ioThinx 4530 Series Linux User Manual

Version 2.1, June 2023

www.moxa.com/products

ioThinx 4530 Series Linux User Manual

The software described in this manual is furnished under a license agreement and may be used only in accordance with the terms of that agreement.

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

Table of Contents

1. Introduction

This user's manual applies to the ioThinx 4530 Series models listed below:

ioThinx 4530 Series

- ioThinx 4533-LX Series

Detailed instructions on configuring advanced settings are covered in Chapters 3 and 4.

2. Getting Started

In this chapter, we describe how to configure basic settings for the ioThinx 4530 controller.

Connecting to the ioThinx 4530 Controller

You will need to use a computer to connect to the ioThinx 4530 controller and to log in through the command line interface. There are two ways to connect: through the serial console port or through the Ethernet port. Refer to the ioThinx 4530 Series Hardware Manual to see how to set up the physical connections.

The default login username and password are:

Username: **moxa**
Password: **moxa**

The username and password are the same for all serial console and SSH remote log in actions. Root account login is disabled until you manually create a password for the account. The user **moxa** is in the **sudo** group so that you can operate system level commands with this user using the **sudo** command. For additional details, see the *Sudo Mechanism* section in chapter 5.

ATTENTION

For security reasons, we recommend that you disable the default user account and create your own user accounts.

Connecting Through the Serial Console

This method is particularly useful when using the computer for the first time. The signal is transmitted over a direct serial connection so you do not need to know either of its two IP addresses in order to connect to the ioThinx 4530 controller. To connect through the serial console, configure your PC's terminal software using the following settings.

Serial Console Port Settings	
Baudrate	115200 bps
Parity	None
Data bits	8
Stop bits	1
Flow control	None
Terminal	VT100

Below we show how to use the terminal software to connect to the ioThinx 4530 controller in a Linux environment and in a Windows environment.

Linux Users

NOTE

These steps apply to the Linux PC you are using to connect to the ioThinx 4530 controller. Do NOT apply these steps to the ioThinx 4530 controller itself.

Take the following steps to connect to the ioThinx 4530 controller from your Linux PC.

1. Install **minicom** from the package repository of your operating system.

   For Centos and Fedora:

   ```
   user@PC1:~# yum -y install minicom
   ```

   For Ubuntu and Debian:

   ```
   user@PC2:~# apt-get install minicom
   ```

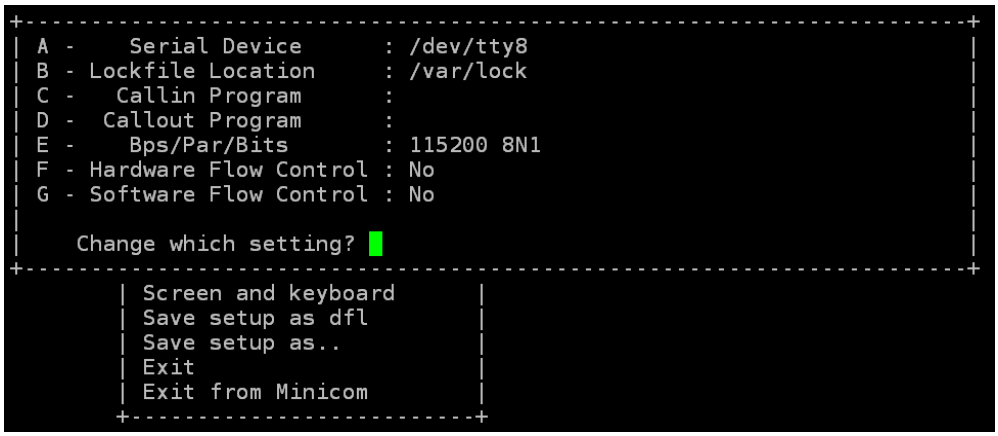
2. Use the **`minicom -s`** command to enter the configuration menu and set up the serial port settings.

   ```
   user@PC1:~# minicom -s
   ```

3. Select **Serial port setup**.

4. Select **A** to change the serial device. Note that you need to know which device node is connected to the ioThinx 4530 controller.

