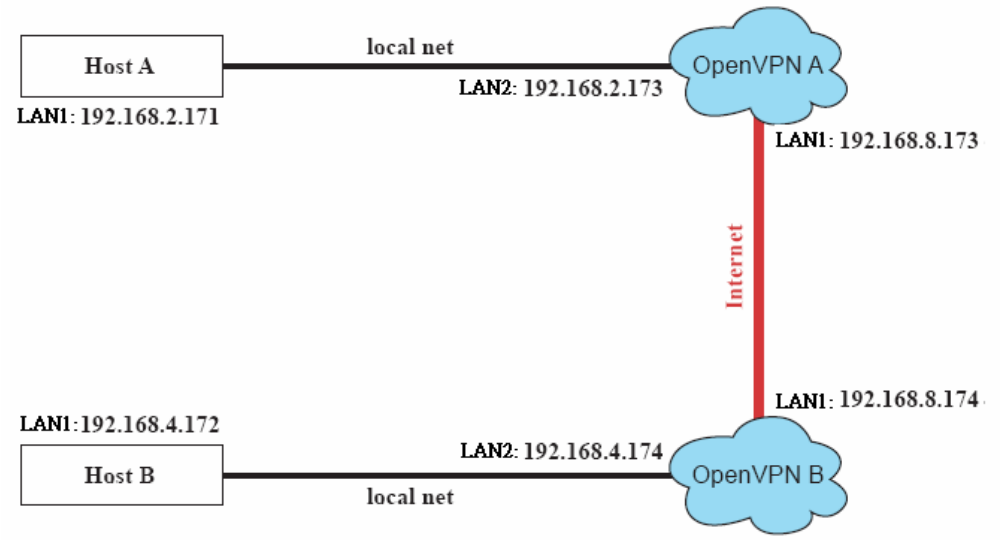


2. The configuration procedure is almost the same as for the previous example. The only difference is that you will need to comment out the parameter **up** in **/etc/openvpn/tap0-br.conf** of OpenVPN A and **/etc/openvpn/tap0-br.conf** of OpenVPN B.

```
# point to the peer
remote 192.168.8.174
dev tap0
secret /etc/openvpn/secrouter.key
cipher DES-EDE3-CBC
auth MD5
tun-mtu 1500
tun-mtu-extra 64
ping 40
#up /etc/openvpn/tap0-br.sh
#comp-lzo
```

Routed IP

1. Set up four machines, as shown in the following diagram.



2. On machine OpenVPN A, modify the remote address in configuration file **/etc/openvpn/tun.conf**.

```
# point to the peer
remote 192.168.8.174
dev tun
secret /etc/openvpn/secrouter.key
cipher DES-EDE3-CBC
auth MD5
tun-mtu 1500
tun-mtu-extra 64
ping 40
ifconfig 192.168.2.173 192.168.4.174
up /etc/openvpn/tun.sh
-----
```

3. Next, modify the routing table in script file **/etc/openvpn/tun.sh**.

```
#----------------------------Start-----------------------------
#!/bin/sh
# value after "-net" is the subnet behind the remote peer
route add -net 192.168.2.0 netmask 255.255.255.0 gw $5
#-----------------------------end-----------------------------
```

4. On machine OpenVPN B, modify the remote address in configuration file **/etc/openvpn/tun.conf**.

```
# point to the peer
remote 192.168.8.173
dev tun
secret /etc/openvpn/secrouter.key
cipher DES-EDE3-CBC
auth MD5
tun-mtu 1500
tun-mtu-extra 64
ping 40
ifconfig 192.168.4.174 192.168.2.173
up /etc/openvpn/tun.sh
```

And then modify the routing table in script file **/etc/openvpn/tun.sh**.

```
#--------------------------Start-----------------------------
#!/bin/sh
# value after "-net" is the subnet behind the remote peer
route add -net 192.168.2.0 netmask 255.255.255.0 gw $5
#---------------------------end-----------------------------
```

The first argument of parameter **ifconfig** is the local internal interface and the second argument is the internal interface at the remote peer.

$5 is the argument that the OpenVPN program passes to the script file. Its value is the second argument of **ifconfig** in the configuration file.

5. Check the routing table after you run the OpenVPN programs, by typing the command **# route**.

```
Destination     Gateway         Genmsk          Flags   Metric  Ref Use Iface
192.168.4.174   *               255.255.255.255 UH      0       0   0   tun0
192.168.4.0     192.168.4.174   255.255.255.0   UG      0       0   0   tun0
192.168.2.0     *               255.255.255.0   U       0       0   0   enp9s0
192.168.8.0     *               255.255.255.0   U       0       0   0
    enp0s31f6
```

4

System Restore and Backup

The NGS computer is installed with the Embedded Linux operating system, which is located in the mSATA shipped with the NGS computer. Although it rarely happens, you may find on occasion that operating system files and/or the disk file system have been damaged. In this chapter we describe how to restore the Linux operating system.

The following topics are covered in this chapter:

- **Restore Environment**
- **Restore the System from the USB Drive**
- **Backup the System to the USB Drive**

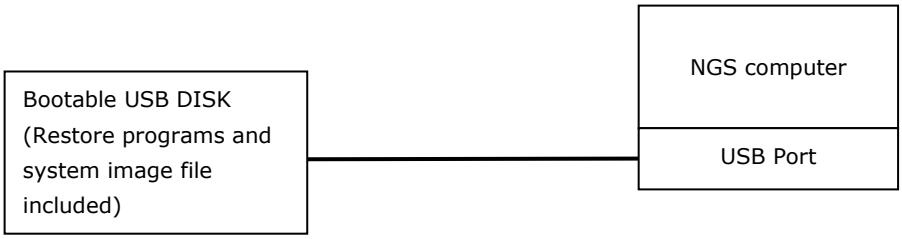
Restore Environment

The restore environment includes the NGS embedded computer and a bootable USB disk with the restore programs and system image file.

Hardware

The hardware used includes a PC, a NGS computer and a USB disk with the restore programs.

NOTE The USB disk should be at least 2GB.



Restore the System from the USB Drive

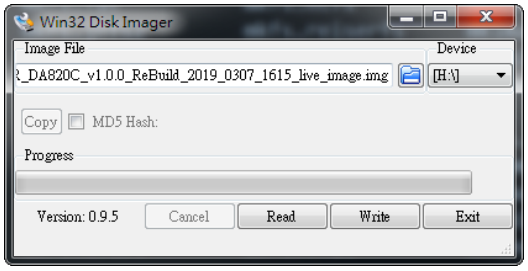
Step 1: Prepare your USB drive

For Windows user:

Execute **Win32DiskImager installer** from the **utility_tools/** folder on the Software CD. Or you can download it from https://sourceforge.net/projects/win32diskimager/

After install processes, execute **Win32DiskImager,** and select the Moxa Live USB image file in the directory of
Restore\moxa_live_image\FWR_<product>_<version>_ReBuild_<date>_live_image.img

The Moxa Live USB image file **contains corresponding firmware image**.



For Debian Linux user:

Copy the ISO file in the directory of

Restore\moxa_live_image\FWR_<product>_<version>_ReBuild_<date>_live_image.img

For example on DA-820C: (/dev/sde is USB storage device node)

```
root@Moxa:/home/moxa# dd if=FWR_DA820C_<version>_ReBuild_<date>_live_image.img
of=/dev/sde conv=noerror,sync status=progress bs=4096
262144+0 records in
262144+0 records out
1073741824 bytes (1.1 GB, 1.0 GiB) copied, 403.449 s, 2.7 MB/s
root@Moxa:/home/moxa# parted /dev/sde print Fix
Warning: Not all of the space available to /dev/sde appears to be used, you can
fix the GPT to use all of the space (an extra 13126656 blocks) or continue with
the
current setting?
```

```
Model: SanDisk Cruzer Blade (scsi)
Disk /dev/sde: 7795MB
Sector size (logical/physical): 512B/512B
Partition Table: gpt
Disk Flags:

Number  Start   End     Size    File system  Name   Flags
 1      1049kB  10.5MB  9437kB  fat16        EFI    boot, esp
 2      10.5MB  268MB   258MB   ext4         ROOT
 3      268MB   1073MB  804MB   ext4         IMAGE
root@Moxa:/home/moxa# parted -s /dev/sde resizepart 3 100%
root@Moxa:/home/moxa# sync
root@Moxa:/home/moxa# e2fsck -fy /dev/sde3
e2fsck 1.43.4 (31-Jan-2017)
Pass 1: Checking inodes, blocks, and sizes
Pass 2: Checking directory structure
Pass 3: Checking directory connectivity
Pass 4: Checking reference counts
Pass 5: Checking group summary information
/dev/sde3: 23/49152 files (0.0% non-contiguous), 136129/196352 blocks
root@Moxa:/home/moxa# resize2fs /dev/sde3
resize2fs 1.43.4 (31-Jan-2017)
Resizing the filesystem on /dev/sde3 to 1837435 (4k) blocks.
The filesystem on /dev/sde3 is now 1837435 (4k) blocks long.
```

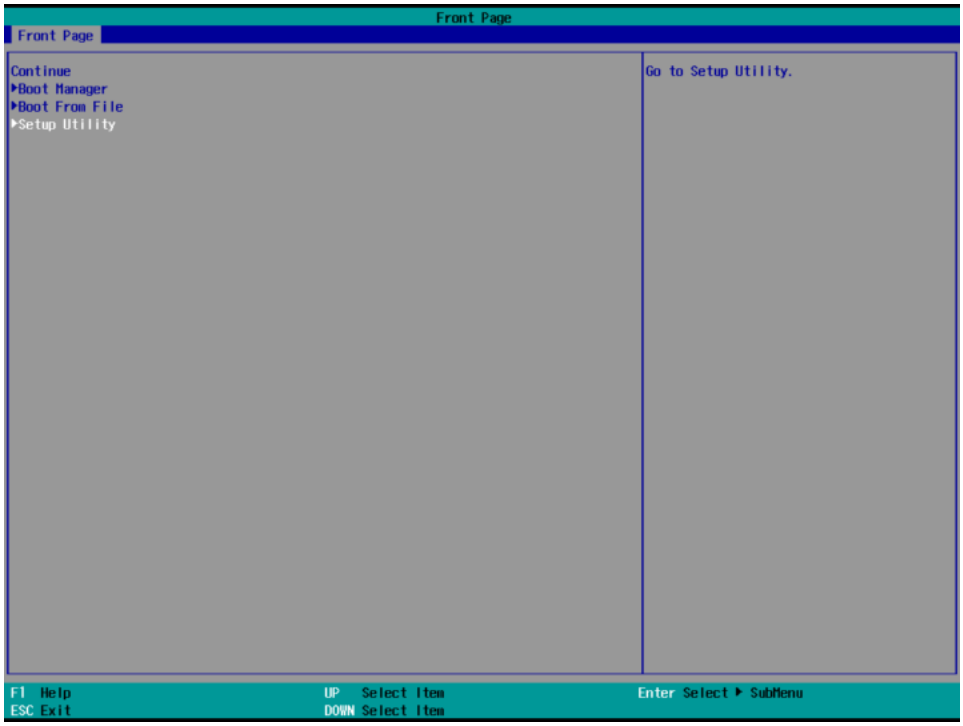
The standalone Debian firmware image file is in the directory:
Restore\firmware\FWR_<product>_<version>_ReBuild_<date>.img

Compressed firmware image is available at:
Restore\firmware\FWR_<product>_<version>_ReBuild_<date>.img.gz

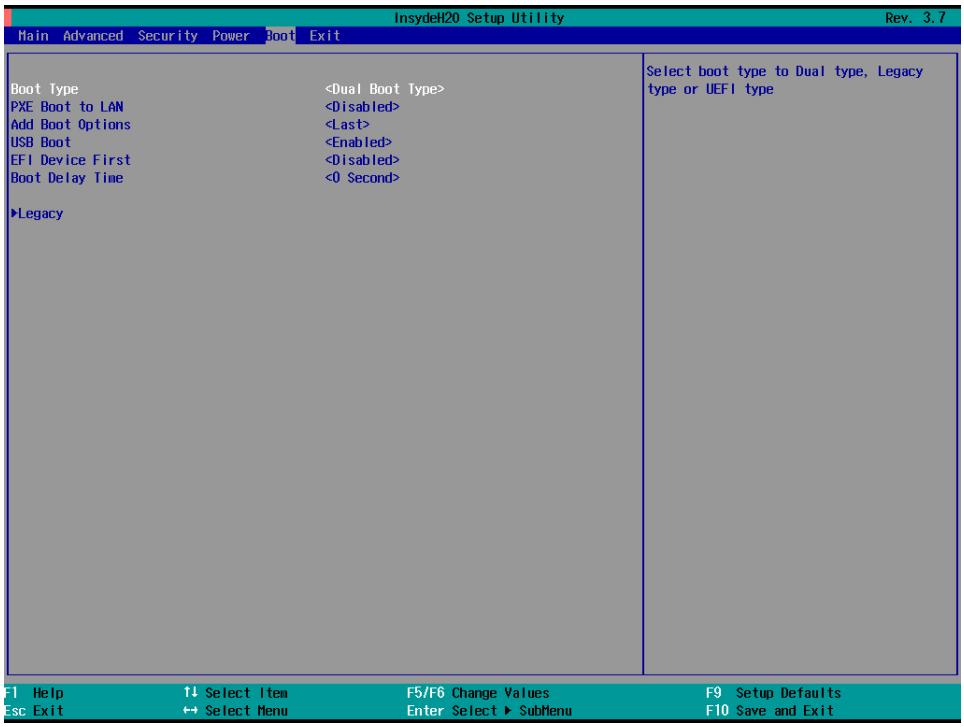
Step 2: Change the BIOS Settings

You will need to change the BIOS settings to boot from the USB disk.

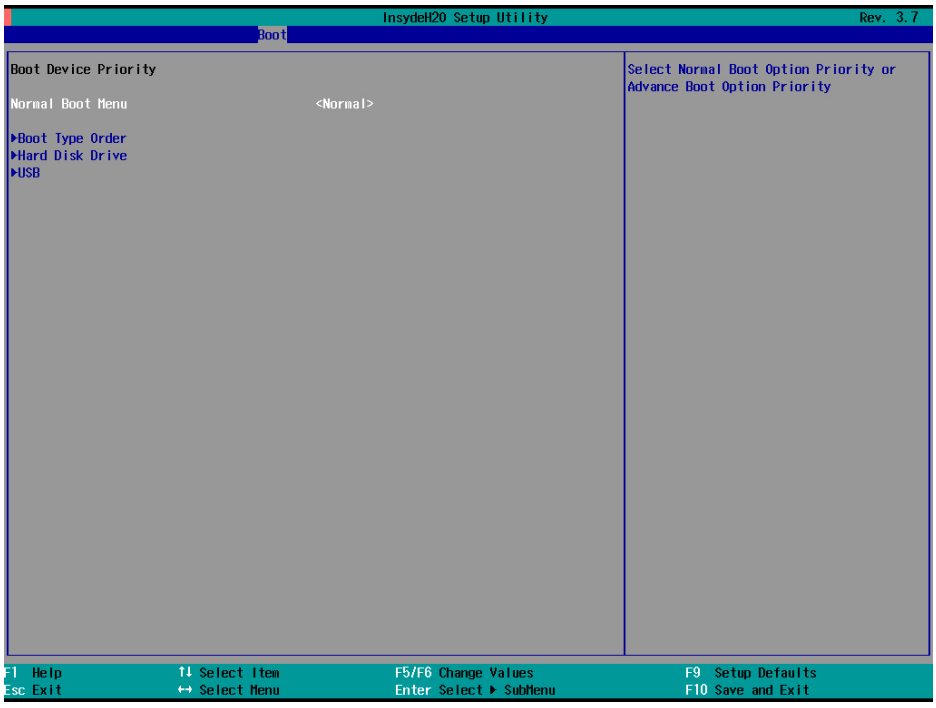
1. Turn on the computer and press **F2.** Select **Setup Utility** in the following screen.



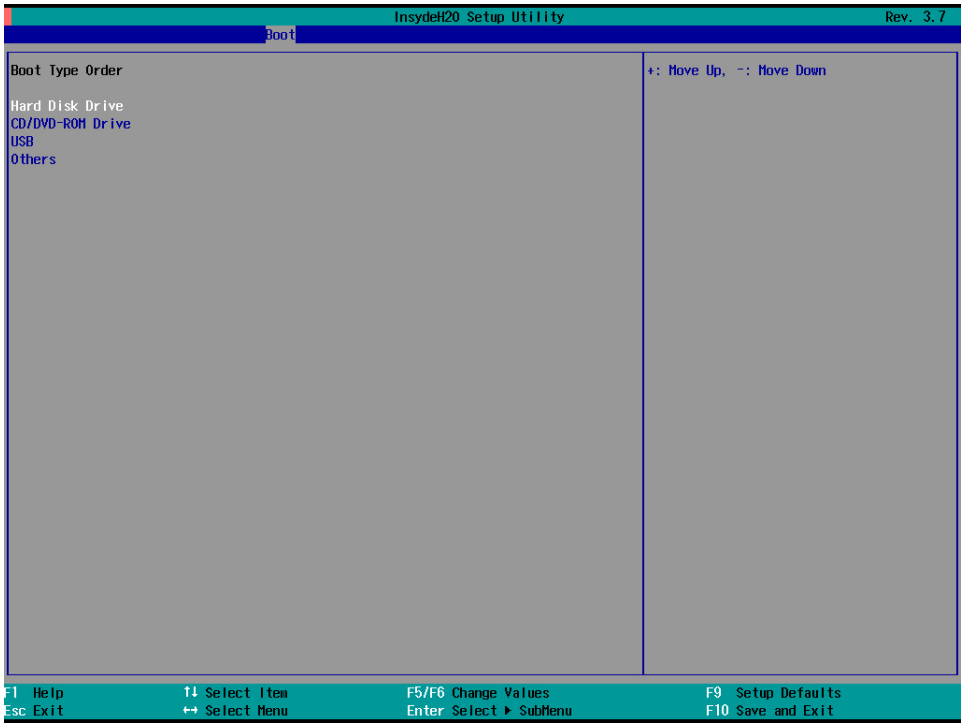
2. Select **Boot** and then select UEFI **Boot Type**. Press **Enter** to continue.



3. Select **Boot order**.



4. Select USB disk and then press "+" to move it to the first boot device position.
   Warning: An incorrect boot priority will lead to restore or boot failure.



