DRP/BXP/RKP Series Computers Linux Installation Manual

Version 1.0, October 2023

www.moxa.com/products

DRP/BXP/RKP Series Computers Linux Installation 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

The MOXA logo is a registered trademark of Moxa Inc.
All other trademarks or registered marks in this manual belong to their respective manufacturers.

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. Moxa x86 Linux SDK

Introduction

The Moxa x86 Linux SDK enables easy deployment of Linux on the RKP/BXP/DRP series x-86. The SDK includes peripheral drivers, peripheral control tools, and configuration files. The SDK also provides deployment functions such as build & installation log, dry-run, and self-test on target models.

Supported Series and Linux Distributions

Series Name	Debian 11	Ubuntu 22.04	RHEL 9
RKP-A110	✓	✓	✓
RKP-C110	✓	✓	✓
BXP-A100	✓	✓	✓
BXP-C100	✓	✓	✓
DRP-A100	✓	✓	✓
DRP-C100	✓	✓	✓

Prerequisites

- A system running Linux (Debian, Ubuntu, RedHat)
- Access to the terminal/command line
- A user account with **sudo/root** privileges
- The network settings configured before installation

The x86 Linux Installation Wizard

The x86 Linux SDK zip file consists of the following:

```
<wizard>.tgz - the tarball file of x86 Linux SDK Install Wizard
README.docx  - this file in docx format
sources_list - the list of source of x86 Linux SDK Install Wizard
build_info   - x86 Linux SDK build information
Support Models and Linux Distributions
```

Extract the files from the zip file. The installation wizard files are packaged in a tarball (*tgz) file.

Extracting the Installation Wizard Files

✏️ **NOTE**

The installation file should be extracted to a system running a Linux OS (Debian, Ubuntu, or RedHat) environment.

Extract the wizard installation files from the tarball (*tgz) file.

```
tar xvf Moxa_x86_Linux_Install_Wizard_<ver>_Build_<build_date>.tgz
cd Moxa_x86_Linux_Install_Wizard_<ver>_Build_<build_date>
```

The x86 Linux Installation Wizard consists of the following directories and files:

```
product.d/ - congfiguration files for products
scripts/   - install wizard generic programs
src/       - source code for the drivers and tools
install.sh - entry program for build and install the SDK
README.md  - this file: introduction and build instructions
CHANGELOG  - change log and difference to the SDK release
LICENSE    - MOXA license statement file
version    - current version of x86 Linux Install Wizard
```

Installing the Linux Drivers

By default, the installation wizard installs the latest version. If you want to reinstall the current version or install an older version, run **install.sh** with the **--force** option .

Command

```
./install.sh
```

✏️ **NOTE**

This command requires **sudo** or **root** privileges.

Result (E.g., RKP-A110)

```
[info] Product Name: RKPA110
[info] OS Name: Ubuntu
[info] OS Version: 22.04
[info] Kernel Info: Linux moxa 5.19.0-32-generic #33~22.04.1-Ubuntu SMP
PREEMPT_DYNAMIC Mon Jan 30 17:03:34 UTC 2 x86_64 x86_64 x86_64 GNU/Linux
Do you want to continue? [Y/n]y
[info] >>> Execute hook script "install-dev-tools.sh".
[info] <<< Execute hook script "install-dev-tools.sh" done.
[info] >>> Execute hook script "build-and-install-source.sh".
[info] === Run pre-install
[info] === Install driver
Do you want to install moxa-it87-gpio-driver (5.2+1.5.0-1)? [Y/n]y
[info] Installing moxa-it87-gpio-driver (5.2+1.5.0-1)
Do you want to install moxa-it87-wdt-driver (5.2+1.5.0-1)? [Y/n]y
[info] Installing moxa-it87-wdt-driver (5.2+1.5.0-1)
...
... (skip)
...
[info] Done. Please reboot machine for installation to take effect.
Do you want to reboot now? [Y/n]
```

A prompt asking if you want to reboot the system is displayed.

```
Do you want to reboot now? [Y/n]
```

Enter **y**, **Y**, or **yes** to reboot the system, or **n**, **N**, or **no** to exit the installation process.