5. Select **E** to configure the port settings according to the **Serial Console Port Settings** table provided.
6. Select **Save setup as dfl** (from the main configuration menu) to use default values.
7. Select **Exit from minicom** (from the configuration menu) to leave the configuration menu.
8. Execute **minicom** after completing the above configurations.

   ```
   user@PC1:~# minicom
   ```

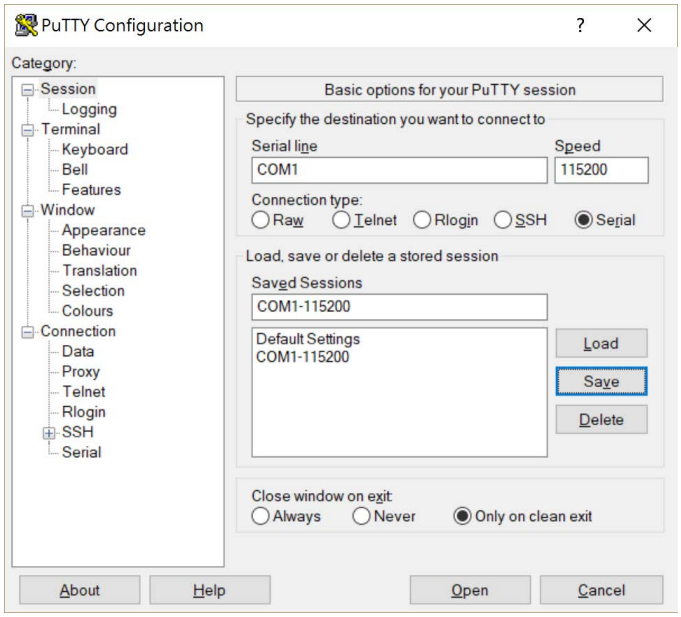
Windows Users

NOTE

These steps apply to the Windows PC you are using to connect to the ioThinx 4530 controller. Do NOT apply these steps to the ioThinx 4530 controller itself.

Take the following steps to connect to the ioThinx 4530 controller from your Windows PC.

1. Download PuTTY http://www.chiark.greenend.org.uk/~sgtatham/putty/download.html to set up a serial connection with the ioThinx 4530 controller in a Windows environment.
2. Once the connection is established, the following window will open.



3. Select the **Serial** connection type and choose settings.

Connecting Through the SSH Console

The ioThinx 4530 controller supports SSH connections over an Ethernet network. Use the following default IP addresses to connect to the ioThinx 4530 controller.

Port	Default IP
LAN 1	192.168.127.254
LAN 2	192.168.126.254

Linux Users

NOTE

These steps apply to the Linux PC you are using to connect to the ioThinx 4530 controller. Do NOT apply these steps to the ioThinx 4530 controller itself. Before you run the ssh command, be sure to configure the IP address of your notebook/PC's Ethernet interface in the range of 192.168.127.0/24 for LAN1 and 192.168.126.0/24 for LAN2.

Use the **ssh** command from a Linux computer to access the ioThinx 4530 controller's LAN1 port.

```
user@PC1:~ ssh moxa@192.168. 127.254
```

Type **yes** to complete the connection.

```
The authenticity of host '192.168. 127.254' 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_
```

ATTENTION

Rekey SSH regularly

In order to secure your system, we suggest doing a regular SSH-rekey, as shown in the following steps:

When prompted for a passphrase, leave the passphrase empty and press enter.

```
moxa@Moxa:~$ cd /etc/ssh
moxa@Moxa:~$ sudo rm -rf
ssh_host_ed25519_key2       ssh_host_ecdsa_key      ssh_host_rsa_key
ssh_host_ed25519_key.pub   ssh_host_ecdsa_key.pub   ssh_host_rsa_key.pub

moxa@Moxa:~$ sudo ssh-keygen -t rsa -f /etc/ssh/ssh_host_rsa_key
moxa@Moxa:~$ sudo ssh-keygen -t dsa -f /etc/ssh/ssh_host_dsa_key
moxa@Moxa:~$ sudo ssh-keygen -t ecdsa -f /etc/ssh/ssh_host_ecdsa_key
moxa@Moxa:~$ sudo /etc/init.d/ssh restart
```