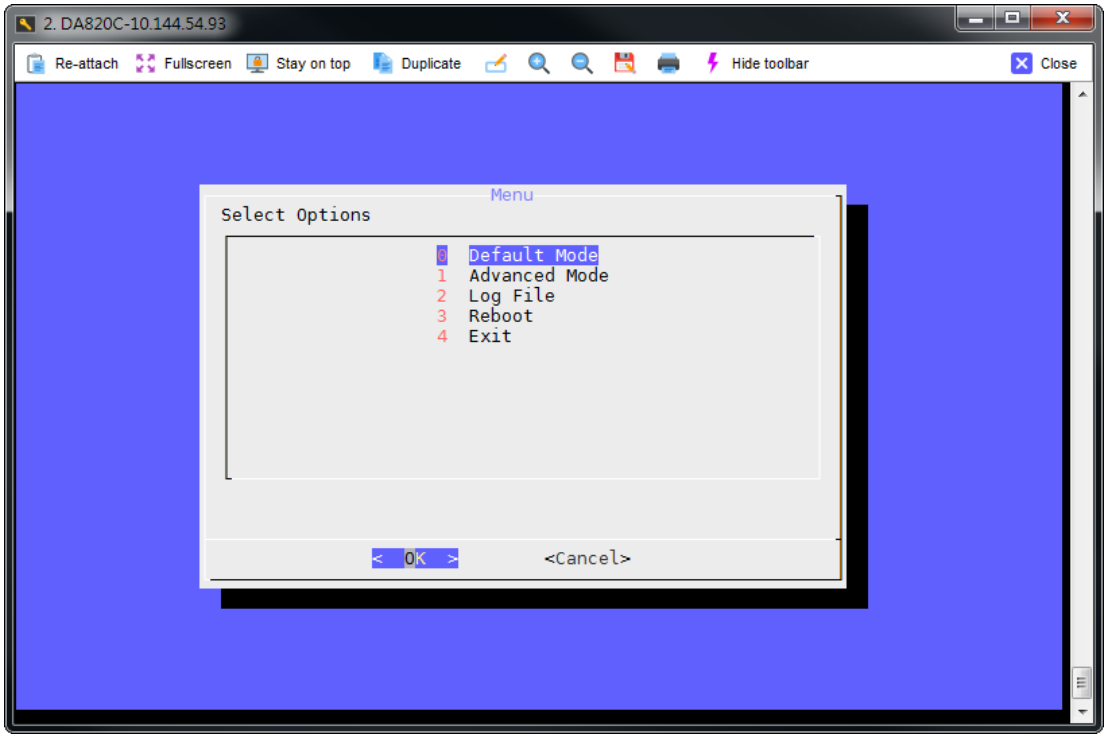
5. Press **F10** and then press **Enter** to save and exit BIOS setup.
6. Insert the USB disk and then reboot the computer.
7. Press F2 to enter the BIOS setting.
8. Select the **Boot Manager**.
9. Select EFI **USB device.**
   The system will boot from the restore utility.

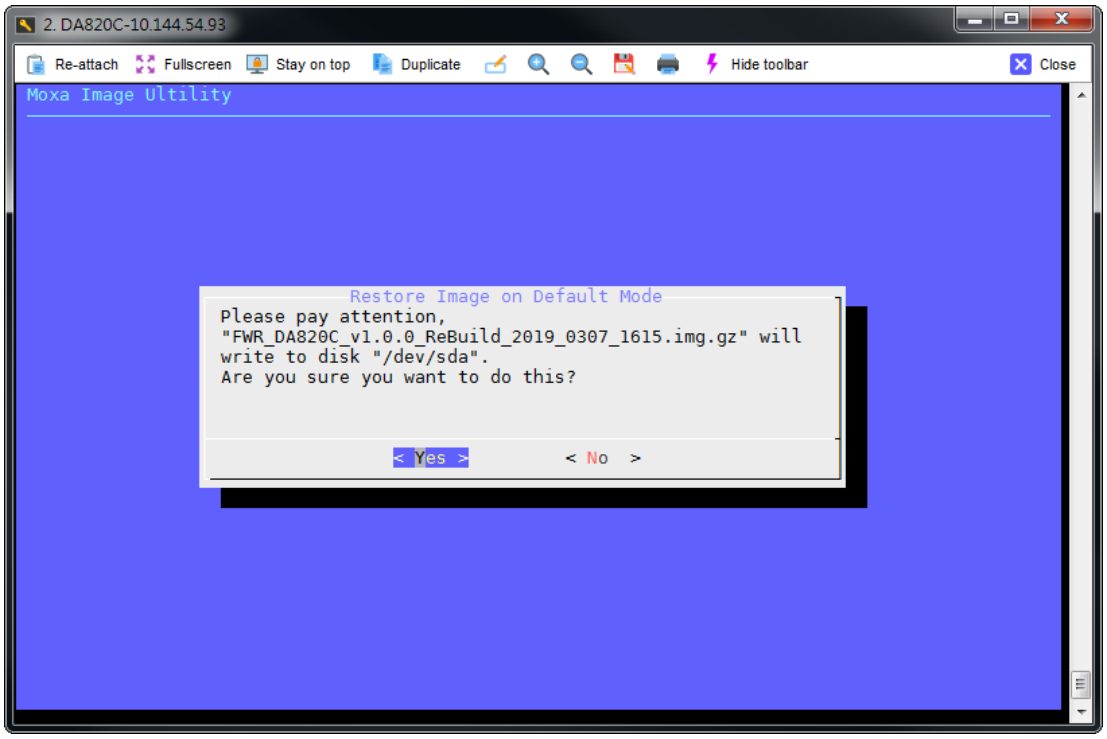
Step 3: Restore the system from the USB drive

Connect the USB disk to any of the NGS computer's USB ports and then reboot the computer. The system will boot from the USB disk and the Pre-installation Environment and the restore utility will appear.

[Default Mode]

Select "Default Mode" will write default image to default mSATA disk. **If you have multiple images or storage disks, to select "Advanced mode" is strongly suggested.**

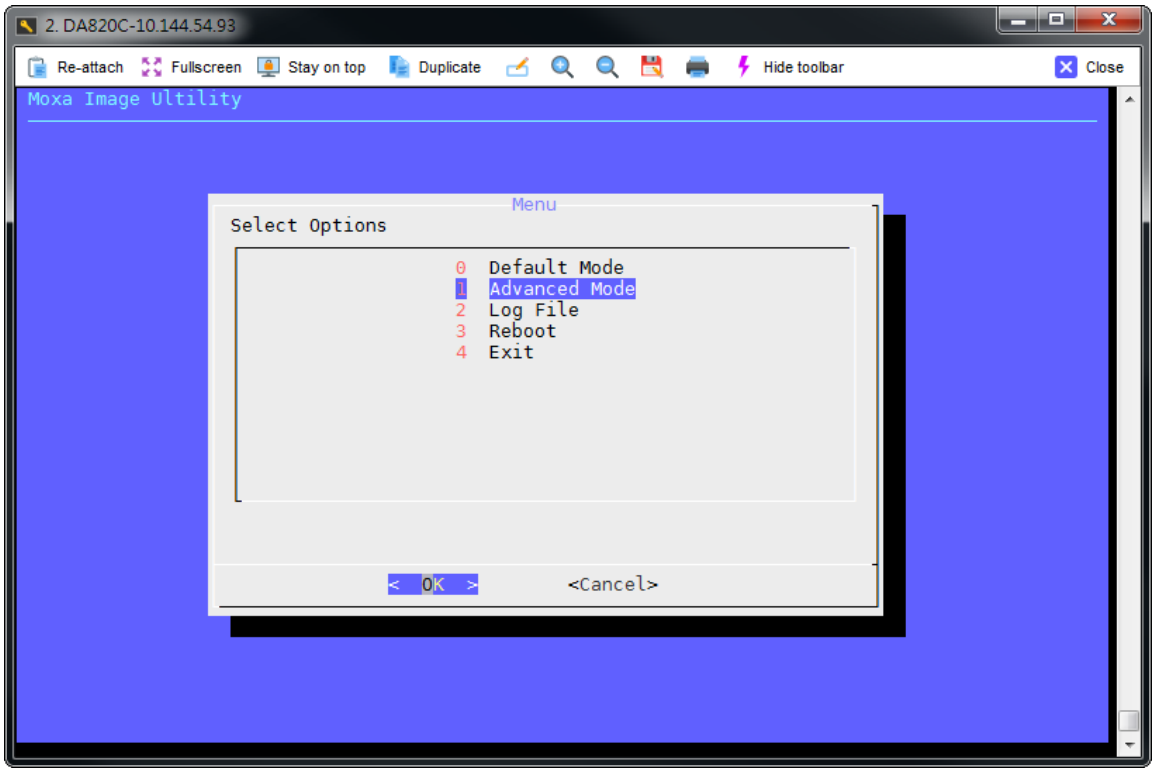




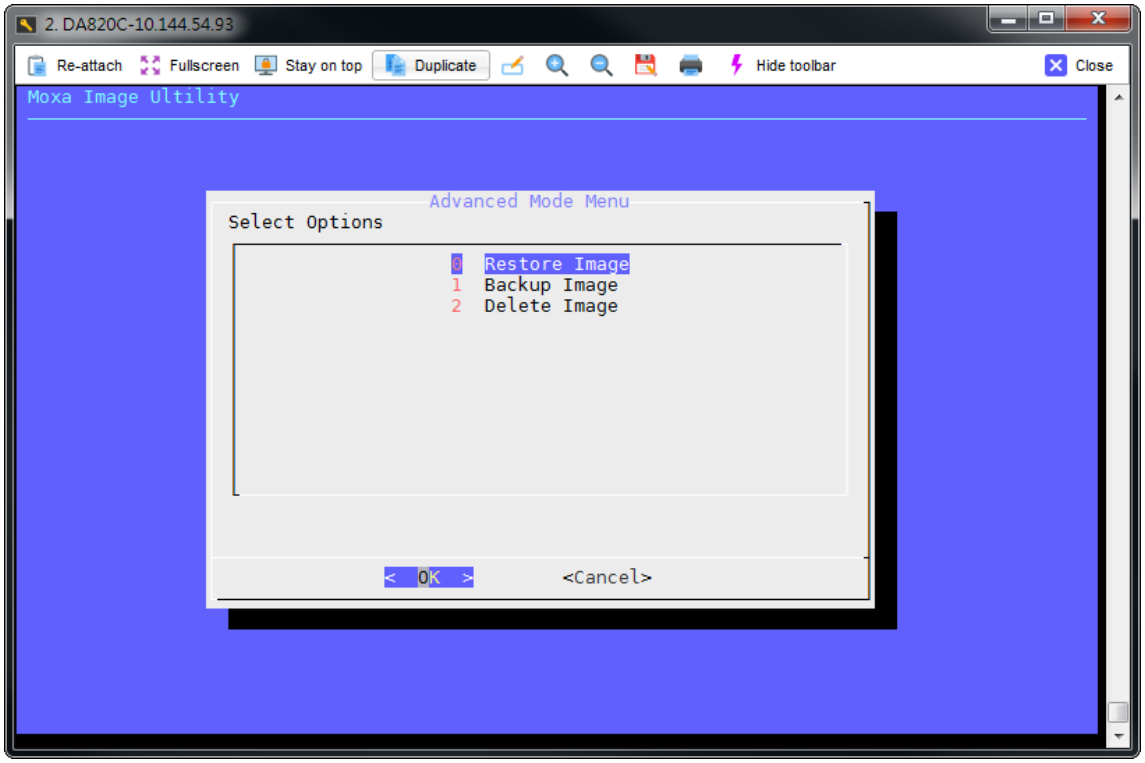
Press OK and wait for restore image process. If the process was finished, you can select to reboot, and remove the USB drive after the computer has been powered off, and jump to Step4.

[Advanced Mode]

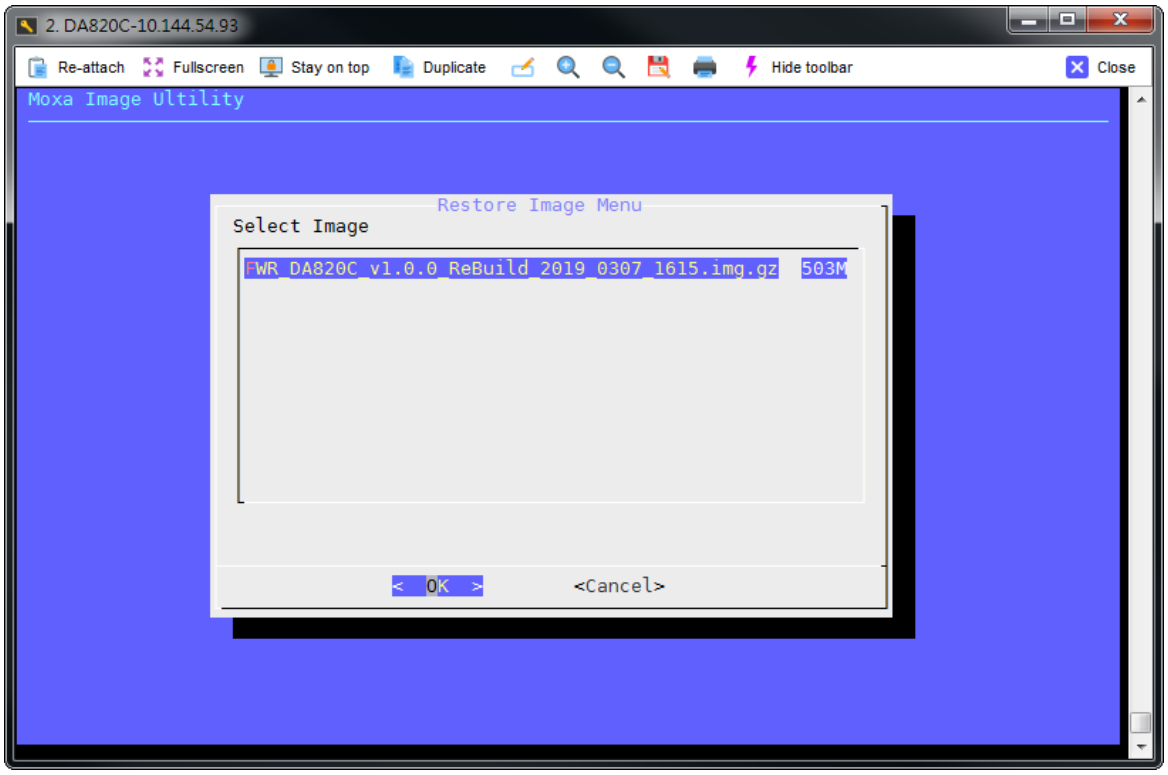
To select "Advanced Mode":



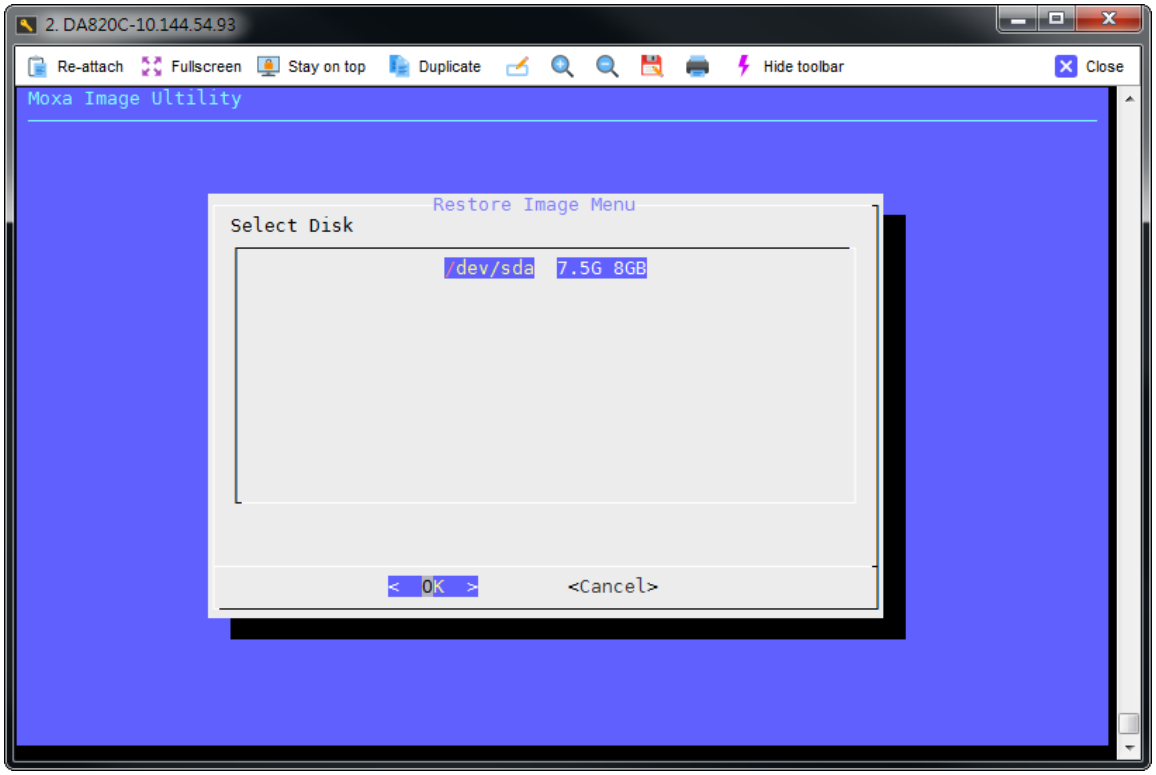
To select "Restore Image":



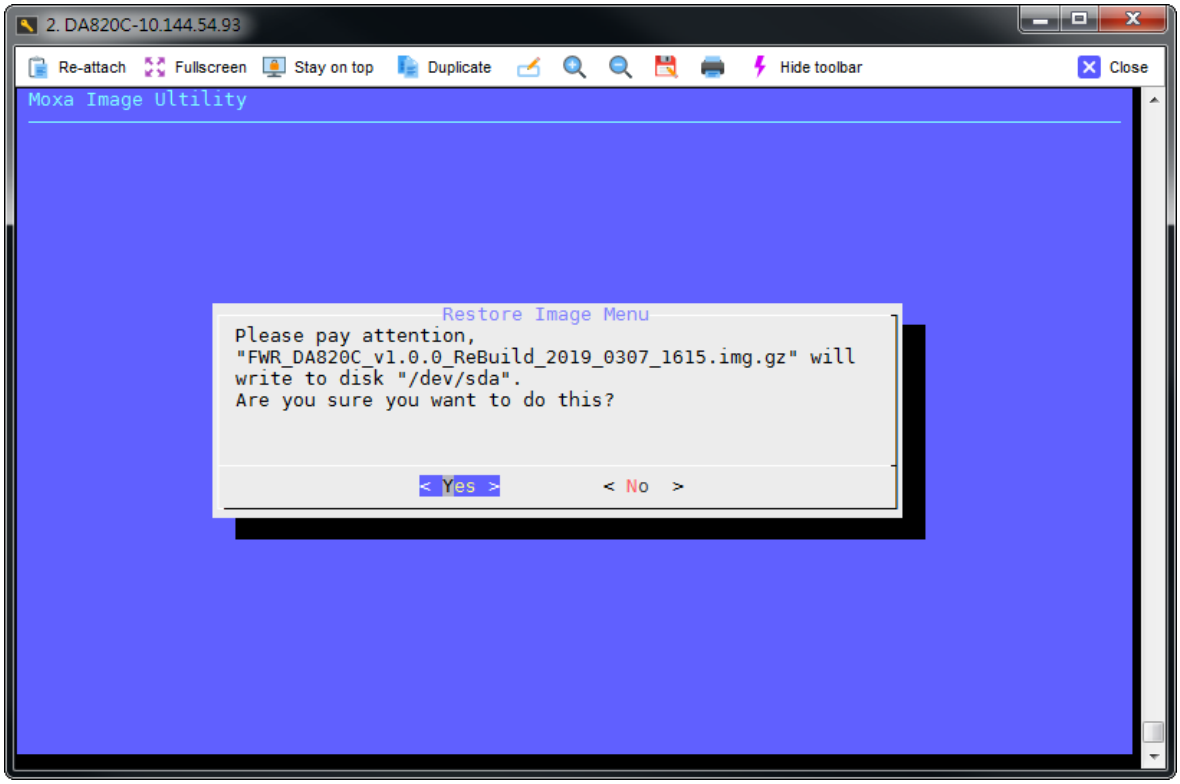
To select the target image:



To select the target storage disk:



And make sure again, this step will erase all partitions in the disk.