Checking the Installation Status

To check the installation status of the driver, run **install.sh** with the **--selftest** option.

Command

```
./install.sh --selftest
```

Result (E.g., RKP-A110)

- **Name**: Driver or tool name
- **Installed**: Installation status of the driver or tool
  Yes: The driver/tool is installed
  No: The driver/tool is not installed
- **Status**: Shows the readiness of the installed driver or tool
  Loaded: The driver is loaded
  Active: The tool or service is active
- **Version**: The version of the driver or tool

```
[info] Product Name: RKPA110
[info] OS Name: Ubuntu
[info] OS Version: 22.04
[info] Kernel Info: Linux moxa 5.19.0-32-generic #33~22.04.1-Ubuntu SMP
PREEMPT_DYNAMIC Mon Jan 30 17:03:34 UTC 2 x86_64 x86_64 x86_64 GNU/Linux
[info] >>> Execute hook script "self-test.sh".
[info] ------------------------------------------------------------------------
---
[info]  Name                            Installed       Status          Version
[info]
========================================================================
[info]  moxa-it87-gpio-driver                                           5.2+1.5.0-1
[info]  - gpio_it87                     Yes             Loaded
[info]  moxa-it87-wdt-driver                                            5.2+1.5.0-1
[info]  - it87_wdt                      Yes             Loaded
[info]  - watchdog service              Yes             Active
[info]  moxa-it87-serial-driver                                         1.4.1+u2
[info]  - it87_serial                   Yes             Loaded
[info]  moxa-mxuport-driver
5.1.1_build_23080316
[info]  - mxuport                       Yes             Loaded
[info]  moxa-x86-control-tools                                          1.8.1
[info]  - mx-uart-ctl                   Yes             6 ports
[info]  - mx-dio-ctl                    Yes             8 DI / 8 DO
[info] ------------------------------------------------------------------------
---
[info] <<< Execute hook script "self-test.sh" done.
```

Displaying the Help Page

Run the **install.sh --help** command to show the help page that contains a usage summary of all the command options.

Command

```
./install.sh --help
```

Result

```
             __  __  _____  __ __ __ _
            |  \/  |/ _ \ \ / /| \ | |
            | |\/| | | | \ V / |  \| |
            | |  | | |_| || |  | |\  |
            |_|  |_|\___/ |_|  |_| \_|

       ----------------------------------
                 X86 INSTALL WIZARD

Usage: install.sh [option]

Options:
        -h, --help       Display this help page
        -y, --yes        Automatically answer yes
        -v, --version    Display the version information
        -s, --selftest   Run the self test cases
            --dry-run    List available driver(s) and tool(s) only
            --uninstall  Uninstall driver(s) and tool(s)
            --force      Install driver and tool even if the
                         version is the same or older (default is
                         to install newer version)

Without passing any option, it would run the installation directly.
```

Displaying the Driver Version

Command

```
./install.sh --version
```

Expected result

```
1.0.0
```

Using the --yes Option

The **--yes** (alternatively, **-y**) option automatically returns yes at all prompts. It can be used with other options as indicated below.

For example, assume "yes" as answer to all prompts during installation process.

```
./install.sh --yes
```

Assume "yes" as answer to all prompts during uninstallation process.

```
./install.sh --yes --uninstall
```

Using the --dry-run Option

The **--dry-run** option simulates the installation process to show what would be installed without installing anything or making any changes to the system.

Command

```
./install.sh --dry-run
```

Result (E.g., RKP-A110)

- **Name**: Driver or tool name
- **Version**: The version of the driver or tool
- **Tag**: The tag name of the driver or tool in the Git repository

```
-----------------------------------------------------------------
 Product Name: RKPA110
-----------------------------------------------------------------
 Name                        Version                  Tag
=================================================================
 moxa-it87-gpio-driver       5.2+1.5.0-1              master
 moxa-it87-wdt-driver        5.2+1.5.0-1              5.2-master
 moxa-it87-serial-driver     1.4.1+u2                 master
 moxa-mxuport-driver         5.1.1_build_23080316     5.x-
5.1.1_build_23080316-develop
 moxa-x86-control-tools      1.8.1                    master
-----------------------------------------------------------------
```

Uninstalling the Linux Drivers

Use the `install.sh --uninstall` command to unstill the drivers and tools.