For more information about SSH, refer to the following link.

https://wiki.debian.org/SSH

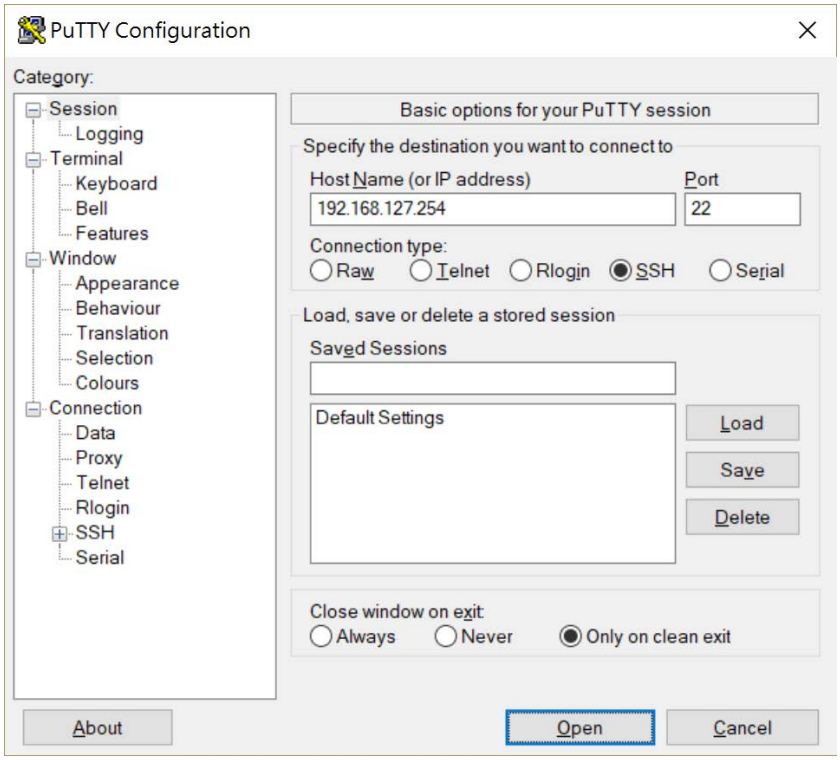
Windows Users

NOTE

These steps apply to the Windows PC you are using to connect to the ioThinx 4530 controller. Do NOT apply these steps to the ioThinx 4530 controller itself.

Take the following steps from your Windows PC.

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 ioThinx 4530 controller in a Windows environment. The following figure shows a simple example of the configuration that is required.



NOTE

The ioThinx 4530 Series only supports SSH connections.

User Account Management

Switching to the Root Account

You can switch to root using **sudo -i** (or **sudo su**). For security reasons, do not operate the **all** commands from the **root** account.

NOTE

Click the following link for more information on the sudo command.

https://wiki.debian.org/sudo

ATTENTION

You might get the **permission denied** message when using pipe or redirect behavior with a non-root account.

You must use 'sudo su –c' to run the command instead of using >, <, >>, <<, etc.

Note: The single quotes around the full command are required.

Creating and Deleting User Accounts

You can use the **useradd** and **userdel** commands to create and delete user accounts. Be sure to reference the main page of these commands to set relevant access privileges for the account. The following example shows how to create a **test1** user in the **sudo** group whose default login shell is **bash** and has home directory at **/home/test1**:

```
moxa@Moxa:~# sudo useradd -m -G sudo -s /bin/bash test1
```

To change the password for test1, use the **passwd** option along with the new password. Retype the password to confirm the change.

```
moxa@Moxa:~# sudo passwd test1
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
```

To delete user test1, use the **userdel** command.

```
moxa@Moxa:~# sudo userdel test1
```

Disabling the Default User Account

ATTENTION

You should first create a user account before you disable the default account.