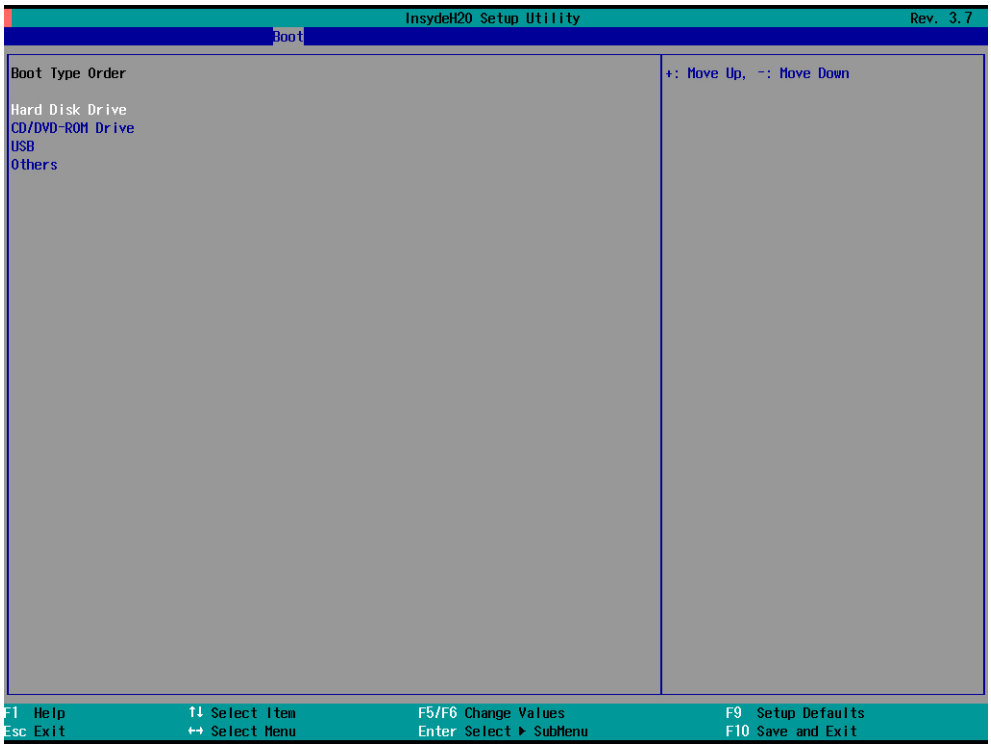


Press OK and wait for restore image process. If the process was finished, you can select to reboot, and remove the USB drive after the computer has been powered off, and jump to Step4.

Step 4: Change the BIOS Settings to Boot from the Original Disk

Now you will need to change the boot priority so that it can boot from the original disk. As the system reboots, press **F2** to enter the BIOS setup menu.

1. Select **Hard Disk Drive** and then press + to move to the first boot device position, and then press **Enter.** Make sure the hard disk has first boot priority.



2. Press **F10** and then press **Enter** to save and exit BIOS settings.

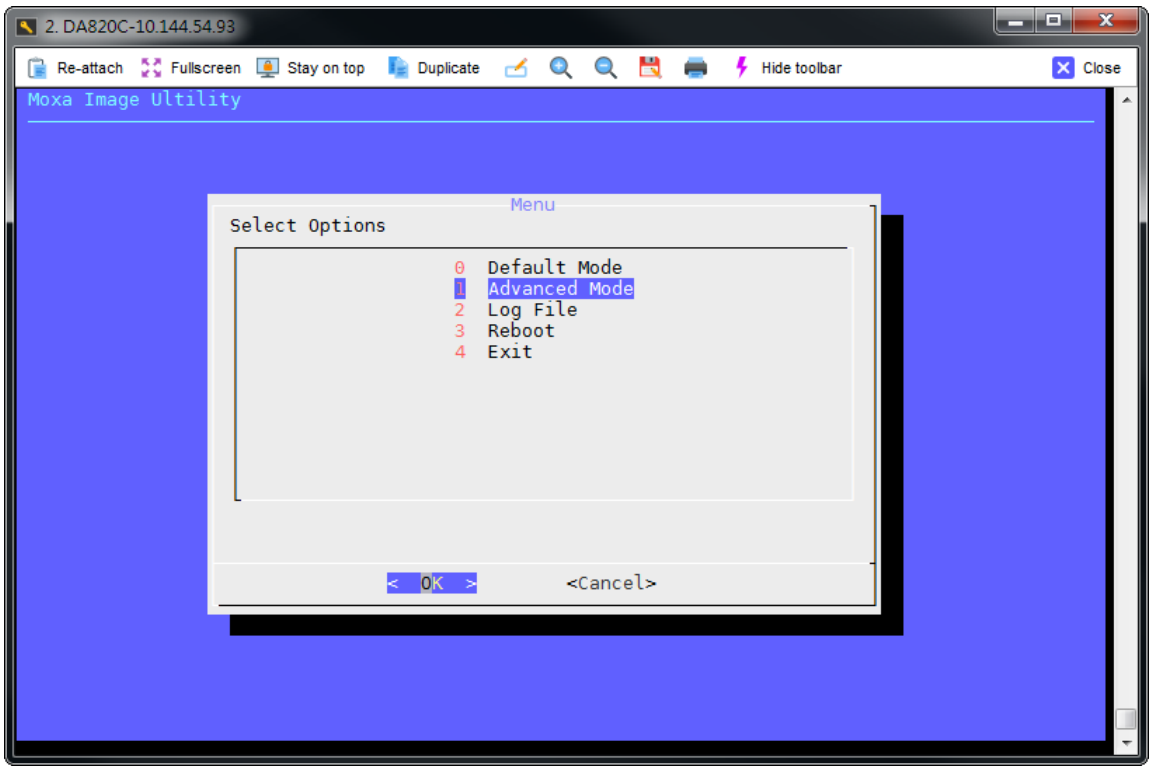
Step 5: Reboot the Computer

You need to wait about 10 to 15 minutes for the system to restart, since the system configuration files will be initiated while booting up for the first time. **Do not turn off the computer or shut down the computer** while the system is restarting; otherwise, the IIS service will be terminated. When the operating system has successfully launched, you will need to restart your computer so that the new settings can be activated.

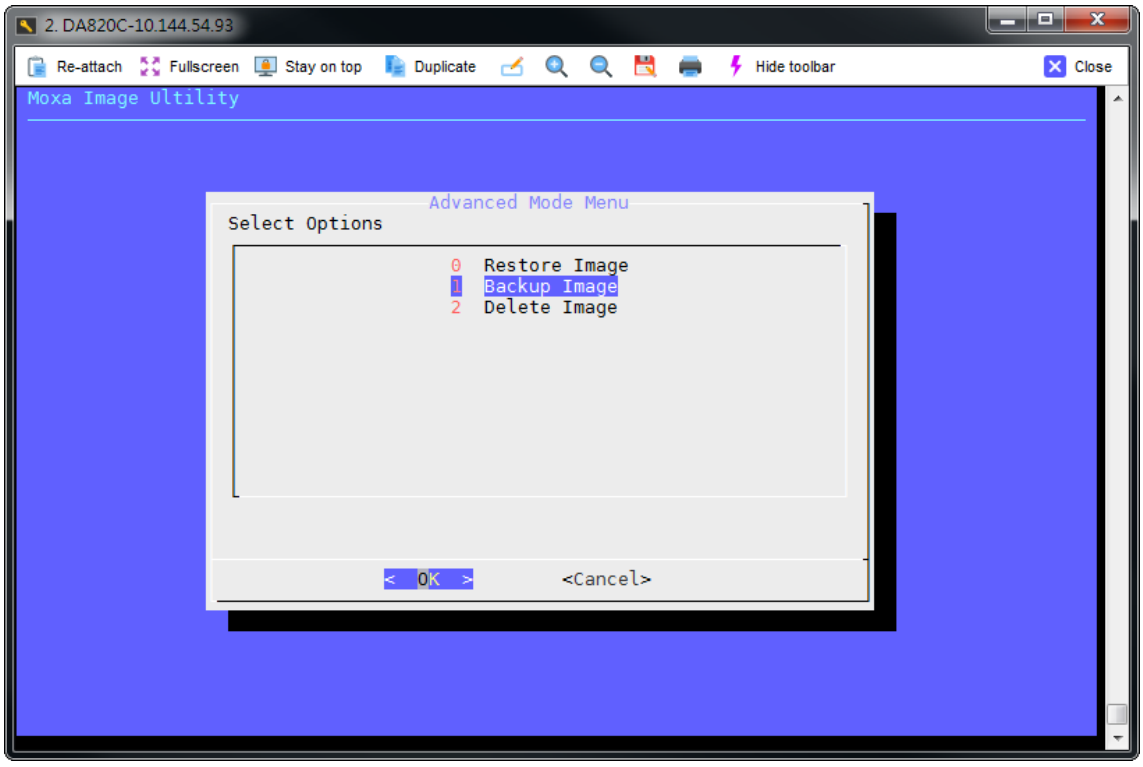
Backup the System to the USB Drive

You may also backup the current system to the USB drive for system restore in case the system crashes. Change the BIOS settings to make the USB drive the first boot priority. When the system has been launched, take the following steps.

To select "Advanced Mode":



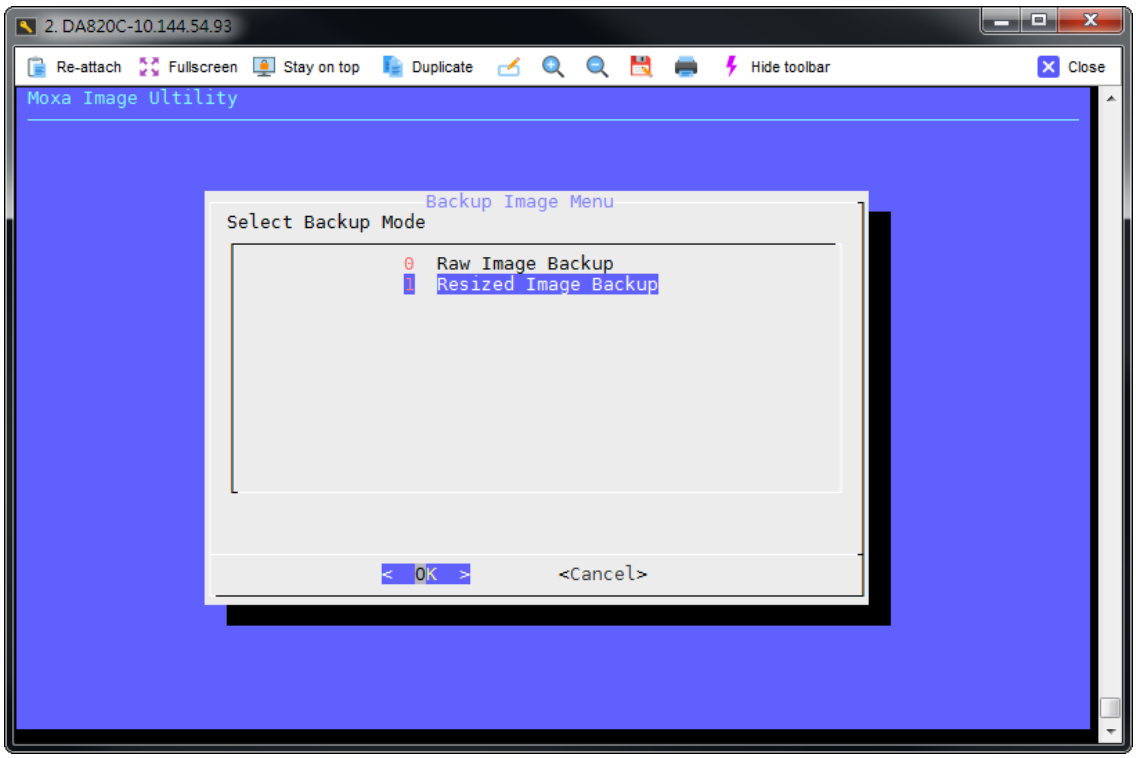
To select “Backup Image”:



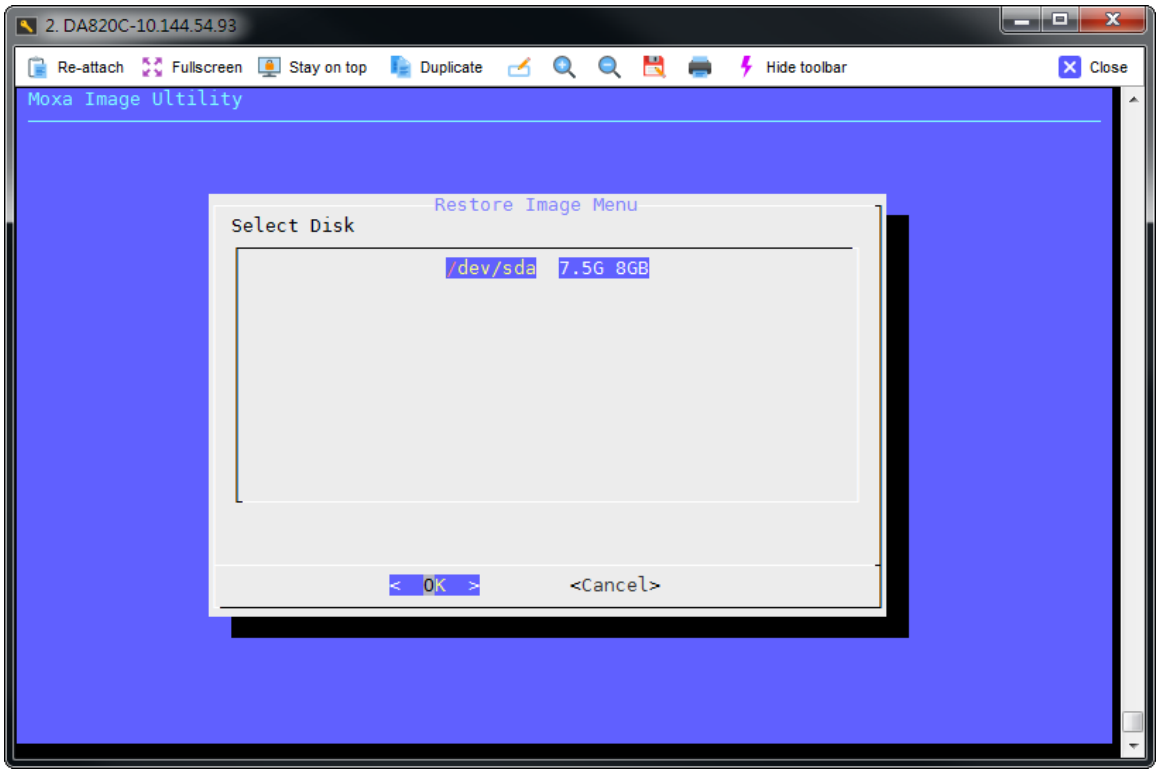
If you want to backup raw image, please select “Raw Image Backup”.

If you want to backup image and resize disk size, please select “Resized Image Backup”.

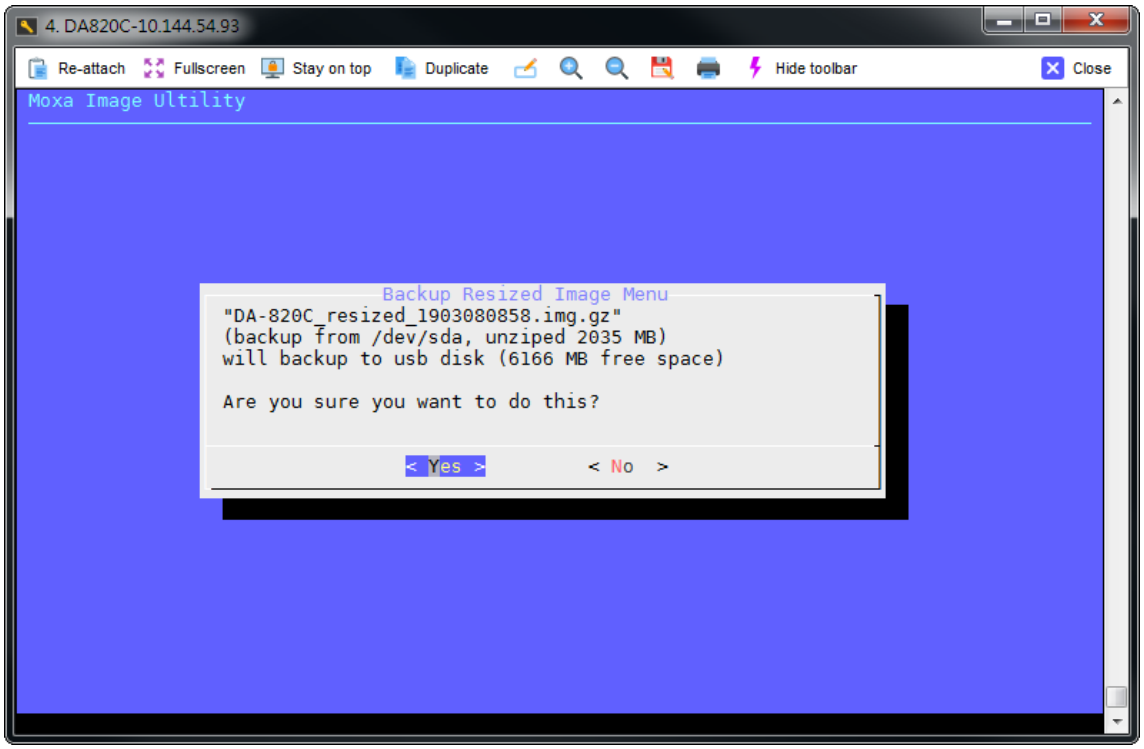
For example, to select “Resized Image Backup”.