Command

```
./install.sh --uninstall
```

✏️ **NOTE**

This command requires `sudo` or `root` privileges.

Expected result (RKP-A110)

```
[info] Product Name: RKPA110
[info] OS Name: Ubuntu
[info] OS Version: 22.04
[info] Kernel Info: Linux moxa 5.19.0-32-generic #33~22.04.1-Ubuntu SMP
PREEMPT_DYNAMIC Mon Jan 30 17:03:34 UTC 2 x86_64 x86_64 x86_64 GNU/Linux
Do you want to continue? [Y/n]y
[info] >>> Execute hook script "uninstall.sh".
[info] === Uninstall driver
Do you want to uninstall moxa-it87-gpio-driver (5.2+1.5.0-1)? [Y/n]y
[info] Uninstall moxa-it87-gpio-driver (5.2+1.5.0-1)
Do you want to uninstall moxa-it87-wdt-driver (5.2+1.5.0-1)? [Y/n]y
[info] Uninstall moxa-it87-wdt-driver (5.2+1.5.0-1)
[info]  Remove Watchdog Service
Do you want to uninstall moxa-it87-serial-driver (1.4.1+u2)? [Y/n]y
[info] Uninstall moxa-it87-serial-driver (1.4.1+u2)
...
... (skip)
...
[info] <<< Execute hook script "uninstall.sh" done.
[info] Done. Please reboot machine for uninstallation to take effect.
Do you want to reboot now? [Y/n]
```

And finally, you'll see a prompt asking if you want to reboot the system.

```
Do you want to reboot now? [Y/n]
```

Enter y, Y, or yes to reboot the system, or n, N, or no to exit out of uninstall process.

Checking the Log file

The installation log file `install.log` contains information on all events that have taken place during the installation process. The file is in the same as the driver. Run the following command to access the log file.

Command

```
cat install.log
```

Moxa x86 Peripherals Control Tools

The Moxa x86 Linux SDK includes tools to manage the serial and the digital I/O ports of the supported devices.

mx-uart-ctl

The serial port management tool **mx-uart-ctl** retrieves information on the serial ports of the computer and sets the operating mode (RS-232/422/RS-485 2-wire/ RS-485 4-wire) for each port.

Supported Series

- BXP-A100
- BXP-C100
- RKP-A110
- RKP-C110
- DRP-A100
- DRP-C100

Usage

```
Usage:
            mx-uart-ctl -p <port_number> [-m <uart_mode>]

OPTIONS:

            -p <port_number>
                        Set target port.
            -m <uart_mode>
                        Set target port to uart_mode
                        0 --> set to RS-232 mode
                        1 --> set to RS-485-2W mode
                        2 --> set to RS-422 mode
                        3 --> set to RS-485-4W mode

Example:
            Get mode from port 0
            # mx-uart-ctl -p 0

            Set port 1 to mode RS232
            # mx-uart-ctl -p 1 -m 0
```

mx-dio-ctl

The D I/O port management tool mx-dio-ctl is used to retrieving information on the DI and DO ports and for setting the DO port status (low/high).

Supported Series

- BXP-A100
- BXP-C100
- RKP-A110
- RKP-C110

Usage of mx-dio-ctl

```
Usage:
        mx-dio-ctl <-i|-o <#port number> [-s <#state>]>

OPTIONS:
        -i <#DIN port number>
        -o <#DOUT port number>
        -s <#state>
                Set state for target DOUT port
                0 --> LOW
                1 --> HIGH

Example:
        Get value from DIN port 0
        # mx-dio-ctl -i 0
        Get value from DOUT port 0
        # mx-dio-ctl -o 0

        Set DOUT port 0 value to LOW
        # mx-dio-ctl -o 0 -s 0
        Set DOUT port 0 value to HIGH
        # mx-dio-ctl -o 0 -s 1
```