Use the **passwd** command to lock the default user account so the user **moxa** cannot log in.

```
root@Moxa:~# passwd -l moxa
```

Type the following command to unlock the user moxa:

```
root@Moxa:~# passwd -u moxa
```

Network Settings

Configuring Ethernet Interfaces

After the first login, you can configure the ioThinx 4530 controller's network settings to fit your application better. Note that it is more convenient to manipulate the network interface settings from the serial console than from an SSH login because an SSH connection can disconnect when there are network issues and the connection must be reestablished.

Modifying Network Settings via the Serial Console

In this section, we use the serial console to configure the ioThinx 4530 controller's network settings. Follow the instructions in the *Connecting to the ioThinx 4530 controller* section under *Getting Started* to access the Console Utility of the target computer via the serial Console port and then type **`cd /etc/network`** to change directories.

```
moxa@Moxa:~$ cd /etc/network/
moxa@Moxa:/etc/network/~$
```

Type **`sudo vi interfaces`** to edit the network configuration file in the **vi** editor. You can configure the ioThinx 4530 controller's Ethernet ports to use either **static** or **dynamic** (DHCP) IP addresses.

Setting a Static IP address

To set a static IP address for the ioThinx 4530 controller, use the **`iface`** command to modify the **default gateway**, **address**, **network**, **netmask**, and **broadcast** parameters of the Ethernet interface.

```
# interfaces(5) file used by ifup(8) and ifdown(8)
# Include files from /etc/network/interfaces.d:
source-directory /etc/network/interfaces.d
auto eth0 eth1 lo
iface lo inet loopback
iface eth0 inet static
        address 192.168.127.254
        network 192.168.127.0
        netmask 255.255.255.0
        broadcast 192.168.127.255
iface eth1 inet static
        address 192.168.126.254
        network 192.168.126.0
        netmask 255.255.255.0
        broadcast 192.168.126.255
```

Setting Dynamic IP Addresses:

To configure one or both LAN ports to request an IP address dynamically use the **`dhcp`** option in place of **`static`** in the **`iface`** command, as follows**:**

Default Setting for LAN1	Dynamic Setting using DHCP
iface eth0 inet static address 192.168.127.254 network 192.168.127.0 netmask 255.255.255.0 broadcast 192.168.127.255	iface eth0 inet dhcp

```
iface eth0 inet dhcp
```

System Administration

Querying the Firmware Version

To check the ioThinx 4530 controller's firmware version, type:

```
moxa@Moxa:~$ kversion
ioThinx 4533-LX version 1.0
```

Add the **-a** option to create a full build version:

```
moxa@Moxa:~$ kversion -a
ioThinx 4533-LX version 1.0 Build 19032720

Master Infomation:
45MR Module Scan Time: 0 ms
```

Adjusting the Time

The ioThinx 4530 controller has two time settings. One is the system time, and the other is the RTC (Real Time Clock) time kept by the ioThinx 4530 controller's hardware. Use the **`date`** command to query the current system time or set a new system time. Use the **`hwclock`** command to query the current RTC time or set a new RTC time.

Use the **`date MMDDhhmmYYYY`** command to set the system time:

MM = Month
DD = Date
hhmm = hour and minute

```
moxa@Moxa:~$ sudo date 032123192019
Thu Mar 21 23:19:00 UTC 2019
```

Use the following command to set the RTC time to system time:

```
moxa@Moxa:~$ sudo hwclock -w
moxa@Moxa:~$ sudo hwclock
2019-03-21 02:09:00.628145+0000
```

NOTE

Click the following links for more information on date and time:

https://www.debian.org/doc/manuals/system-administrator/ch-sysadmin-time.html

https://wiki.debian.org/DateTime

Setting the Time Zone

There are two ways to configure the Moxa embedded computer's **timezone**. One is using the **TZ** variable. The other is using the **/etc/localtime** file.

Using the TZ Variable

The format of the TZ environment variable looks like this:

TZ=*<Value>HH[:MM[:SS]][daylight[HH[:MM[:SS]]][,start date[/starttime], enddate[/endtime]]]*

Here are some possible settings for the North American Eastern time zone:

1. **TZ=EST5EDT**
2. **TZ=EST0EDT**
3. **TZ=EST0**

In the first case, the reference time is GMT and the stored time values are correct worldwide. A simple change of the TZ variable can print the local time correctly in any time zone.