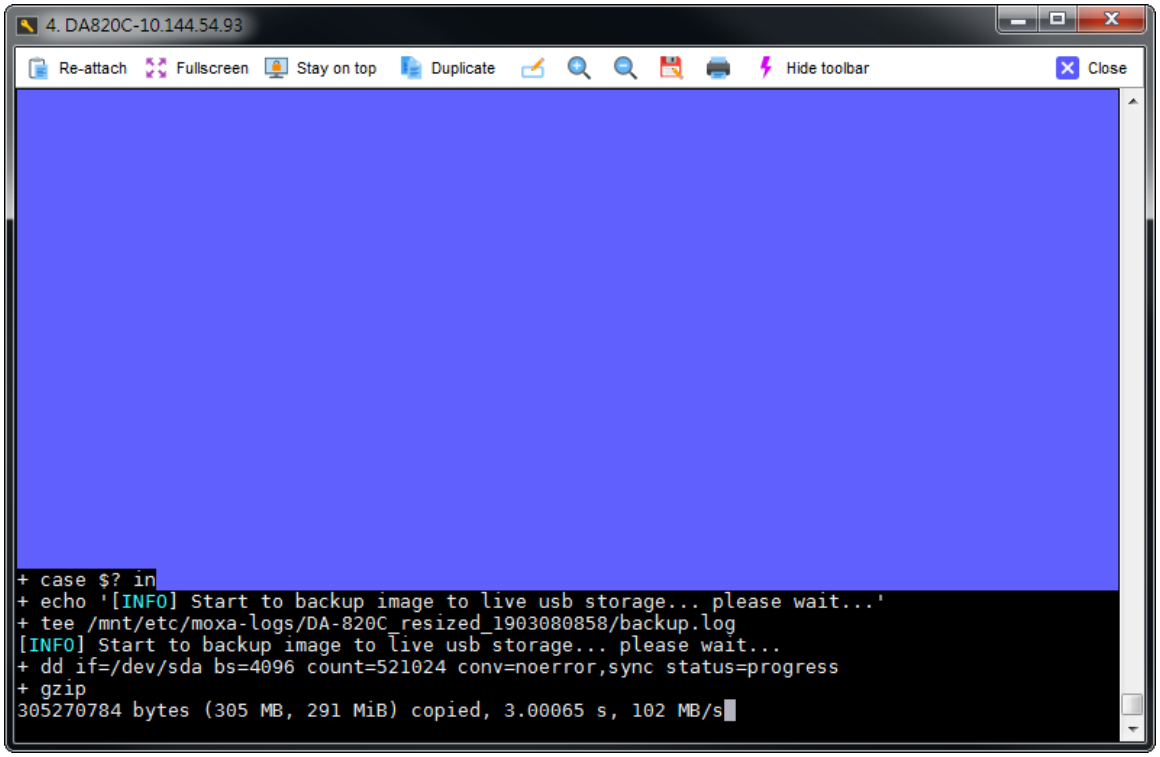
To select the target storage disk to backup:



The check box will show the backup information, including image size and the rest free space of USB live disk.

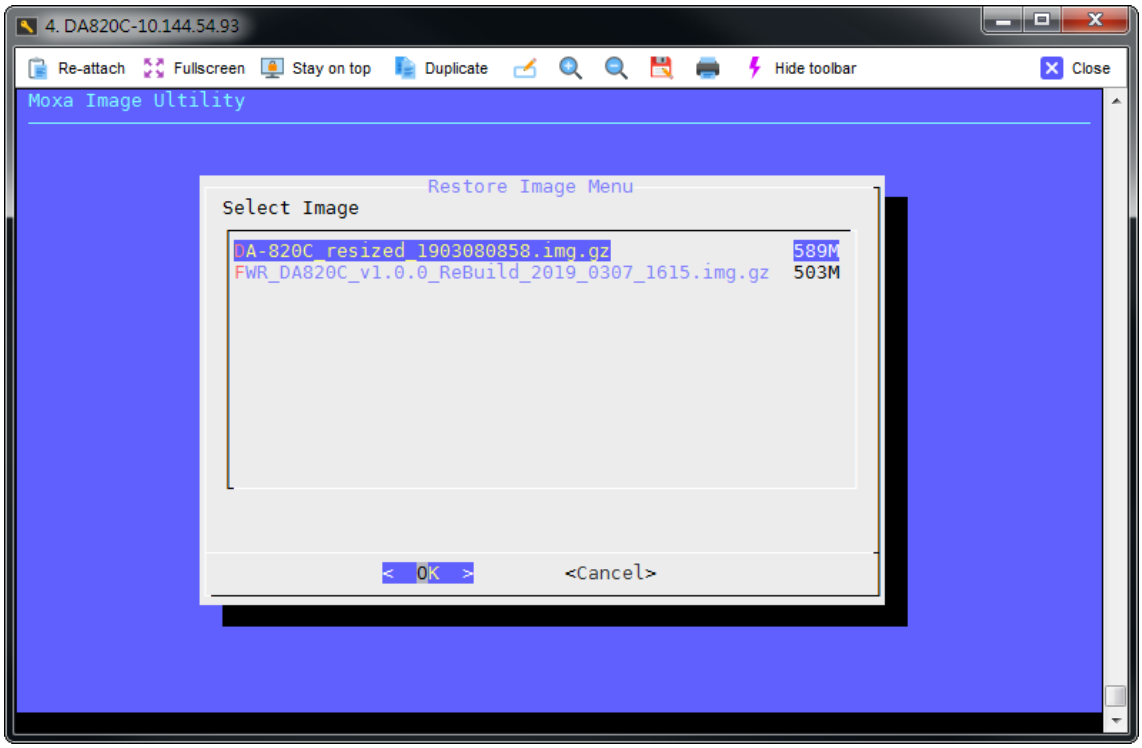


Select “Yes” button and start to backup disk.

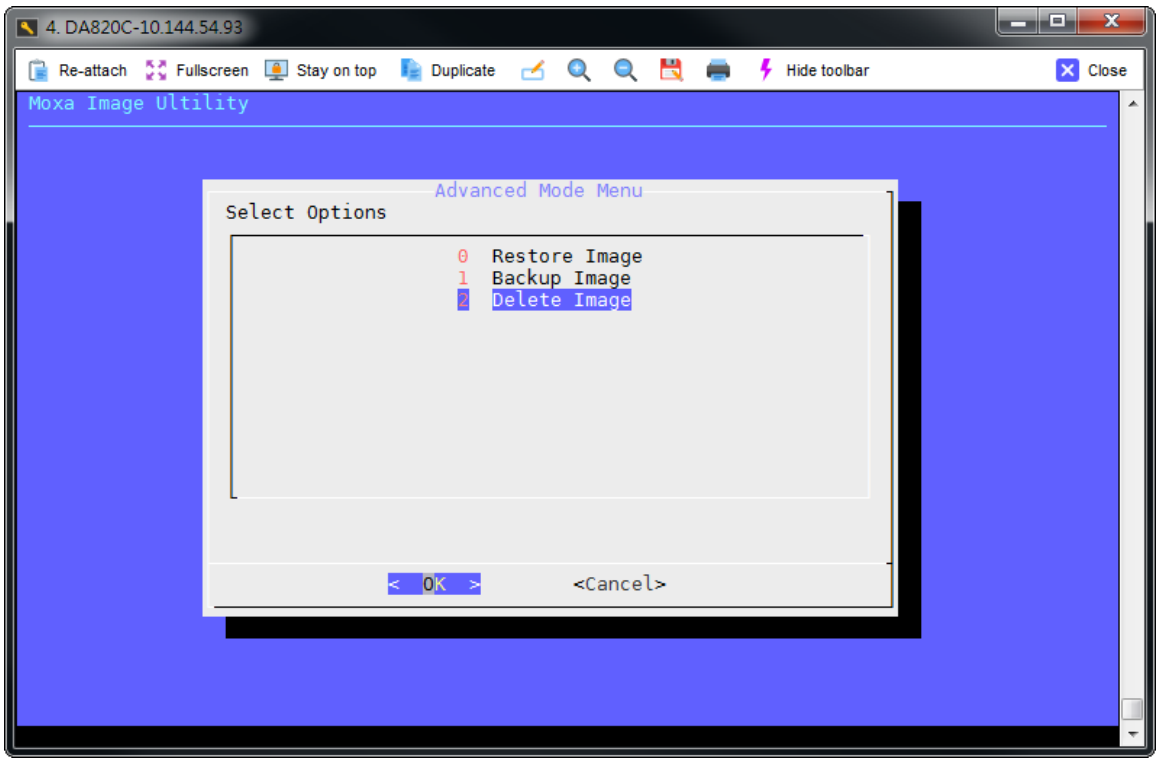


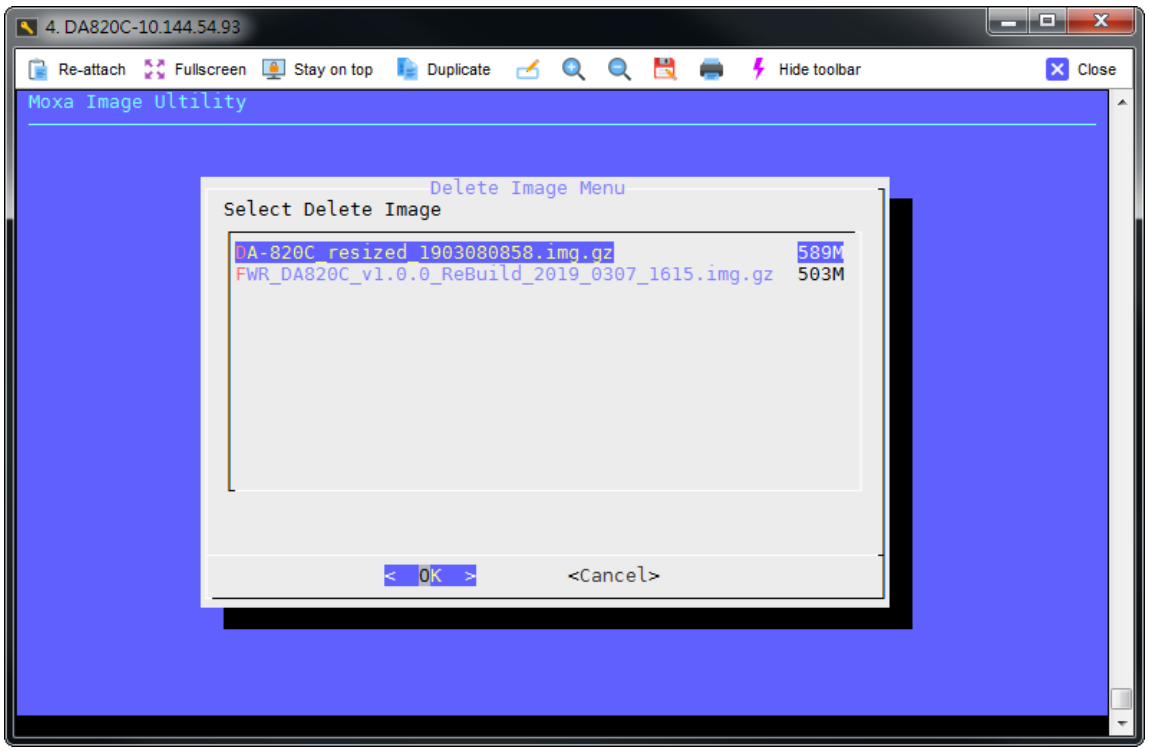
After backup process, the backup image was stored at live USB storage.

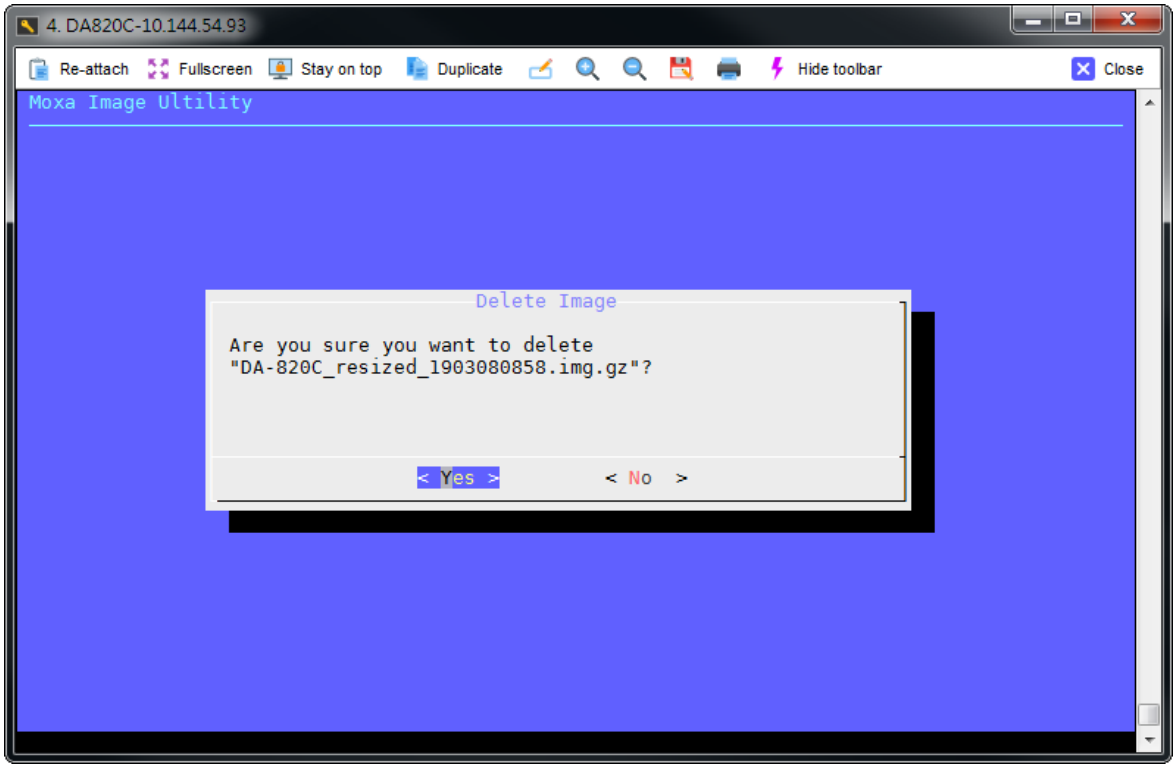
To select restore page, you can find out the backup image is located on USB storage.

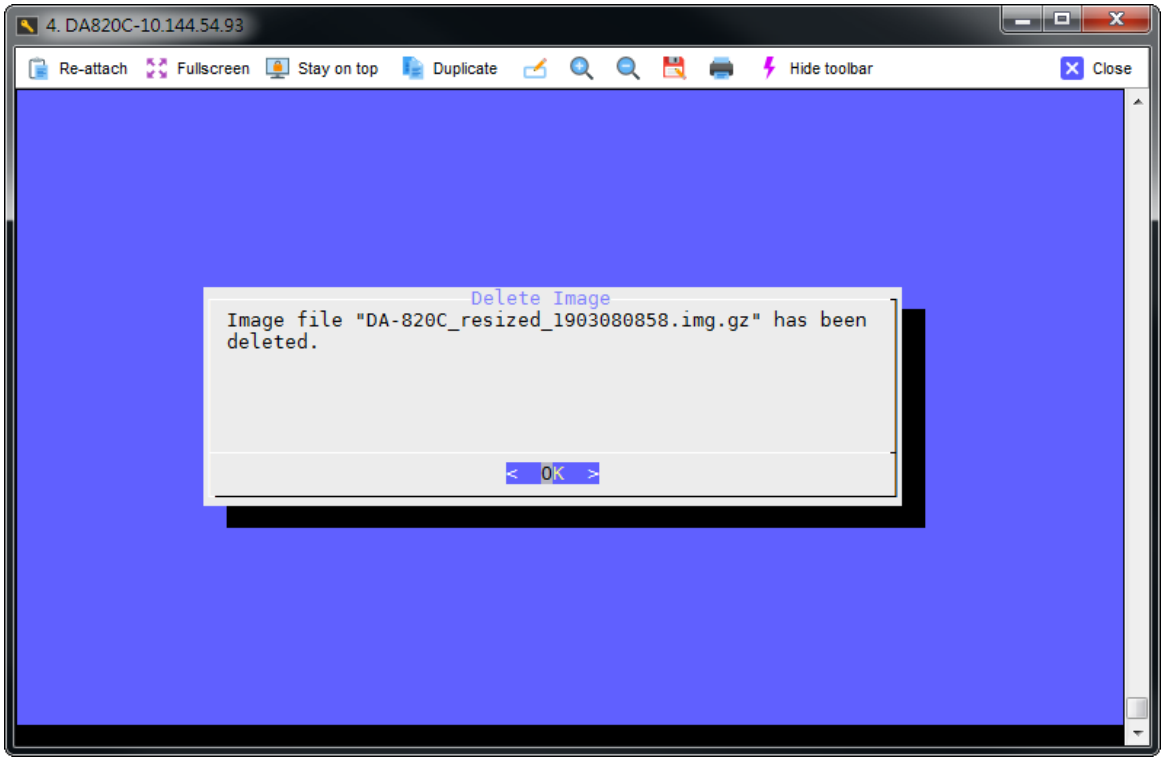


If you need to delete backup images in live USB storage, to select “Delete image” in advanced mode to delete image.






4. DA820C-10.144.54.93
Re-attach
Fullscreen
Stay on top
Duplicate
Hide toolbar
Close
Moxa Image Ultility
Delete Image
Are you sure you want to delete
"DA-820C_resized_1903080858.img.gz"?
< Yes >
< No >


4. DA820C-10.144.54.93
Re-attach
Fullscreen
Stay on top
Duplicate
Hide toolbar
Close
Delete Image
Image file "DA-820C_resized_1903080858.img.gz" has been
deleted.
< OK >

5

DA-820C Series

The following topics are covered in this chapter:

Checking the Linux Version

The program **uname**, which stands for "UNIX Name" and is part of the UNIX operating system, prints the name, version, and other details about the operating system running on the computer. Use the -**a** option to generate a response similar to the one shown below:

```
moxa@Moxa:~$ uname -a
Linux Moxa 4.9.0-6-amd64 #1 SMP Debian 4.9.88-1 (2018-04-29) x86_64 GNU/Linux
```

Checking Moxa's Control Version

The program **kversion** determines the Linux system release version controlled by Moxa. Use the -**a** option to check the built date. The built date format is YYYYMMDDHHmm. You can use kversion to check the released image version in troubleshooting. This example shows the built date, 201901091804, which indicates that it was built on 2019/01/09 at 18:04.

According to EPPROM information on IO board, use the -s option to show the PCBA serial number, and -t option to show PCBA type number(00-CPU Board, 01-Carrier Board, 02-Riser card), and -v option to show PCBA version.

```
root@Moxa:/home/moxa# kversion -h
Usage:
      kversion [OPTIONS]
Options:
      -a, --all            show firmware version and build date
      -s, --serial          show PCBA serial number
      -t, --type           show PCBA type number(00-CPU Board, 01-Carrier Board,
02-Riser card)
      -v, --version         show PCBA version
      -h, --help           show this help page
root@Moxa:/home/moxa# kversion -a
DA-820C firmware version v1.0.0 build 201901091804
root@Moxa:/home/moxa# kversion -s
0
root@Moxa:/home/moxa# kversion -t
01
root@Moxa:/home/moxa# kversion -v
0
moxa@MOXA:~# apt-get clean
```

Device Suspend

The DA-820C supports ACPI S3 (suspend to ram). You should enable option S3 in the BIOS, and then use the "pm-suspend --qurik-s3-bios" command.

```
MOXA:~# pm-suspend --quirk-s3-bios
```

After suspend is in effect, press the power button to wake up the computer.

If you login in as administrator (root) in X windows, you can use **System** → **Shutdown** → **Suspend** to suspend your device.

NOTE This does not work for non-root users.

Some components on Moxa's embedded computer may need to be reset after resuming. You can write a simple script in the directory **/usr/lib/pm-utils/sleep.d/** to complete this procedure. For example, you could create a **script 99serial** for your application.

```
#!/bin/sh

case "$1" in
      hibernate|suspend)
            echo "close AP and tty ports which are opened"
            echo "operations before serial ports suspend"
            ;;
      thaw|resume)
            echo "restart AP"
            echo "operations after serial ports resume"
            ;;
      *) exit $NA
            ;;
esac
```

NOTE If you want to see how to execute the script, start rsyslogd with the command "/etc/init.d/rsyslogd start" and then view the file /var/log/pm-suspend.log.

Wake on LAN

The DA-820C supports wake on LAN, a feature used to wake up a device for suspend (S3) and shutdown (S5).

To check the WOL support on Ethernet port x, type **ethtool enpx**, where "enpx" is the network interface name.

```
root@Moxa:/home/moxa# ethtool enp0s31f6
Settings for enp0s31f6:
      Supported ports: [ TP ]
      Supported link modes:   10baseT/Half 10baseT/Full
                              100baseT/Half 100baseT/Full
                              1000baseT/Full
      Supported pause frame use: No
      Supports auto-negotiation: Yes
      Advertised link modes:  10baseT/Half 10baseT/Full
                              100baseT/Half 100baseT/Full
                              1000baseT/Full
      Advertised pause frame use: No
      Advertised auto-negotiation: Yes
      Speed: 1000Mb/s
      Duplex: Full
      Port: Twisted Pair
      PHYAD: 1
      Transceiver: internal
      Auto-negotiation: on
      MDI-X: on (auto)
      Supports Wake-on: pumbg
      Wake-on: g
      Current message level: 0x00000007 (7)
                             drv probe link
      Link detected: yes
```

As you can see, the default WOL support is g (wake on Magic packet).

If the WOL setting is not g, we suggest that you only enable wake up on magic packet. Modify the default setting with the command " ethtool -s enpx wol g".

The following example illustrates how to wake up on suspend (S3):

1. Moxa's embedded computer

   Enable S3 options in BIOS
   Get its MAC by issuing "ifconfig ethx" (x is the port number)
   Suspend to RAM with command "pm-suspend --quirk-s3-bios"

2. Remote computer

   Issue the command **etherwake -b mac_of_this_device** to wake it up. For example:

```
etherwake -b 00:90:e8:00:d7:38
```

The following example illustrates how to wake up on shutdown (S5):

1. Moxa's embedded computer

   Shut down your computer with "shutdown -h now"

2. Remote computer

   Issue the command **etherwake -b mac_of_this_device** to wake it up. For example:

```
etherwake -b 00:90:e8:00:d7:38
```

ATTENTION

Configure the Ethernet Link Speed for Realtek's Ethernet Chip.

The Ethernet link speed is automatically adjust with the connected switch. However if the switch doesn't support auto-negotiation, you can configure the link speed manually. The Realtek Ethernet chip used vendor's r8168 Ethernet device driver to bring up. It does not fully support ethtool/mii-tools ioctl command call. You can configure the link speed by module options.

EX: modprobe r8168 speed=10 duplex=0 autoneg=0

Or use ethtool to configure the Ethernet speed 10 or 100 Mhz.

Set link speed 10 and half duplex.
EX: ethtool -s eth1 speed 10 duplex half autoneg off

Set link speed 10 and full duplex.
EX: ethtool -s eth1 speed 10 duplex full autoneg off

Set link speed 100 and half duplex.
EX: ethtool -s eth1 speed 100 duplex half autoneg off

Set link speed 100 and full duplex.
EX: ethtool -s eth1 speed 100 duplex full autoneg off

Default Network Interface Name

Debian 9 "Stretch" adopts the systemd predictable network interface naming by default. The network interface name is no long as "ethX". The new interface name depends on the hardware design and physical connections. You may observe different interface naming types, for examples:

1. Names incorporating Firmware/BIOS provided index numbers for on-board devices (example: eno1)
2. Names incorporating Firmware/BIOS provided PCI Express hotplug slot index numbers (example: ens1)
3. Names incorporating physical/geographical location of the connector of the hardware (example: enp2s0)
4. Names incorporating the interfaces' MAC address (example: enx78e7d1ea46da)
5. Classic, unpredictable kernel-native ethX naming (example: eth0)

For more details, you can refer to below:
https://www.freedesktop.org/wiki/Software/systemd/PredictableNetworkInterfaceNames/

The DA-820C default LAN port and network interface name mapping is as below:

LAN port	Network Interface Name
LAN1	enp0s31f6
LAN2	enp9s0
LAN3	enp10s0
LAN4	enp11s0

Renaming the Network Interfaces

You can use the udev rule to rename the network interfaces. For example, if you would like to rename them to classic "ethX" naming, you can create a rules file, **/etc/udev/rules.d/70-persistent-net.rules**, and edit the content as below.

Method 1: Rename interfaces via MAC address

```
SUBSYSTEM=="net", ACTION=="add", ATTR{address}=="00:90:e8:00:d7:38", NAME="eth0"
SUBSYSTEM=="net", ACTION=="add", ATTR{address}=="00:90:e8:00:d7:58", NAME="eth1"
SUBSYSTEM=="net", ACTION=="add", ATTR{address}=="00:90:e8:00:d7:59", NAME="eth2"
SUBSYSTEM=="net", ACTION=="add", ATTR{address}=="00:90:e8:00:d7:5a", NAME="eth3"
```

Method 2: Rename interfaces via PCI bus number

```
SUBSYSTEM=="net", SUBSYSTEMS=="pci", ACTION=="add", KERNELS=="0000:00:1f.6",
NAME="eth0"
SUBSYSTEM=="net", SUBSYSTEMS=="pci", ACTION=="add", KERNELS=="0000:09:00.0",
NAME="eth1"
SUBSYSTEM=="net", SUBSYSTEMS=="pci", ACTION=="add", KERNELS=="0000:0a:00.0",
NAME="eth2"
SUBSYSTEM=="net", SUBSYSTEMS=="pci", ACTION=="add", KERNELS=="0000:0b:00.0",
NAME="eth3"
```

Method 3: Rename interfaces incorporating the interfaces' MAC address

```
SUBSYSTEM=="net", SUBSYSTEMS=="pci", ACTION=="add", IMPORT{builtin}="net_id",
NAME="$env{ID_NET_NAME_MAC}"
```

ATTENTION

Due to DA-820C hardware design limitation, when LAN card is installed on PCI/PCIe slots, the PCI bus id (e.g. 0000:09:00.0) generated by Linux kernel PCI driver may be changed. It causes the network interfaces (e.g. enp9s0) were renamed when LAN cards are installed. If you would like to rename them to fixed interface name, you can create a rules file, /etc/udev/rules.d/70-persistent-net.rules as mentioned above, rename interfaces via MAC address, or names incorporating the interfaces' MAC address.

Getting Product Serial Number

The product information can read by dmidecode. You can use following commands to get these information:

```
moxa@Moxa:~$ sudo dmidecode -t 1
# dmidecode 3.0
Getting SMBIOS data from sysfs.
SMBIOS 3.0.0 present.

Handle 0x0001, DMI type 1, 27 bytes
System Information
       Manufacturer: Moxa
       Product Name: DA-820C
       Version:
       Serial Number: 123456789
       UUID: 12345678-1234-5678-90AB-CDDEEFAABBCC
       Wake-up Type: Power Switch
       SKU Number:
       Family:
```

RTC (Real-time Clock)

The device node is located at **/dev/rtc**. The DA-820C supports standard Linux simple RTC control. You must include **<linux/rtc.h>**.

1. Function: RTC_RD_TIME

   int ioctl(fd, RTC_RD_TIME, struct rtc_time *time);
   Description: read time information from the RTC. It will return the value on argument 3.

2. Function: RTC_SET_TIME

   int ioctl(fd, RTC_SET_TIME, struct rtc_time *time);
   Description: set RTC time. Argument 3 will be passed to RTC.

UART

The normal tty device nodes are **/dev/ttyM0**, **/det/ttyM1**, … in Linux. The DA-820C supports standard Linux termios control with the serial ports. The setinterface utility supports to configure the RS-232/422/485 mode. By default, the serial interface is set to RS-232.

setinterface device-node [interface-no]

device-node: /dev/ttyMn; n = 0,1,2,...
interface-no: [see following table]:

Interface-no	Operation Mode
None	Display current setting
0	RS-232
1	RS-485 2-wires
2	RS-422 or RS-485 4-wires

For example, use the following commands to set /dev/ttyM0 to RS-485 w-wires mode:

```
root@Moxa:/home/moxa# setinterface /dev/ttyM0
Now setting is RS232 mode
root@Moxa:/home/moxa# setinterface /dev/ttyM0 1
Now setting is RS485-2W mode
```

In DA-820C series, only ttyM0 and ttyM1 support RS-232/422/485 mode switching. If you installed a Moxa multiport serial board (e.g. DN-SP08-I-TB/DB), the tty device nodes are auto generated from ttyM2 to ttyM9.

COM port	Device name in/dev	Mode supported
P1	ttyM0	RS-232/422/485
p2	ttyM1	RS-232/422/485

Expansion card COM port	Device name in/dev	Mode supported
P1	ttyM2	RS-232/422/485
P2	ttyM3	RS-232/422/485
P3	ttyM4	RS-232/422/485
P4	ttyM5	RS-232/422/485
P5	ttyM6	RS-232/422/485
P6	ttyM7	RS-232/422/485
P7	ttyM8	RS-232/422/485
P8	ttyM9	RS-232/422/485

In the DA-820C Series computer, the setinterface ultilty saves the current UART mode in **/etc/moxa-configs/setinterface.conf** configuration file, which is used to setup the UART mode during boot up.

```
root@Moxa:/home/moxa# setinterface -s
Save current status to /etc/moxa-configs/setinterface.conf ...
Now setting is RS485-2W mode
Get port 0 mode is 1
Now setting is RS232 mode
Get port 1 mode is 0
Now setting is RS232 mode
Get port 2 mode is 0
Now setting is RS232 mode
Get port 3 mode is 0
Now setting is RS232 mode
Get port 4 mode is 0
Now setting is RS232 mode
```

```
Get port 5 mode is 0
Now setting is RS232 mode
Get port 6 mode is 0
Now setting is RS232 mode
Get port 7 mode is 0
Now setting is RS232 mode
Get port 8 mode is 0
Now setting is RS232 mode
Get port 9 mode is 0
```

The **/etc/moxa-configs/setinterface.conf** configuration file describes the UART default port mode for DA-820C during the boot sequence.

```
root@Moxa:/home/moxa# cat /etc/moxa-configs/setinterface.conf
# configuration file for setup UART mode when boot
# format: PORT[n]=[mode]
# /dev/ttyM0 as PORT0
# mode 0 is RS232, mode 1 is RS485(RS485-2w), mode 2 is RS422(RS485-4w)

PORT0=1
PORT1=0
PORT2=0
PORT3=0
PORT4=0
PORT5=0
PORT6=0
PORT7=0
PORT8=0
PORT9=0
```

Relay

There is one relay output on the front panel of the DA-820C computer. The relay device file is located at **/sys/class/gpio/relay1/value**. The relay can be accessed via **/sys/class/gpio/relay1/value** device node. The following examples show how to control the relay.

Example to turn on/off the relay:

To turn on the relay:
 # echo 1 > /sys/class/gpio/relay1/value

To turn off the relay:
 # echo 0 > /sys/class/gpio/relay1/value

Programmable LED Indicators

There are eight programmable LED indicators on the front panel of the DA-820C. The programmable LED device file is located at **/dev/pled**. Each LED can be accessed via **/dev/pled** device node. These are the examples to control the programmable LED indicators.

Example to turn on/off the LED:

To turn on the first LED and turn off the rest LED:
 # echo 10000000 > /dev/pled

To turn off all the LEDs:
 # echo 00000000 > /dev/pled

To turn on the second LED and turn off other LEDs:
 # echo 01000000 > /dev/pled

To turn on the first and the last LEDs and turn off the second and third LEDs:
 # echo 11010000 > /dev/pled

The sysfs file system also provides interfaces to control a programmable LED indicator. The device file is located at /sys/class/gpio/pled[n]/value, n is LED indicators index from 1 to 8.

Example to turn on/off the LED by sysfs:

To turn on the first LED:
 # echo 1 > /sys/class/gpio/pled1/value

To turn off the first LED:
 # echo 0 > /sys/class/gpio/pled1/value

To turn on the last LED:
 # echo 1 > /sys/class/gpio/pled8/value

To turn off the last LED:
 # echo 0 > /sys/class/gpio/pled8/value

Digital I/O

Digital Output channels can be set to high or low. The channels are controlled by /sys file. Following is the DI/DO number mapping table.

Function	Sysfs file
DI1	/sys/class/gpio/di1/value
DI2	/sys/class/gpio/di2/value
DI3	/sys/class/gpio/di3/value
DI4	/sys/class/gpio/di4/value
DI5	/sys/class/gpio/di5/value
DI6	/sys/class/gpio/di6/value
DO1	/sys/class/gpio/do1/value
DO2	/sys/class/gpio/do2/value

Return All the GPIO sysfs files have been exported by /etc/systemd/system/multi-user.target.wants/da820c_platform_init.service at boot sequence. You don't need to export the GPIO entry. You can use echo to control the GPIO directly in a shell program.

To set DO1 to status low

```
moxa@Moxa:~# sudo echo 0 > /sys/class/gpio/do1/value
```

To set DO1 to status high

```
moxa@Moxa:~# sudo echo 1 > /sys/class/gpio/do1/value
```

To read the DI1 status

```
moxa@Moxa:~# sudo cat /sys/class/gpio/di1/value
```

In case of setting the value of DO1 in C

```
sprint(buf, "/sys/class/gpio/do%d/value", gpio);

fd = open(buf, O_WRONLY);

// Set GPIO high status
write(fd, "1", 1);
// Set GPIO low status
```

```
write(fd, "0", 1);

close(fd);
```

In case of getting the value of DI1 in C

```
sprint(buf, "/sys/class/gpio/di%d/value", gpio);

fd = open(buf, O_RDONLY);

read(fd, &value, 1);

if(value == '0')
{
    // Current GPIO status low
}
else
{
    // Current GPIO status high
}
close(fd);
```

WDT (Watch Dog Timer)

Introduction

The WDT works like a watchdog function, and can be enabled or disabled. When the WDT function is enabled and the application does not acknowledge it, the system will reboot. The watchdog driver is load with default timeout 60 seconds. The watchdog application should acknowledge in 60 seconds.

How the WDT Works

Debian project supports a watchdog daemon. The watchdog daemon checks if your system is still working. If programs are no longer executed it will perform the hard reset of the system. The standard watchdog driver and package have been installed in the DA-820C.

To enable it, first modify the **/etc/watchdog.conf** to remove the '**#**' in front of the "**watchdog-device**" setting

```
…
watchdog-device = /dev/watchdog
…
```

Then enable the **watchdog** service via **systemctl**

```
moxa@Moxa:~$ sudo systemctl enable watchdog
```

The watchdog configure file is located in **/etc/watchdog.conf**.

The acknowledgement interval can be set to any number between 2 seconds and 58 seconds. Currently we configure the watchdog daemon to acknowledge in 29 seconds because the watchdog daemon suggests to acknowledge twice before the watchdog timer timeout and the daemon might sleep. The realtime option is to lock itself into memory, so it is never swapped out to prevent the delay of watchdog acknowledge. You can configure this file to enable the watchdog as your system requirement. The priority set the schedule priority for realtime mode.

```
…
interval             = 29
```

```
realtime              = yes
priority              = 1
…
```

If you want to monitor the execution of some daemon, e.g.: crond, you can configure the pidfile to monitor the daemon.

```
pidfile=/var/run/crond.pid
```

If you want to monitor network is alive, you can configure the ping and interface to check the gateway in the local area network. For example, the system will reset if we ping the gateway 10.144.7.254 from eth0 over 5 times.

```
ping=10.144.7.254
interface=eth0
```

If you want to remove it from systemd, you can use this command:

```
moxa@Moxa:~$ sudo systemctl disable watchdog
```

Check the watchdog daemon status.

```
moxa@Moxa:~# sudo systemctl status watchdog
```

The Watchdog Device IOCTL Commands

IOCTL	**WDIOC_GETSUPPORT**
Description	This returns the support of the card itself
Input	None
Output	(struct watchdog_info *) arg
Return	On success, return 0. Otherwise, return < 0 value.
IOCTL	**WDIOC_GETSTATUS**
Description	This returns the status of the card
Input	None
Output	(int *)arg
Return	On success, return 0. Otherwise, return < 0 value.
IOCTL	**WDIOC_GETBOOTSTATUS**
Description	This returns the status of the card that was reported at bootup.
Input	None
Output	(int *)arg)
Return	On success, return 0. Otherwise, return < 0 value.
IOCTL	**WDIOC_SETOPTIONS**
Description	This lets you set the options of the card. You can either enable or disable the card this way.
Input	None
Output	(int *)arg)
Return	On success, return 0. Otherwise, return < 0 value.
IOCTL	**WDIOC_KEEPALIVE**
Description	This pings the card to tell it not to reset your computer.
Input	None
Output	None
Return	On success, return 0. Otherwise, return < 0 value.
IOCTL	**WDIOC_SETTIMEOUT**
Description	Set the watchdog timeout
Input	arg: 2 ~ 255 seconds
Output	None
Return	On success, return 0. Otherwise, return < 0 value.
IOCTL	**WDIOC_GETTIMEOUT**
Description	Get the current watchdog timeout.

Input	None
Output	arg: 2 ~ 255 seconds
Return	On success, return 0. Otherwise, return < 0 value.