In the second case, the reference time is Eastern Standard Time and the only conversion performed is for Daylight Saving Time. Therefore, there is no need to adjust the hardware clock for Daylight Saving Time twice per year.

In the third case, the reference time is always the time reported. You can use this option if the hardware clock on your machine automatically adjusts for Daylight Saving Time or you would like to manually adjust the hardware time twice a year.

```
moxa@Moxa:~$ TZ= EST5EDT
moxa@Moxa:~$ export TZ
```

You must include the TZ setting in the **/etc/rc.local** file. The timezone setting will be activated when you restart the computer.

The following table lists other possible values for the TZ environment variable:

Hours From Greenwich Mean Time (GMT)	Value	Description
0	GMT	Greenwich Mean Time
+1	ECT	European Central Time
+2	EET	European Eastern Time
+2	ART	
+3	EAT	Saudi Arabia
+3.5	MET	Iran
+4	NET	
+5	PLT	West Asia
+5.5	IST	India
+6	BST	Central Asia
+7	VST	Bangkok
+8	CTT	China
+9	JST	Japan
+9.5	ACT	Central Australia
+10	AET	Eastern Australia
+11	SST	Central Pacific
+12	NST	New Zealand
-11	MIT	Samoa
-10	HST	Hawaii
-9	AST	Alaska
-8	PST	Pacific Standard Time
-7	PNT	Arizona
-7	MST	Mountain Standard Time
-6	CST	Central Standard Time
-5	EST	Eastern Standard Time
-5	IET	Indiana East
-4	PRT	Atlantic Standard Time

Hours From Greenwich Mean Time (GMT)	Value	Description
-3.5	CNT	Newfoundland
-3	AGT	Eastern South America
-3	BET	Eastern South America
-1	CAT	Azores

Using the Localtime File

The local timezone is stored in the **/etc/localtime** and is used by GNU Library for C (glibc) if no value has been set for the TZ environment variable. This file is either a copy of the **/usr/share/zoneinfo/** file or a symbolic link to it. The ioThinx 4530 controller does not provide **/usr/share/zoneinfo/** files. You should find a suitable time zone information file and write over the original local time file in the ioThinx 4530 controller

Determining Available Drive Space

To determine the amount of available drive space, use the **df** command with the **–h** tag. The system will return the amount of drive space broken down by file system. Here is an example:

```
moxa@Moxa:~$ sudo df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/root       941M  812M   65M  93% /
devtmpfs        240M     0  240M   0% /dev
/dev/mmcblk1p3  6.0G  5.3M  5.7G   1% /overlayfs
overlay         6.0G  5.3M  5.7G   1% /var
overlay         6.0G  5.3M  5.7G   1% /etc
overlay         6.0G  5.3M  5.7G   1% /home
overlay         6.0G  5.3M  5.7G   1% /root
overlay         6.0G  5.3M  5.7G   1% /sbin
overlay         6.0G  5.3M  5.7G   1% /bin
overlay         6.0G  5.3M  5.7G   1% /usr
overlay         6.0G  5.3M  5.7G   1% /lib
overlay         6.0G  5.3M  5.7G   1% /tmp
overlay         6.0G  5.3M  5.7G   1% /mnt
overlay         6.0G  5.3M  5.7G   1% /opt
overlay         6.0G  5.3M  5.7G   1% /media
tmpfs           248M     0  248M   0% /dev/shm
tmpfs           248M  3.5M  245M   2% /run
tmpfs           5.0M     0  5.0M   0% /run/lock
tmpfs           248M     0  248M   0% /sys/fs/cgroup
tmpfs            50M     0   50M   0% /run/user/1000
```

Shutting Down the Device

To shut down the device, disconnect the power source to the computer. When the computer is powered off, main components such as the CPU, RAM, and storage devices are powered off, although an internal clock powered by a super capacitor may keep running.

You can use the Linux **shutdown** command to close all software running on the device and halt the system. However, main components such as the CPU, RAM, and storage devices will continue to be powered after you run this command.

```
moxa@Moxa:~$ sudo shutdown -h now
```

3. Firmware Update and System Recovery

Firmware Update and Set-to-Default Functions

Set-to-Default

1. Power off the device.
2. Press and hold the reset button; while holding the reset button:
   a. Power on the device; the RDY LED will blink green while the device is booting up.
   b. After the device has booted up, the RDY LED will blink red; continue holding the reset button until the RDY LED stops blinking.
3. Release the reset button to load the factory default settings.

For additional details on the LEDs, refer to the quick installation guide or the user's manual for your ioThinx 4530 controller.

NOTE

It should take about 20 seconds from the time the RDY LED starts blinking green until it stops blinking red.

ATTENTION

Reset-to-default will erase all the data stored on the boot storage

Back up your files before resetting the system to factory defaults. All the data stored in the ioThinx 4530 controller's boot storage will be destroyed after resetting to factory defaults.

You can also use the **`mx-set-def`** command to restore the ioThinx 4530 controller to factory defaults:

```
moxa@Moxa:~$ sudo mx-set-def
```

Firmware Update Using an SFTP Server or microSD Card

Updating the Firmware Under OS Mode

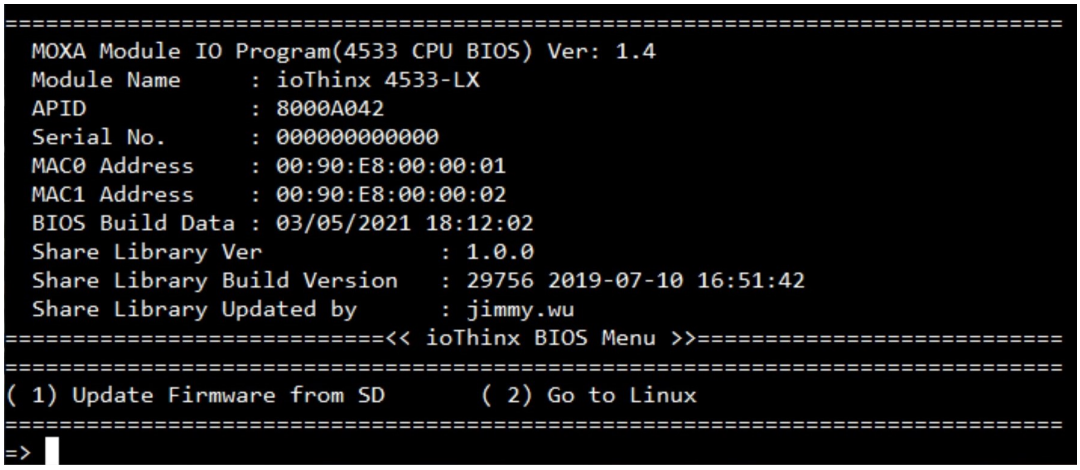
1. To update the firmware, log in to the product through the serial console. Instructions on how to connect to the serial console can be found in the ioThinx 4530 Hardware User's Manual.
2. Put the firmware (*.sh) file to the ioThinx 4530 device via an SFTP server or MicroSD card.
3. Use the following commands to update the firmware.

```
moxa@Moxa:~$ sudo ./FWR_ioThinx_4533_V1.1.0_Build_2019_0321_1305.sh
[sudo] password for moxa:
Upgrade firmware version from [1.0] to [1.1].
This step will destory all your firmware.
Continue ? (Y/N) :
```