Examples

The example file **watchdog-simple.c** acks the watchdog every 10 seconds.

```
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <fcntl.h>

int main(void)
{
	int fd = open("/dev/watchdog", O_WRONLY);
	int ret = 0;
	if (fd == -1) {
		perror("watchdog");
		exit(EXIT_FAILURE);
	}
	while (1) {
		ret = write(fd, "\0", 1);
		if (ret != 1) {
			ret = -1;
			break;
		}
		sleep(10);
	}
	close(fd);
	return ret;
}
```

DA-820C Series Expansion Module

Installing the HSR/PRP Module Utility

Linux Platform

1. Install the HSR/PRP module on Moxa's embedded computer. There are two ways to install the HSR/PRP utility and status daemon on Moxa's embedded computers. One way is to install it from Moxa's APT server. Another is off-line installation. You can choose one of these methods to install the HSR/PRP packages on Moxa's embedded computers.
2. Add acpi_enforce_resources=lax in the **grub** option.

```
root@Moxa:~# sudo vi /etc/default/grub
  ...
# add acpi_enforce_resources=lax to GRUB_CMDLINE_LINUX_DEFAULT line
GRUB_CMDLINE_LINUX_DEFAULT="quiet acpi_enforce_resources=lax"
root@Moxa:~# sudo update-grub
root@Moxa:~# sudo reboot
```

3. Install the packages required for the HSR/PRP module utility.

Online installation

```
root@Moxa:~# vi /etc/apt/sources.list
...
# Add Moxa's apt server
deb mirror://debian.moxa.com/debian/mirrors stretch main contrib non-free

root@Moxa:~# apt-get update && apt install moxa-hsrprp-tools
```

Offline Installation

1. Upload or copy the packages to target moxa-hsrprp-tools_<version>+deb9_amd64.deb
2. Install the HSR/PRP utility and status daemon. For example:

```
root@Moxa:~# dpkg -i moxa-hsrprp-tools_1.0.1+deb9_amd64.deb
```

3. After that, the HSR/PRP utility and status daemon should be installed. The status daemon should be running with default configure.

```
root@Moxa:/home/moxa#  ps -aux | grep mxhsrprpd
root      371  0.3  0.0 153820   100 ?       Sl   13:54   0:45
/usr/sbin/mxhsrprpd -b /dev/i2c-0 -t 2 -B
```

The HSR/PRP Module Utility

Linux Platform

The HSR/PRP module utility is locate at /usr/sbin

Program	Usage
mxhsrprpd	Used to set the HSR or PRP mode and correct the HSR/PRP protocol statistics.
mxprpinfo	Depending on the setting in the mxhsrprpd, this program is used to either get HSR/PRP protocol statistics or set the HSR/PRP mode.
chk-mx-prp-card	Reports on the current HSR/PRP module name on the Ethernet
mxprpsuper	Sends the HSR/PRP supervision frame Moxa HSR/PRP module.
mxprpalarm	This program is called by mxhsrprpd when the Link status has changed. The action should be modified by user.

mxhsrprpd

This HSR/PRP module status daemon is used to set the HSR or PRP mode and correct the HSR/PRP protocol statistics.

```
root@Moxa:/home/moxa# mxhsrprpd -h
Usage:
      -h: Show this information.
      -B: Run daemon in the background
      -b: SMBUS device, default is /dev/i2c-0
      -t: HSR/PRP Status update period. Default is 2 second.
      -m: configure to prp or hsr mode, default is prp mode.
            The argurement is [index]:[mode]
            index range from 0~7, mode 0 is prp, mode 1 is hsr.
            Ex: Set card 0 to hsr mode, card 1 to prp mode.
            # mxhsrprpd -t 2 -m 0:1,1:0
```

Usage example: Set status update interval to 2 seconds and set Card 0 to HSR mode and Card 1 to PRP mode.

```
root@Moxa:/home/moxa# mxhsrprpd -h
Usage:
      -h: Show this information.
      -B: Run daemon in the background
      -b: SMBUS device, default is /dev/i2c-0
      -t: HSR/PRP Status update period. Default is 2 second.
      -m: configure to prp or hsr mode, default is prp mode.
            The argurement is [index]:[mode]
            index range from 0~7, mode 0 is prp, mode 1 is hsr.
            Ex: Set card 0 to hsr mode, card 1 to prp mode.
            # mxhsrprpd -t 2 -m 0:1,1:0
```

Usage example: Set the PRP mode on card index 1

mxprpinfo

This program is used to get the HSR/PRP protocol statistics or set the HSR/PRP mode.

```
root@Moxa:/home/moxa# mxprpinfo -h
Usage:
      -h: Show this information.
      -l: Show link status.
      -s: Show link speed.
      -c: Show counter.
      -m: Show current mode. 0 is PRP, 1 is HSR.
      -p: Set current mode. 0 is PRP, 1 is HSR.
      -i: Specify card index, the range is 0~7
```

Usage example: Get the mode, counters, and link status of the two HSR/PRP modules.

```
root@Moxa:/home/moxa# mxprpinfo -m
Mode:
index:0
mode:0
index:2
mode:1

root@Moxa:/home/moxa# mxprpinfo -l
Link Status:
index:0
link_status_i:0
link_status_a:0
link_status_b:0
index:2
link_status_i:0
link_status_a:0
link_status_b:0

root@Moxa:/home/moxa# mxprpinfo -i 0 -c
Counters:
index:0

port: interlink
rx_good_octets:0
```

```
[...]

port: lana
rx_good_octets:0
[...]

port: lanb
rx_good_octets:0
[...]

root@Moxa:/home/moxa# mxprpinfo -i 2 -c
Counters:
index:2

port: interlink
rx_good_octets:0
[...]

port: lana
rx_good_octets:0
[...]

port: lanb
rx_good_octets:0
[...]
```

Usage example: Set the PRP mode on card index 0 & Set the HSR mode on card index 2

```
root@Moxa:/home/moxa# mxprpinfo -i 0 -p 0
Set HSR/PRP mode:
index:0, curr: 0, new:0

root@Moxa:/home/moxa# mxprpinfo -i 2 -p 1
Set HSR/PRP mode:
index:2, curr: 1, new:1

root@Moxa:/home/moxa# mxprpinfo -m
Mode:
index:0
mode:0
index:2
mode:1
```

chk-mx-prp-card

This program is used to get the Ethernet interface name of the Moxa HSR/PRP module.

```
root@Moxa:/home/moxa# chk-mx-prp-card
Searching Moxa HSR/PRP card...
enp4s0 - pcibus id is 0000:04:00.0
enp8s0 - pcibus id is 0000:08:00.0
```

mxprpsuper

This program is used to send the HSR/PRP supervision frame. You can get the interface name using the chk-mx-prp-card program.

```
root@Moxa:/home/moxa# mxprpsuper
This program is used to send a supervision frame to the Moxa HSR/PRP Card.

Usage:
       -h: Show this information.
       -t: send HSR/PRP supervision frame period.
          Default is 2 second.
       -m: send HSR or PRP supervision frame mode.
          0 is PRP; 1 is HSR. Default is 0.
       -i: interface name, Ex.: eth0.
          Execute chk-mx-prp-card to list supported interface.
       -a: Destination MAC address last byte.
          The range is 00~FF. Default is 00
```

Usage example: Send the PRP supervision frame to eth14 in the time period of 2 seconds. The program should run in the background.

```
root@Moxa:/home/moxa# mxprpsuper -t 2 -m 0 -i enp4s0 &
```

mxprpalarm

This script is called by the mxhsrprpd program when the HSR/PRP link status changes. The script can be modified as necessary.

```
root@Moxa:/home/moxa# cat /usr/sbin/mxprpalarm
#!/bin/bash
#
# This shell script will call by mxhsrprpd.
# You can design custom action in this script. Please search 'TODO'
#
# $1 - Card index, 0~7.
# $2 - Event name
#        li: inter link status
#        la: lan A link status
#        lb: lab B link status
# $3 - Event argument
#

IS_DA720=$(cat /sys/class/dmi/id/product_name | grep "DA-720")

function run_alarm_link_down()
{
       echo "Card $1 $2 Link down"
       #
       # Relay trig
       #if [ x"$IS_DA720" != x"" ]; then
       #      echo 0 >/sys/class/gpio/relay1/value
       #fi
       #

       #
```

```
	# Snmp trap example, for more information man snmptrap
	# snmptrap -v 2c -c $SNMPTRAP_COMMUNITY $SNMPTRAP_IP "" $ALERT_OID
$ALERT_SRC_OID i 0 1>/dev/null 2>/dev/null
	#

	#
	# Sendmail example, the pop/smtp server should be configured first. For
more information man mutt
	# mutt -s "Card $1 $2 Link down" user@example.com </dev/null
	#

	#
	# For the other Relay/Alarm device, Ex: Moxa iologik series.
	#
}

function run_alarm_link_up()
{
echo "Card $1 $2 Link up"
	#
	# Relay trig
	#if [ x"$IS_DA720" != x"" ]; then
	#	echo 1 >/sys/class/gpio/relay1/value
	#fi
	#

	#
	# Snmp trap, for more information man snmptrap
	# snmptrap -v 2c -c $SNMPTRAP_COMMUNITY $SNMPTRAP_IP "" $ALERT_OID
$ALERT_SRC_OID i 0 1>/dev/null 2>/dev/null
	#

	#
	# Sendmail, the pop/smtp server should be configured first. For more
information man mutt
	# mutt -s "Card $1 $2 Link up" user@example.com </dev/null
	#

	#
	# For the other Relay/Alarm device, Ex: Moxa iologik series.
	#
}



#
# Process inter link status event
#
if [ x"$2" = x"li" ]; then
	#TODO: do something when event occur
	if [ x"$3" = x"0" ]; then
		run_alarm_link_down $1 "Interlink"
	elif [ x"$3" = x"1" ]; then
		run_alarm_link_up $1 "Interlink"
	fi
elif [ x"$2" = x"la" ]; then
```

```
#
# Process Lan A link status event
#
       #TODO: do something when event occur
       if [ x"$3" = x"0" ]; then
              run_alarm_link_down $1 "LAN A"
       elif [ x"$3" = x"1" ]; then
              run_alarm_link_up $1 "LAN A"
       fi
elif [ x"$2" = x"lb" ]; then
#
# Process Lan B link status event
#
       #TODO: do something when event occur
       if [ x"$3" = x"0" ]; then
              run_alarm_link_down $1 "LAN B"
       elif [ x"$3" = x"1" ]; then
              run_alarm_link_up $1 "LAN B"
       fi
fi
```

Installing the IRIG-B Module Utility

IRIG-B modules Introduction

The DA-IRIG-B module has two input ports. One input port is Fiber port and the other is IRIG-B Port 1. The fiber port only supports TTL signal type. The IRIG-B Port 1 supports TTL or DIFF type IRIG-B signal. You can select ether the Fiber port or the Port 1 as the IRIG-B sync time source. After you select the time source, the ServiceSyncTime daemon is the time sync utility for syncing the system time with the IRIG-B module.

There are 4 output ports of the DA-IRIG-B module. It's suitable to be an IRIG-B time server to output the IRIG-B time signal from one of these ports. Each of the output port supports to select the IRIG-B out signal from Fiber port, IRIG-B Port 1, Internal RTC or PPS encoded module. The output signal supports TTL or DIFF signal type. The mxIrigUtil utility is a command-line tool for user to configure the IRIG-B signal in their application.

Install the IRIG-B utility

The default DA-820C Linux operating system has pre-install the IRIG-B driver and time sync daemon. You can check the installation by this command.

```
root@Moxa:/home/moxa# dpkg -l | grep moxa-irigb-tools
ii  moxa-irigb-tools              1.0.0+deb9                   amd64
Utility for controlling DA-IRIG-B expansion module
```

Online Installation

```
root@Moxa:~# vi /etc/apt/sources.list
...
# Add Moxa's apt server
deb mirror://debian.moxa.com/debian/mirrors stretch main contrib non-free

root@Moxa:~# apt-get update && apt install moxa-irigb-tools
```

Offline Installation

1. Upload or copy the packages to target moxa-hsrprp-tools_<version>+deb9_amd64.deb
2. Install the IRIG-B utility and status daemon

```
root@Moxa:~# dpkg -i moxa-irigb-tools_1.0.0+deb9_amd64.deb
```

3. After that, the IRIG-B utility and status service and daemon should be installed. The status service should be running with default configure. And time sync daemon should be running with default configure.

```
root@Moxa:/home/moxa# systemctl status mx_irigb.service
● mx_irigb.service - Moxa DA-IRIG-B daemon service
   Loaded: loaded (/lib/systemd/system/mx_irigb.service; enabled; vendor
preset: enabled)
   Active: active (exited) since Wed 2019-04-03 10:05:17 CST; 14min ago
  Process: 347 ExecStart=/usr/sbin/mx_irigb.sh start (code=exited,
status=0/SUCCESS)
 Main PID: 347 (code=exited, status=0/SUCCESS)
    Tasks: 1 (limit: 4915)
   CGroup: /system.slice/mx_irigb.service
           └─386 /usr/sbin/ServiceSyncTime -t 1 -i 10 -B

Apr 03 10:05:17 Moxa systemd[1]: Starting Moxa DA-IRIG-B daemon service...
Apr 03 10:05:17 Moxa systemd[1]: Started Moxa DA-IRIG-B daemon service.

root@Moxa:/home/moxa# ps aux | grep ServiceSyncTime
root      386  0.0  0.0  17332   192 ?       S    10:05   0:00
/usr/sbin/ServiceSyncTime -t 1 -i 10 -B
```

IRIG-B Time-synchronization Daemon

The usage of IRIG-B time sync daemon.

```
root@Moxa:/home/moxa# ServiceSyncTime -h
Found the IRIG-B module, Hardware ID = 7
IRIG-B time sync daemon.
Usage: ServiceSyncTime -t [signal type] -I -d -i [Time sync interval] -s [Time
Source] -p [Parity check mode] -B
  -t - [signal type]
     0 - TTL
     1 - DIFF
     default value is 1
  -I - Inverse the input signal
  -s - [Time Source] The sync source from FREERUN(Internal RTC), Fiber or IRIG-B
port
     0 - FREERUN(Internal RTC) module
     1 - Fiber port
     2 - IRIG-B port
     default value is 2
  -i - [Time sync interval] The time interval in seconds to sync the IRIG-B time
into system time.
     1 ~ 86400 Time sync interval. Default is 10 second.
  -p - [Parity check mode] Set the parity bit
     0: EVEN
     1: ODD
```

```
      2: NONE
      default value is 0
   -B - Run daemon in the background
Usage example: Enable to sync time from IRIG-B Port 1, in TTL signal type every
10 seconds. The input signals is not inverse.
root@Moxa:~# ServiceSyncTime -t 0 -i 10
```

Usage example: Enable to sync time from IRIG-B Port 1 in DIFF signal type every 10 seconds. The input signal is not inversed. The ServiceSyncTime runs in foreground.

```
root@Moxa:/home/moxa# ServiceSyncTime -t 0 -i 10
```

Usage example: Enable to sync time from FIBER port in TTL signal type every 10 seconds. The input signal is not inversed. The ServiceSyncTime runs in foreground.

```
root@Moxa:/home/moxa# ServiceSyncTime -t 0 -s 2 -I 10
```

Usage example: Enable to sync time from IRIG-B Port 1 in DIFF signal type every 10 seconds. Set to use ODD parity check mode. The ServiceSyncTime runs in foreground.

```
root@Moxa:/home/moxa# ServiceSyncTime -t 1 -s 2 -i 10 -I
```

Configure the IRIG-B time sync daemon.

The IRIG-B time sync daemon is managed by /usr/sbin/mx_irigb.sh script. The default configure, MX_IRIGB_OPTS, is written in /usr/sbin/mx_irigb.sh.

```
root@Moxa:/home/moxa# vim /usr/sbin/mx_irigb.sh
...
MX_IRIGB_SERVICESYNCTIME_OPTS="-t 1 -i 10 -B"
...
```

After you configure it, the daemon should restarted.

```
root@Moxa:/home/moxa# systemctl restart mx_irigb.service
```

IRIG-B Utility

The usage of IRIG-B utility. Notice that DIO functions only available on DA-IRIGB-4DIO-PCI104 module.

```
root@Moxa:/home/moxa# mxIrigUtil -h
Get/set Moxa DA-IRIGB utility
Usage: mxIrigUtil -f function_id [-p parameters] [-c] [-h]
   Show the utility information if no argument apply.
   -h: Show this information.
   -c: Indicate the n-the IRIG-B Card.
   -f: Pass function id argument to execute specify functionality
   -p: Parameters for each function, use comma to pass multiple variables

For example: Set IRIG-B RTC Time 2014/01/01 03:25:00
      mxIrigUtil -f 2 -p 2014,1,1,3,25,0

Function description list:
      0: Get Hardware ID
      1: Get IRIG-B RTC Time
      2: Set IRIG-B RTC Time
        -p year,month,day,hour,minute,second
             [2000-2099],[1-12],[1-31],[0-23],[0-59],[0-59]
        default value is 2014,01,01,00,00,00 if no argument.
      3: Get IRIG-B RTC Sync. Source
```

```
4: Set IRIG-B RTC Sync. Source
  -p Source
       Source: 0: FreeRun In (Internal RTC)
              1: Port 0/Fiber In, 2: Port 1 In
  default value is 2 if no argument.
5: Get IRIG-B Signal Status
  -p Source
       Source: 1: Port 0/Fiber In, 2: Port 1 In
  default value is 2 if no argument.
6: Get IRIG-B Input Parity Check Mode
  -p Source
       Source: 1: Port 0/Fiber In, 2: Port 1 In
  default value is 2 if no argument.
7: Set IRIG-B Input Parity Check Mode
  -p Source,Mode
       Source: 1: Port 0/Fiber In, 2: Port 1 In
       Mode:   0: Even, 1: Odd, 2: None
  default value is 2,0 if no argument.
8: Get IRIG-B Output Parity Check Mode
9: Set IRIG-B Output Parity Check Mode
  -p Mode
       Mode:   0: Even, 1: Odd
  default value is 0 if no argument.
10: Get Pulse per second width(ms)
11: Set Pulse per second width(ms)
  -p Width
       [0-999] (width: 0-999 ms)
  default value is 0 if no argument.
12: Get input signal type
  -p Port
       Port:   0: Port 0/Fiber, 1: Port 1
  default value is 1 if no argument.
13: Set input signal type
  -p Port,Type,Inverse
       Port:   0: Port 0/Fiber, 1: Port 1
       Type:   0: TTL, 1: Differential
      Inverse:0: No inverse, 1: Inverse
  default value is 1,1,0 if no argument.
14: Get output signal type
  -p Port
       [1-4] (output port[1-4])
 default value is 1 if no argument.
15: Set output signal type
  -p Port,Type,Mode,Inverse
       Port:   output port[1-4]
       Type:   0: TTL, 1=Differential
       Mode:   0: From Port 0/Fiber Input Port, 1: From Port 1 Input
              2: From IRIG-B encode(Internal RTC), 3: From PPS encode
       Inverse:0: No inverse, 1: Inverse)
  default value is 1,1,2,0 if no argument.
16: Get Digital Output
  -p Port
       [0-3] (digital output port 0-3)
  default value is 0 if no argument.
17: Set Digital Output
```

```
        -p Port,Level
             Port:   [0-3] (digital output port[0-3])
             Level:  [0-1] Output level)
        default value is 0,0 if no argument.
      18: Get Digital Input
        -p Port
             [0-3] (digital input port 0-3)
        default value is 0 if no argument.
      19: Get FPGA firmware build date
```

Usage example: Get the IRIG-B module hardware ID

```
root@Moxa:/home/moxa# mxIrigUtil -f 0
Hardware ID = 7 (DA_IRIGB_S)
```

Usage example: Get the IRIG-B module internal RTC time

```
root@Moxa:/home/moxa# mxIrigUtil -f 1
Hardware ID = 7 (DA_IRIGB_S)
IRIG-B RTC = 2019/4/3 10:42:32.486130880
TZ = +0, TQ = 0, LSP = 0, LS = 0, DSP = 0, DST = 0
```

Usage example: Set IRIG-B module internal RTC time to 2014/11/19 11:19:50.

```
root@Moxa:/home/moxa# mxIrigUtil -f 2 -p 2014,11,19,11,19,50
Hardware ID = 7 (DA_IRIGB_S)
Set IRIGB RTC = 2014/11/19 11:19:50
root@Moxa:/home/moxa# mxIrigUtil -f 1
Hardware ID = 7 (DA_IRIGB_S)
IRIG-B RTC = 2014/11/19 11:19:54.472544720
TZ = +0, TQ = 0, LSP = 0, LS = 0, DSP = 0, DST = 0
```

Usage example: Get IRIG-B module time sync source setting. It this case the time source is IRIG-B in Port.

```
root@Moxa:/home/moxa# mxIrigUtil -f 3
Hardware ID = 7 (DA_IRIGB_S)
Sync. Source = 2 (Port 1 In)
```