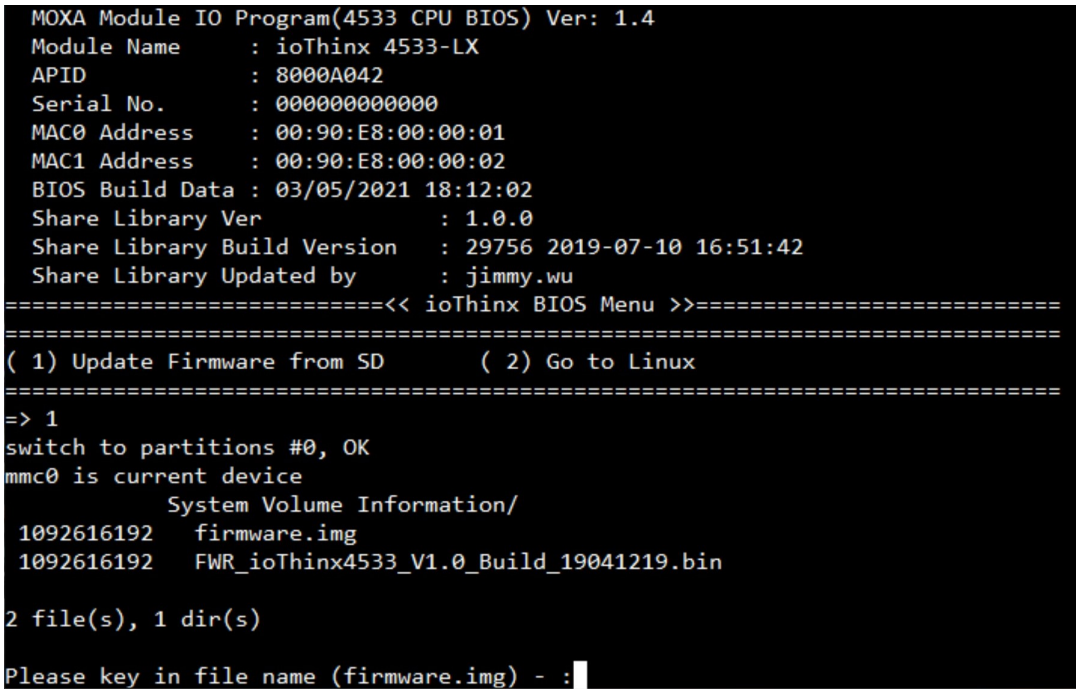
4. After the firmware update is complete, the ioThinx 4530 will restart automatically. Use the kversion command to check the firmware version.

Updating the Firmware Under BIOS Mode

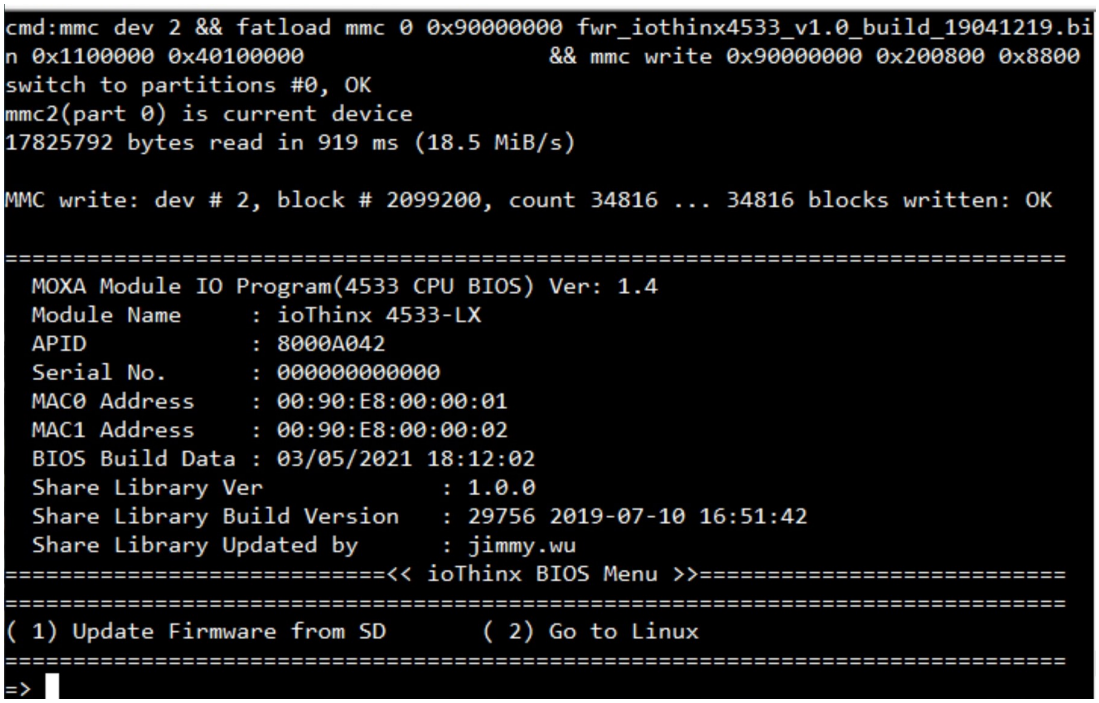
1. To update the firmware, log in through the serial console. Instructions on how to connect to the serial console can be found in the Hardware User's Manual for the ioThinx 4533.
2. After powering up the computer, press **Delete** to enter the bootloader configuration settings.



3. Enter **1** to update the firmware through the microSD card. Key in the file name of the firmware.



4. After updating the firmware, select **Go to Linux** to open the OS command-line console.



ATTENTION

The BIOS menu is not supported on the BIOS versions before v1.4.

If your OS has crashed, and you are unable to enter the BIOS menu when pressing **Delete** after powering up, then please send your device to RMA.

4. Programming Guide

Click the following link to download the ioThinx 4530 Programming Guide:

https://www.moxa.com/en/products/industrial-edge-connectivity/controllers-and-ios/advanced-controllers-and-i-os/iothinx-4530-series#resources

The ioThinx 4530 Programming Guide includes the following sections:

Tutorials:

Shows users how to build code, use the cloud SDK, and use Python to access I/O data.

I/O Libraries:

Shows users how to access ioThinx 45M modules.

Module Information:

Shows users how to access module information.

Rotary Switch:

Shows users how to read the status of rotary switches.

User Defined LED Indicator:

Shows users how to access LED indicators.

Error Codes:

Provides the meaning of the return code to help users perform troubleshooting tasks.

A. Cycle Time Calculation

The controller's cycle time is defined as how much time the CPU needs to poll the status of all IO modules. This information is important since it allows users to make sure the controller can control their application within a designated time period. The cycle time calculation is based on the following table. A cycle time is calculated for each group of eight appended 45M modules. The cycle time of a group is the sum of the cycle time of the first module in the group (the times in column 1) plus the cycle times of the 2nd through 8th modules (the times in column 2) in the group. To calculate the cycle time of ioThinx 4530 Series CPU, simply add up the cycle times of all of the groups connected to the ioThinx, and then round the time up to the nearest millisecond.

	Cycle time as 1st module in one group (μs)	Cycle time as 2nd to 8th module of the one group (μs)
45MR-1600	1200	100
45MR-1601	1200	100
45MR-2404	1300	100
45MR-2600	1200	100
45MR-2601	1200	100
45MR-2606	1200	100
45MR-3800	1300	200
45MR-3810	1300	200
45MR-6600	1500	300
45MR-6810	1500	300

We provide two examples to illustrate cycle time calculations.

Case 1. 4-piece 45MR-1600 and 4-piece 45MR-2601.

1st module: 45MR-1600	2nd module: 45MR-1600	3rd module: 45MR-1600	4th module: 45MR-1600	5th module: 45MR-2601	6th module: 45MR-2601	7th module: 45MR-2601	8th module: 45MR-2601

In this case, the eight modules form one group. The cycle time of this combination is 1900 μs = 1200 μs + 7 x 100 μs. The ioThinx 4530 Series will round up the cycle time to the ms level, and consequently the cycle time of this combination is 2 ms.

Case 2. 4 x 45MR-1600, 4 x 45MR-2601, 2 x 45MR-3800.

1st module: 45MR-1600	2nd module: 45MR-1600	3rd module: 45MR-1600	4th module: 45MR-1600	5th module: 45MR-2601	6th module: 45MR-2601	7th module: 45MR-2601	8th module: 45MR-2601	9th module: 45MR-3800	10th module: 45MR-3800

In this case, the 10 modules are separated in two groups. The first group is outlined in red above, whereas the second group is outlined in orange. The combination of modules in the first group is the same as in **Case 1**, which was shown to have a cycle time = 1900 µs. For the second group, the cycle time is 1500 µs = 1300 µs + 200 µs. Therefore, the total cycle time of the two groups is 3400 µs = 1900 µs + 1500 µs, which when rounded up to the nearest ms results in a total cycle time = 4 ms.