Usage example: Set IRIG-B module time sync source.

```
root@Moxa:/home/moxa# mxIrigUtil -f 4 -p 1
Hardware ID = 7 (DA_IRIGB_S)
Set Sync. Source = 1
```

Usage example: Get IRIG-B signal status.

```
root@Moxa:/home/moxa# mxIrigUtil -f 5 -p 1
Hardware ID = 7 (DA_IRIGB_S)
Port 0/Fiber In Signal status = 1(Off Line)
root@Moxa:/home/moxa# mxIrigUtil -f 5 -p 2
Hardware ID = 7 (DA_IRIGB_S)
Port 1 In Signal status = 1(Off Line)
```

Software Components

This packages table is based on Firmware v1.0 and build date 19061314.

```
$ apt list --installed
adduser/stable,now 3.115 all [installed]
apt/now 1.4.8 amd64 [installed,upgradable to: 1.4.9]
apt-listchanges/stable,now 3.10 all [installed]
apt-utils/now 1.4.8 amd64 [installed,upgradable to: 1.4.9]
base-files/now 9.9+deb9u4 amd64 [installed,upgradable to: 9.9+deb9u9]
base-passwd/stable,now 3.5.43 amd64 [installed]
bash/stable,now 4.4-5 amd64 [installed]
bash-completion/stable,now 1:2.1-4.3 all [installed]
bind9-host/stable,now 1:9.10.3.dfsg.P4-12.3+deb9u4 amd64 [installed,upgradable to: 1:9.10.3.dfsg.P4-
12.3+deb9u5]
binutils/stable,now 2.28-5 amd64 [installed,automatic]
bsdmainutils/stable,now 9.0.12+nmu1 amd64 [installed]
bsdutils/stable,stable,now 1:2.29.2-1+deb9u1 amd64 [installed]
build-essential/stable,now 12.3 amd64 [installed]
busybox/stable,now 1:1.22.0-19+b3 amd64 [installed]
bzip2/stable,now 1.0.6-8.1 amd64 [installed]
ca-certificates/now 20161130+nmu1 all [installed,upgradable to: 20170717]
console-setup/stable,now 1.164 all [installed]
console-setup-linux/stable,now 1.164 all [installed,automatic]
coreutils/stable,now 8.26-3 amd64 [installed]
cpio/stable,now 2.11+dfsg-6 amd64 [installed]
cpp/stable,now 4:6.3.0-4 amd64 [installed,automatic]
cpp-6/stable,stable,now 6.3.0-18+deb9u1 amd64 [installed,automatic]
cracklib-runtime/stable,now 2.9.2-5 amd64 [installed,automatic]
cron/stable,now 3.0pl1-128+deb9u1 amd64 [installed]
da820c-base-system/now 1.0.7+deb9 amd64 [installed,local]
da820c-modules/now 1.0.7+deb9 amd64 [installed,local]
da820c-moxa-firmware/now 1.0.7+deb9 amd64 [installed,local]
dash/stable,now 0.5.8-2.4 amd64 [installed]
dbus/stable,now 1.10.26-0+deb9u1 amd64 [installed,upgradable to: 1.10.28-0+deb9u1]
debconf/stable,now 1.5.61 all [installed]
debconf-i18n/stable,now 1.5.61 all [installed]
debian-archive-keyring/stable,now 2017.5 all [installed]
debian-faq/stable,now 8.1 all [installed]
debianutils/stable,now 4.8.1.1 amd64 [installed]
dh-python/stable,now 2.20170125 all [installed,automatic]
dictionaries-common/stable,now 1.27.2 all [installed,automatic]
diffutils/stable,now 1:3.5-3 amd64 [installed]
discover/now 2.1.2-7.1 amd64 [installed,upgradable to: 2.1.2-7.1+deb9u1]
discover-data/stable,now 2.2013.01.11 all [installed,automatic]
distro-info-data/stable,now 0.36 all [installed,automatic]
dmidecode/stable,now 3.0-4 amd64 [installed]
dmsetup/stable,now 2:1.02.137-2 amd64 [installed]
doc-debian/stable,now 6.4 all [installed]
dpkg/now 1.18.24 amd64 [installed,upgradable to: 1.18.25]
dpkg-dev/stable,now 1.18.25 all [installed,automatic]
e2fslibs/stable,now 1.43.4-2 amd64 [installed]
e2fsprogs/stable,now 1.43.4-2 amd64 [installed]
efibootmgr/stable,now 14-2 amd64 [installed,automatic]
eject/stable,now 2.1.5+deb1+cvs20081104-13.2 amd64 [installed]
```

```
emacsen-common/stable,now 2.0.8 all [installed,automatic]
ethtool/stable,now 1:4.8-1+b1 amd64 [installed]
fakeroot/stable,now 1.21-3.1 amd64 [installed,automatic]
file/stable,now 1:5.30-1+deb9u1 amd64 [installed,upgradable to: 1:5.30-1+deb9u2]
findutils/stable,now 4.6.0+git+20161106-2 amd64 [installed]
g++/stable,now 4:6.3.0-4 amd64 [installed,automatic]
g++-6/stable,stable,now 6.3.0-18+deb9u1 amd64 [installed,automatic]
g++-6-multilib/stable,stable,now 6.3.0-18+deb9u1 amd64 [installed,automatic]
g++-multilib/stable,now 4:6.3.0-4 amd64 [installed]
gcc/stable,now 4:6.3.0-4 amd64 [installed,automatic]
gcc-6/stable,stable,now 6.3.0-18+deb9u1 amd64 [installed,automatic]
gcc-6-base/stable,stable,now 6.3.0-18+deb9u1 amd64 [installed]
gcc-6-multilib/stable,stable,now 6.3.0-18+deb9u1 amd64 [installed,automatic]
gcc-multilib/stable,now 4:6.3.0-4 amd64 [installed,automatic]
geoip-database/stable,now 20170512-1 all [installed,automatic]
gettext-base/stable,now 0.19.8.1-2 amd64 [installed]
gnupg/now 2.1.18-8~deb9u1 amd64 [installed,upgradable to: 2.1.18-8~deb9u4]
gnupg-agent/now 2.1.18-8~deb9u1 amd64 [installed,upgradable to: 2.1.18-8~deb9u4]
gpgv/now 2.1.18-8~deb9u1 amd64 [installed,upgradable to: 2.1.18-8~deb9u4]
grep/stable,now 2.27-2 amd64 [installed]
groff-base/stable,now 1.22.3-9 amd64 [installed]
grub-common/now 2.02~beta3-5 amd64 [installed,upgradable to: 2.02~beta3-5+deb9u1]
grub-efi-amd64/now 2.02~beta3-5 amd64 [installed,upgradable to: 2.02~beta3-5+deb9u1]
grub-efi-amd64-bin/now 2.02~beta3-5 amd64 [installed,upgradable to: 2.02~beta3-5+deb9u1]
grub2-common/now 2.02~beta3-5 amd64 [installed,upgradable to: 2.02~beta3-5+deb9u1]
gzip/stable,now 1.6-5+b1 amd64 [installed]
hdparm/now 9.51+ds-1 amd64 [installed,upgradable to: 9.51+ds-1+deb9u1]
hostname/stable,now 3.18+b1 amd64 [installed]
i2c-tools/stable,now 3.1.2-3 amd64 [installed]
iamerican/stable,now 3.4.00-5 all [installed,automatic]
ibritish/stable,now 3.4.00-5 all [installed,automatic]
ienglish-common/stable,now 3.4.00-5 all [installed,automatic]
ifupdown/stable,now 0.8.19 amd64 [installed]
init/stable,now 1.48 amd64 [installed]
init-system-helpers/stable,now 1.48 all [installed]
initramfs-tools/stable,now 0.130 all [installed]
initramfs-tools-core/stable,now 0.130 all [installed,automatic]
installation-report/stable,now 2.62 all [installed]
iproute2/stable,now 4.9.0-1+deb9u1 amd64 [installed]
iptables/stable,now 1.6.0+snapshot20161117-6 amd64 [installed]
iputils-ping/stable,now 3:20161105-1 amd64 [installed]
isc-dhcp-client/stable,stable,now 4.3.5-3+deb9u1 amd64 [installed]
isc-dhcp-common/stable,stable,now 4.3.5-3+deb9u1 amd64 [installed]
iso-codes/stable,now 3.75-1 all [installed,automatic]
ispell/stable,now 3.4.00-5 amd64 [installed,automatic]
kbd/stable,now 2.0.3-2+b1 amd64 [installed,automatic]
keyboard-configuration/stable,now 1.164 all [installed]
keyutils/stable,now 1.5.9-9 amd64 [installed,automatic]
klibc-utils/stable,now 2.0.4-9 amd64 [installed,automatic]
kmod/stable,now 23-2 amd64 [installed]
krb5-locales/stable,now 1.15-1+deb9u1 all [installed]
laptop-detect/stable,now 0.13.8 amd64 [installed]
less/stable,now 481-2.1 amd64 [installed]
lib32asan3/stable,stable,now 6.3.0-18+deb9u1 amd64 [installed,automatic]
lib32atomic1/stable,stable,now 6.3.0-18+deb9u1 amd64 [installed,automatic]
```

```
lib32cilkrts5/stable,stable,now 6.3.0-18+deb9u1 amd64 [installed,automatic]
lib32gcc-6-dev/stable,stable,now 6.3.0-18+deb9u1 amd64 [installed,automatic]
lib32gcc1/stable,stable,now 1:6.3.0-18+deb9u1 amd64 [installed,automatic]
lib32gomp1/stable,stable,now 6.3.0-18+deb9u1 amd64 [installed,automatic]
lib32itm1/stable,stable,now 6.3.0-18+deb9u1 amd64 [installed,automatic]
lib32mpx2/stable,stable,now 6.3.0-18+deb9u1 amd64 [installed,automatic]
lib32quadmath0/stable,stable,now 6.3.0-18+deb9u1 amd64 [installed,automatic]
lib32stdc++-6-dev/stable,stable,now 6.3.0-18+deb9u1 amd64 [installed,automatic]
lib32stdc++6/stable,stable,now 6.3.0-18+deb9u1 amd64 [installed,automatic]
lib32ubsan0/stable,stable,now 6.3.0-18+deb9u1 amd64 [installed,automatic]
libacl1/stable,now 2.2.52-3+b1 amd64 [installed]
libalgorithm-diff-perl/stable,now 1.19.03-1 all [installed,automatic]
libalgorithm-diff-xs-perl/stable,now 0.04-4+b2 amd64 [installed,automatic]
libalgorithm-merge-perl/stable,now 0.08-3 all [installed,automatic]
libapparmor1/stable,now 2.11.0-3+deb9u2 amd64 [installed]
libapt-inst2.0/now 1.4.8 amd64 [installed,upgradable to: 1.4.9]
libapt-pkg5.0/now 1.4.8 amd64 [installed,upgradable to: 1.4.9]
libasan3/stable,stable,now 6.3.0-18+deb9u1 amd64 [installed,automatic]
libassuan0/stable,now 2.4.3-2 amd64 [installed]
libatomic1/stable,stable,now 6.3.0-18+deb9u1 amd64 [installed,automatic]
libattr1/stable,now 1:2.4.47-2+b2 amd64 [installed]
libaudit-common/stable,now 1:2.6.7-2 all [installed]
libaudit1/stable,now 1:2.6.7-2 amd64 [installed]
libbind9-140/stable,now 1:9.10.3.dfsg.P4-12.3+deb9u4 amd64 [installed,upgradable to: 1:9.10.3.dfsg.P4-
12.3+deb9u5]
libblkid1/stable,stable,now 2.29.2-1+deb9u1 amd64 [installed]
libbsd0/stable,now 0.8.3-1 amd64 [installed]
libbz2-1.0/stable,now 1.0.6-8.1 amd64 [installed]
libc-bin/now 2.24-11+deb9u3 amd64 [installed,upgradable to: 2.24-11+deb9u4]
libc-dev-bin/stable,now 2.24-11+deb9u4 amd64 [installed,automatic]
libc-l10n/now 2.24-11+deb9u3 all [installed,upgradable to: 2.24-11+deb9u4]
libc6/stable,now 2.24-11+deb9u4 amd64 [installed]
libc6-dev/stable,now 2.24-11+deb9u4 amd64 [installed,automatic]
libc6-dev-i386/stable,now 2.24-11+deb9u4 amd64 [installed,automatic]
libc6-dev-x32/stable,now 2.24-11+deb9u4 amd64 [installed,automatic]
libc6-i386/stable,now 2.24-11+deb9u4 amd64 [installed,automatic]
libc6-x32/stable,now 2.24-11+deb9u4 amd64 [installed,automatic]
libcap-ng0/stable,now 0.7.7-3+b1 amd64 [installed]
libcap2/stable,now 1:2.25-1 amd64 [installed]
libcc1-0/stable,stable,now 6.3.0-18+deb9u1 amd64 [installed,automatic]
libcilkrts5/stable,stable,now 6.3.0-18+deb9u1 amd64 [installed,automatic]
libclass-isa-perl/stable,now 0.36-5 all [installed]
libcomerr2/stable,now 1.43.4-2 amd64 [installed]
libcrack2/stable,now 2.9.2-5 amd64 [installed,automatic]
libcryptsetup4/stable,now 2:1.7.3-4 amd64 [installed]
libcurl3-gnutls/now 7.52.1-5+deb9u5 amd64 [installed,upgradable to: 7.52.1-5+deb9u9]
libdb5.3/stable,now 5.3.28-12+deb9u1 amd64 [installed]
libdbus-1-3/stable,now 1.10.26-0+deb9u1 amd64 [installed,upgradable to: 1.10.28-0+deb9u1]
libdebconfclient0/stable,now 0.227 amd64 [installed]
libdevmapper1.02.1/stable,now 2:1.02.137-2 amd64 [installed]
libdiscover2/now 2.1.2-7.1 amd64 [installed,upgradable to: 2.1.2-7.1+deb9u1]
libdns-export162/stable,now 1:9.10.3.dfsg.P4-12.3+deb9u4 amd64 [installed,upgradable to:
1:9.10.3.dfsg.P4-12.3+deb9u5]
libdns162/stable,now 1:9.10.3.dfsg.P4-12.3+deb9u4 amd64 [installed,upgradable to: 1:9.10.3.dfsg.P4-
12.3+deb9u5]
```

```
libdpkg-perl/stable,now 1.18.25 all [installed,automatic]
libedit2/stable,now 3.1-20160903-3 amd64 [installed,automatic]
libefiboot1/stable,now 30-2 amd64 [installed,automatic]
libefivar1/stable,now 30-2 amd64 [installed,automatic]
libelf1/stable,now 0.168-1 amd64 [installed]
libestr0/stable,now 0.1.10-2 amd64 [installed]
libevent-2.0-5/stable,now 2.0.21-stable-3 amd64 [installed,automatic]
libexpat1/stable,stable,now 2.2.0-2+deb9u1 amd64 [installed,automatic]
libfakeroot/stable,now 1.21-3.1 amd64 [installed,automatic]
libfastjson4/stable,now 0.99.4-1 amd64 [installed]
libfdisk1/stable,stable,now 2.29.2-1+deb9u1 amd64 [installed]
libffi6/stable,now 3.2.1-6 amd64 [installed]
libfile-fcntllock-perl/stable,now 0.22-3+b2 amd64 [installed,automatic]
libfreetype6/stable,now 2.6.3-3.2 amd64 [installed,automatic]
libfuse2/now 2.9.7-1 amd64 [installed,upgradable to: 2.9.7-1+deb9u2]
libgcc-6-dev/stable,stable,now 6.3.0-18+deb9u1 amd64 [installed,automatic]
libgcc1/stable,stable,now 1:6.3.0-18+deb9u1 amd64 [installed]
libgcrypt20/now 1.7.6-2+deb9u2 amd64 [installed,upgradable to: 1.7.6-2+deb9u3]
libgdbm3/stable,now 1.8.3-14 amd64 [installed]
libgeoip1/stable,now 1.6.9-4 amd64 [installed,automatic]
libgmp10/stable,now 2:6.1.2+dfsg-1 amd64 [installed]
libgnutls30/now 3.5.8-5+deb9u3 amd64 [installed,upgradable to: 3.5.8-5+deb9u4]
libgomp1/stable,stable,now 6.3.0-18+deb9u1 amd64 [installed,automatic]
libgpg-error0/stable,now 1.26-2 amd64 [installed]
libgpm2/stable,now 1.20.4-6.2+b1 amd64 [installed,automatic]
libgssapi-krb5-2/stable,now 1.15-1+deb9u1 amd64 [installed,automatic]
libhogweed4/stable,now 3.3-1+b2 amd64 [installed]
libi2c-dev/stable,now 3.1.2-3 all [installed]
libicu57/stable,stable,now 57.1-6+deb9u2 amd64 [installed,automatic]
libidn11/stable,now 1.33-1 amd64 [installed]
libidn2-0/stable,stable,now 0.16-1+deb9u1 amd64 [installed]
libip4tc0/stable,now 1.6.0+snapshot20161117-6 amd64 [installed]
libip6tc0/stable,now 1.6.0+snapshot20161117-6 amd64 [installed]
libiptc0/stable,now 1.6.0+snapshot20161117-6 amd64 [installed]
libisc-export160/stable,now 1:9.10.3.dfsg.P4-12.3+deb9u4 amd64 [installed,upgradable to:
1:9.10.3.dfsg.P4-12.3+deb9u5]
libisc160/stable,now 1:9.10.3.dfsg.P4-12.3+deb9u4 amd64 [installed,upgradable to: 1:9.10.3.dfsg.P4-
12.3+deb9u5]
libisccc140/stable,now 1:9.10.3.dfsg.P4-12.3+deb9u4 amd64 [installed,upgradable to: 1:9.10.3.dfsg.P4-
12.3+deb9u5]
libisccfg140/stable,now 1:9.10.3.dfsg.P4-12.3+deb9u4 amd64 [installed,upgradable to: 1:9.10.3.dfsg.P4-
12.3+deb9u5]
libisl15/stable,now 0.18-1 amd64 [installed,automatic]
libitm1/stable,stable,now 6.3.0-18+deb9u1 amd64 [installed,automatic]
libk5crypto3/stable,now 1.15-1+deb9u1 amd64 [installed,automatic]
libkeyutils1/stable,now 1.5.9-9 amd64 [installed,automatic]
libklibc/stable,now 2.0.4-9 amd64 [installed,automatic]
libkmod2/stable,now 23-2 amd64 [installed]
libkrb5-3/stable,now 1.15-1+deb9u1 amd64 [installed,automatic]
libkrb5support0/stable,now 1.15-1+deb9u1 amd64 [installed,automatic]
libksba8/stable,now 1.3.5-2 amd64 [installed]
libldap-2.4-2/now 2.4.44+dfsg-5+deb9u1 amd64 [installed,upgradable to: 2.4.44+dfsg-5+deb9u2]
libldap-common/now 2.4.44+dfsg-5+deb9u1 all [installed,upgradable to: 2.4.44+dfsg-5+deb9u2]
liblocale-gettext-perl/stable,now 1.07-3+b1 amd64 [installed]
liblockfile-bin/stable,now 1.14-1+b1 amd64 [installed]
```

```
liblogging-stdlog0/stable,now 1.0.5-2+b2 amd64 [installed]
liblognorm5/stable,now 2.0.1-1.1+b1 amd64 [installed]
liblsan0/stable,stable,now 6.3.0-18+deb9u1 amd64 [installed,automatic]
liblwres141/stable,now 1:9.10.3.dfsg.P4-12.3+deb9u4 amd64 [installed,upgradable to: 1:9.10.3.dfsg.P4-
12.3+deb9u5]
liblz4-1/stable,now 0.0~r131-2+b1 amd64 [installed]
liblzma5/stable,now 5.2.2-1.2+b1 amd64 [installed]
libmagic-mgc/stable,now 1:5.30-1+deb9u1 amd64 [installed,upgradable to: 1:5.30-1+deb9u2]
libmagic1/stable,now 1:5.30-1+deb9u1 amd64 [installed,upgradable to: 1:5.30-1+deb9u2]
libmnl0/stable,now 1.0.4-2 amd64 [installed]
libmount1/stable,stable,now 2.29.2-1+deb9u1 amd64 [installed]
libmpc3/stable,now 1.0.3-1+b2 amd64 [installed,automatic]
libmpdec2/stable,now 2.4.2-1 amd64 [installed,automatic]
libmpfr4/stable,now 3.1.5-1 amd64 [installed,automatic]
libmpx2/stable,stable,now 6.3.0-18+deb9u1 amd64 [installed,automatic]
libncurses5/stable,now 6.0+20161126-1+deb9u2 amd64 [installed]
libncursesw5/stable,now 6.0+20161126-1+deb9u2 amd64 [installed]
libnetfilter-conntrack3/stable,now 1.0.6-2 amd64 [installed]
libnettle6/stable,now 3.3-1+b2 amd64 [installed]
libnewt0.52/stable,now 0.52.19-1+b1 amd64 [installed]
libnfnetlink0/stable,now 1.0.1-3 amd64 [installed]
libnfsidmap2/stable,now 0.25-5.1 amd64 [installed,automatic]
libnghttp2-14/stable,now 1.18.1-1 amd64 [installed,automatic]
libnpth0/stable,now 1.3-1 amd64 [installed]
libp11-kit0/stable,now 0.23.3-2 amd64 [installed]
libpam-cracklib/stable,now 1.1.8-3.6 amd64 [installed]
libpam-modules/stable,now 1.1.8-3.6 amd64 [installed]
libpam-modules-bin/stable,now 1.1.8-3.6 amd64 [installed]
libpam-runtime/stable,now 1.1.8-3.6 all [installed]
libpam-systemd/now 232-25+deb9u3 amd64 [installed,upgradable to: 232-25+deb9u11]
libpam0g/stable,now 1.1.8-3.6 amd64 [installed]
libpcap0.8/stable,now 1.8.1-3 amd64 [installed,automatic]
libpci3/stable,now 1:3.5.2-1 amd64 [installed,automatic]
libpcre3/stable,now 2:8.39-3 amd64 [installed]
libperl5.24/now 5.24.1-3+deb9u3 amd64 [installed,upgradable to: 5.24.1-3+deb9u5]
libpipeline1/stable,now 1.4.1-2 amd64 [installed]
libpng16-16/stable,now 1.6.28-1 amd64 [installed,upgradable to: 1.6.28-1+deb9u1]
libpopt0/stable,now 1.16-10+b2 amd64 [installed]
libprocps6/now 2:3.3.12-3 amd64 [installed,upgradable to: 2:3.3.12-3+deb9u1]
libpsl5/stable,now 0.17.0-3 amd64 [installed]
libpython-stdlib/stable,now 2.7.13-2 amd64 [installed,automatic]
libpython2.7-minimal/now 2.7.13-2+deb9u2 amd64 [installed,upgradable to: 2.7.13-2+deb9u3]
libpython2.7-stdlib/now 2.7.13-2+deb9u2 amd64 [installed,upgradable to: 2.7.13-2+deb9u3]
libpython3-stdlib/stable,now 3.5.3-1 amd64 [installed,automatic]
libpython3.5-minimal/now 3.5.3-1 amd64 [installed,upgradable to: 3.5.3-1+deb9u1]
libpython3.5-stdlib/now 3.5.3-1 amd64 [installed,upgradable to: 3.5.3-1+deb9u1]
libquadmath0/stable,stable,now 6.3.0-18+deb9u1 amd64 [installed,automatic]
libreadline7/stable,now 7.0-3 amd64 [installed]
librtmp1/stable,now 2.4+20151223.gitfa8646d.1-1+b1 amd64 [installed,automatic]
libsasl2-2/stable,now 2.1.27~101-g0780600+dfsg-3 amd64 [installed,automatic]
libsasl2-modules/stable,now 2.1.27~101-g0780600+dfsg-3 amd64 [installed,automatic]
libsasl2-modules-db/stable,now 2.1.27~101-g0780600+dfsg-3 amd64 [installed,automatic]
libseccomp2/now 2.3.1-2.1 amd64 [installed,upgradable to: 2.3.1-2.1+deb9u1]
libselinux1/stable,now 2.6-3+b3 amd64 [installed]
libsemanage-common/stable,now 2.6-2 all [installed]
```

```
libsemanage1/stable,now 2.6-2 amd64 [installed]
libsensors4/stable,now 1:3.4.0-4 amd64 [installed,automatic]
libsepol1/stable,now 2.6-2 amd64 [installed]
libslang2/stable,now 2.3.1-5 amd64 [installed]
libsmartcols1/stable,stable,now 2.29.2-1+deb9u1 amd64 [installed]
libsnmp-base/stable,stable,now 5.7.3+dfsg-1.7+deb9u1 all [installed,automatic]
libsnmp30/stable,stable,now 5.7.3+dfsg-1.7+deb9u1 amd64 [installed,automatic]
libsqlite3-0/stable,now 3.16.2-5+deb9u1 amd64 [installed]
libss2/stable,now 1.43.4-2 amd64 [installed]
libssh2-1/now 1.7.0-1 amd64 [installed,upgradable to: 1.7.0-1+deb9u1]
libssl1.0.2/now 1.0.2l-2+deb9u3 amd64 [installed,upgradable to: 1.0.2r-1~deb9u1]
libssl1.1/now 1.1.0f-3+deb9u2 amd64 [installed,upgradable to: 1.1.0j-1~deb9u1]
libstdc++-6-dev/stable,stable,now 6.3.0-18+deb9u1 amd64 [installed,automatic]
libstdc++6/stable,stable,now 6.3.0-18+deb9u1 amd64 [installed]
libswitch-perl/stable,now 2.17-2 all [installed]
libsystemd0/now 232-25+deb9u3 amd64 [installed,upgradable to: 232-25+deb9u11]
libtasn1-6/stable,stable,now 4.10-1.1+deb9u1 amd64 [installed]
libtext-charwidth-perl/stable,now 0.04-7+b5 amd64 [installed]
libtext-iconv-perl/stable,now 1.7-5+b4 amd64 [installed]
libtext-wrapi18n-perl/stable,now 0.06-7.1 all [installed]
libtinfo5/stable,now 6.0+20161126-1+deb9u2 amd64 [installed]
libtirpc1/stable,now 0.2.5-1.2+deb9u1 amd64 [installed,automatic]
libtsan0/stable,stable,now 6.3.0-18+deb9u1 amd64 [installed,automatic]
libubsan0/stable,stable,now 6.3.0-18+deb9u1 amd64 [installed,automatic]
libudev1/now 232-25+deb9u3 amd64 [installed,upgradable to: 232-25+deb9u11]
libunistring0/stable,now 0.9.6+really0.9.3-0.1 amd64 [installed]
libusb-0.1-4/stable,now 2:0.1.12-30 amd64 [installed,automatic]
libustr-1.0-1/stable,now 1.0.4-6 amd64 [installed]
libuuid1/stable,stable,now 2.29.2-1+deb9u1 amd64 [installed]
libwrap0/stable,now 7.6.q-26 amd64 [installed,automatic]
libx11-6/now 2:1.6.4-3 amd64 [installed,upgradable to: 2:1.6.4-3+deb9u1]
libx11-data/now 2:1.6.4-3 all [installed,upgradable to: 2:1.6.4-3+deb9u1]
libx32asan3/stable,stable,now 6.3.0-18+deb9u1 amd64 [installed,automatic]
libx32atomic1/stable,stable,now 6.3.0-18+deb9u1 amd64 [installed,automatic]
libx32cilkrts5/stable,stable,now 6.3.0-18+deb9u1 amd64 [installed,automatic]
libx32gcc-6-dev/stable,stable,now 6.3.0-18+deb9u1 amd64 [installed,automatic]
libx32gcc1/stable,stable,now 1:6.3.0-18+deb9u1 amd64 [installed,automatic]
libx32gomp1/stable,stable,now 6.3.0-18+deb9u1 amd64 [installed,automatic]
libx32itm1/stable,stable,now 6.3.0-18+deb9u1 amd64 [installed,automatic]
libx32quadmath0/stable,stable,now 6.3.0-18+deb9u1 amd64 [installed,automatic]
libx32stdc++-6-dev/stable,stable,now 6.3.0-18+deb9u1 amd64 [installed,automatic]
libx32stdc++6/stable,stable,now 6.3.0-18+deb9u1 amd64 [installed,automatic]
libx32ubsan0/stable,stable,now 6.3.0-18+deb9u1 amd64 [installed,automatic]
libx86-1/stable,now 1.1+ds1-10.2 amd64 [installed,automatic]
libxapian30/now 1.4.3-2 amd64 [installed,upgradable to: 1.4.3-2+deb9u3]
libxau6/stable,now 1:1.0.8-1 amd64 [installed,automatic]
libxcb1/stable,now 1.12-1 amd64 [installed,automatic]
libxdmcp6/stable,now 1:1.1.2-3 amd64 [installed,automatic]
libxext6/stable,now 2:1.3.3-1+b2 amd64 [installed,automatic]
libxml2/stable,stable,now 2.9.4+dfsg1-2.2+deb9u2 amd64 [installed,automatic]
libxmuu1/stable,now 2:1.1.2-2 amd64 [installed,automatic]
libxtables12/stable,now 1.6.0+snapshot20161117-6 amd64 [installed]
linux-base/stable,now 4.5 all [installed,automatic]
linux-image-4.9.0-6-amd64/now 4.9.88-1 amd64 [installed,upgradable to: 4.9.88-1+deb9u1]
linux-image-amd64/now 4.9+80+deb9u4 amd64 [installed,upgradable to: 4.9+80+deb9u7]
```

```
linux-libc-dev/stable,now 4.9.168-1+deb9u2 amd64 [installed,automatic]
lm-sensors/stable,now 1:3.4.0-4 amd64 [installed]
locales/now 2.24-11+deb9u3 all [installed,upgradable to: 2.24-11+deb9u4]
login/stable,now 1:4.4-4.1 amd64 [installed]
logrotate/stable,now 3.11.0-0.1 amd64 [installed]
lsb-base/stable,now 9.20161125 all [installed]
lsb-release/stable,now 9.20161125 all [installed,automatic]
lsof/stable,now 4.89+dfsg-0.1 amd64 [installed]
make/stable,now 4.1-9.1 amd64 [installed,automatic]
man-db/stable,now 2.7.6.1-2 amd64 [installed]
manpages/stable,now 4.10-2 all [installed]
manpages-dev/stable,now 4.10-2 all [installed,automatic]
mawk/stable,now 1.3.3-17+b3 amd64 [installed]
mime-support/stable,now 3.60 all [installed]
mount/stable,stable,now 2.29.2-1+deb9u1 amd64 [installed]
moxa-archive-keyring/stretch,now 2018.4.11 all [installed]
moxa-hsrprp-tools/now 1.0.2+deb9 amd64 [installed,local]
moxa-irigb-tools/now 1.0.3+deb9 amd64 [installed,local]
multiarch-support/now 2.24-11+deb9u3 amd64 [installed,upgradable to: 2.24-11+deb9u4]
nano/stable,now 2.7.4-1 amd64 [installed]
ncurses-base/stable,now 6.0+20161126-1+deb9u2 all [installed]
ncurses-bin/stable,now 6.0+20161126-1+deb9u2 amd64 [installed]
ncurses-term/stable,now 6.0+20161126-1+deb9u2 all [installed]
net-tools/stable,now 1.60+git20161116.90da8a0-1 amd64 [installed]
netbase/stable,now 5.4 all [installed]
netcat-traditional/stable,now 1.10-41+b1 amd64 [installed]
nfs-common/stable,now 1:1.3.4-2.1 amd64 [installed]
ntpdate/stable,now 1:4.2.8p10+dfsg-3+deb9u2 amd64 [installed]
openssh-client/now 1:7.4p1-10+deb9u3 amd64 [installed,upgradable to: 1:7.4p1-10+deb9u6]
openssh-server/now 1:7.4p1-10+deb9u3 amd64 [installed,upgradable to: 1:7.4p1-10+deb9u6]
openssh-sftp-server/now 1:7.4p1-10+deb9u3 amd64 [installed,upgradable to: 1:7.4p1-10+deb9u6]
openssl/now 1.1.0f-3+deb9u2 amd64 [installed,upgradable to: 1.1.0j-1~deb9u1]
os-prober/stable,now 1.76~deb9u1 amd64 [installed,automatic]
passwd/stable,now 1:4.4-4.1 amd64 [installed]
patch/stable,now 2.7.5-1+deb9u1 amd64 [installed,automatic]
pciutils/stable,now 1:3.5.2-1 amd64 [installed]
perl/now 5.24.1-3+deb9u3 amd64 [installed,upgradable to: 5.24.1-3+deb9u5]
perl-base/now 5.24.1-3+deb9u3 amd64 [installed,upgradable to: 5.24.1-3+deb9u5]
perl-modules-5.24/now 5.24.1-3+deb9u3 all [installed,upgradable to: 5.24.1-3+deb9u5]
pinentry-curses/stable,now 1.0.0-2 amd64 [installed]
pm-utils/stable,now 1.4.1-17 all [installed]
powermgmt-base/stable,now 1.31+nmu1 all [installed,automatic]
ppp/stable,now 2.4.7-1+4 amd64 [installed]
procps/now 2:3.3.12-3 amd64 [installed,upgradable to: 2:3.3.12-3+deb9u1]
python/stable,now 2.7.13-2 amd64 [installed]
python-apt-common/stable,now 1.4.0~beta3 all [installed,automatic]
python-minimal/stable,now 2.7.13-2 amd64 [installed]
python2.7/now 2.7.13-2+deb9u2 amd64 [installed,upgradable to: 2.7.13-2+deb9u3]
python2.7-minimal/now 2.7.13-2+deb9u2 amd64 [installed,upgradable to: 2.7.13-2+deb9u3]
python3/stable,now 3.5.3-1 amd64 [installed,automatic]
python3-apt/stable,now 1.4.0~beta3 amd64 [installed,automatic]
python3-chardet/stable,now 2.3.0-2 all [installed,automatic]
python3-debian/stable,now 0.1.30 all [installed,automatic]
python3-debianbts/stable,now 2.6.1 all [installed,automatic]
python3-httplib2/stable,now 0.9.2+dfsg-1 all [installed,automatic]
```

```
python3-minimal/stable,now 3.5.3-1 amd64 [installed,automatic]
python3-pkg-resources/stable,now 33.1.1-1 all [installed,automatic]
python3-pycurl/stable,now 7.43.0-2 amd64 [installed,automatic]
python3-pysimplesoap/stable,now 1.16-2 all [installed,automatic]
python3-reportbug/now 7.1.7+deb9u1 all [installed,upgradable to: 7.1.7+deb9u2]
python3-requests/stable,now 2.12.4-1 all [installed,automatic]
python3-six/stable,now 1.10.0-3 all [installed,automatic]
python3-urllib3/stable,now 1.19.1-1 all [installed,automatic]
python3.5/now 3.5.3-1 amd64 [installed,upgradable to: 3.5.3-1+deb9u1]
python3.5-minimal/now 3.5.3-1 amd64 [installed,upgradable to: 3.5.3-1+deb9u1]
read-edid/stable,now 3.0.2-1+b1 amd64 [installed,automatic]
readline-common/stable,now 7.0-3 all [installed]
rename/stable,now 0.20-4 all [installed,automatic]
reportbug/now 7.1.7+deb9u1 all [installed,upgradable to: 7.1.7+deb9u2]
rpcbind/stable,now 0.2.3-0.6 amd64 [installed,automatic]
rsyslog/stable,now 8.24.0-1 amd64 [installed]
sed/stable,now 4.4-1 amd64 [installed]
sensible-utils/stable,stable,now 0.0.9+deb9u1 all [installed]
sgml-base/stable,now 1.29 all [installed,automatic]
snmp/stable,stable,now 5.7.3+dfsg-1.7+deb9u1 amd64 [installed]
snmpd/stable,stable,now 5.7.3+dfsg-1.7+deb9u1 amd64 [installed]
sudo/stable,now 1.8.19p1-2.1 amd64 [installed]
systemd/now 232-25+deb9u3 amd64 [installed,upgradable to: 232-25+deb9u11]
systemd-sysv/now 232-25+deb9u3 amd64 [installed,upgradable to: 232-25+deb9u11]
sysvinit-utils/stable,now 2.88dsf-59.9 amd64 [installed]
tar/stable,now 1.29b-1.1 amd64 [installed]
task-english/stable,now 3.39 all [installed]
task-ssh-server/stable,now 3.39 all [installed]
tasksel/stable,now 3.39 all [installed]
tasksel-data/stable,now 3.39 all [installed]
tcpd/stable,now 7.6.q-26 amd64 [installed,automatic]
telnet/stable,now 0.17-41 amd64 [installed]
traceroute/stable,now 1:2.1.0-2 amd64 [installed]
tzdata/now 2018d-0+deb9u1 all [installed,upgradable to: 2019a-0+deb9u1]
ucf/stable,now 3.0036 all [installed]
udev/now 232-25+deb9u3 amd64 [installed,upgradable to: 232-25+deb9u11]
util-linux/stable,stable,now 2.29.2-1+deb9u1 amd64 [installed]
util-linux-locales/stable,stable,now 2.29.2-1+deb9u1 all [installed,automatic]
vbetool/stable,now 1.1-4 amd64 [installed,automatic]
vim/stable,now 2:8.0.0197-4+deb9u1 amd64 [installed]
vim-common/stable,now 2:8.0.0197-4+deb9u1 all [installed]
vim-runtime/stable,now 2:8.0.0197-4+deb9u1 all [installed,automatic]
vim-tiny/stable,now 2:8.0.0197-4+deb9u1 amd64 [installed]
wamerican/stable,now 7.1-1 all [installed]
watchdog/stable,now 5.15-2 amd64 [installed]
wget/now 1.18-5+deb9u1 amd64 [installed,upgradable to: 1.18-5+deb9u3]
whiptail/stable,now 0.52.19-1+b1 amd64 [installed]
xauth/stable,now 1:1.0.9-1+b2 amd64 [installed,automatic]
xkb-data/stable,now 2.19-1+deb9u1 all [installed,automatic]
xml-core/stable,now 0.17 all [installed,automatic]
xxd/stable,now 2:8.0.0197-4+deb9u1 amd64 [installed]
xz-utils/stable,now 5.2.2-1.2+b1 amd64 [installed]
zlib1g/stable,now 1:1.2.8.dfsg-5 amd64 [installed]
```