

EC2x Series QuecOpen Auto-startup Service Guide

LTE Standard Module Series

Rev. EC2x_Series_QuecOpen_Auto-startup_Service_Guide_V1.0

Date: 2020-04-12

Status: Released

Our aim is to provide customers with timely and comprehensive service. For any assistance, please contact our company headquarters:

Quectel Wireless Solutions Co., Ltd.

Building 5, Shanghai Business Park Phase III (Area B), No.1016 Tianlin Road, Minhang District, Shanghai 200233, China

Email: info@quectel.com

Or our local office. For more information, please visit:

<http://quectel.com/support/sales.htm>

For technical support, or to report documentation errors, please visit:

<http://quectel.com/support/technical.htm>

Or email to: support@quectel.com

GENERAL NOTES

QUECTEL OFFERS THE INFORMATION AS A SERVICE TO ITS CUSTOMERS. THE INFORMATION PROVIDED IS BASED UPON CUSTOMERS' REQUIREMENTS. QUECTEL MAKES EVERY EFFORT TO ENSURE THE QUALITY OF THE INFORMATION IT MAKES AVAILABLE. QUECTEL DOES NOT MAKE ANY WARRANTY AS TO THE INFORMATION CONTAINED HEREIN, AND DOES NOT ACCEPT ANY LIABILITY FOR ANY INJURY, LOSS OR DAMAGE OF ANY KIND INCURRED BY USE OF OR RELIANCE UPON THE INFORMATION. ALL INFORMATION SUPPLIED HEREIN IS SUBJECT TO CHANGE WITHOUT PRIOR NOTICE.

COPYRIGHT

THE INFORMATION CONTAINED HERE IS PROPRIETARY TECHNICAL INFORMATION OF QUECTEL WIRELESS SOLUTIONS CO., LTD. TRANSMITTING, REPRODUCTION, DISSEMINATION AND EDITING OF THIS DOCUMENT AS WELL AS UTILIZATION OF THE CONTENT WITHOUT PERMISSION ARE FORBIDDEN. OFFENDERS WILL BE HELD LIABLE FOR PAYMENT OF DAMAGES. ALL RIGHTS ARE RESERVED IN THE EVENT OF A PATENT GRANT OR REGISTRATION OF A UTILITY MODEL OR DESIGN.

Copyright © Quectel Wireless Solutions Co., Ltd. 2020. All rights reserved.

About the Document

Revision History

Version	Date	Author	Description
1.0	2020-04-12	Francis HUAN	Initial

Contents

About the Document	2
Contents	3
Table Index	4
Figure Index	5
1 Startup Sequence for Linux Applications	6
1.1. Applicable Modules	6
2 Add Linux Auto-startup Programs	7
3 Methods of Adding with Linux Console	8
3.1. Method One	8
3.2. Method Two	9
3.3. Method Three	9
4 Method of Adding with SDK Packaging	10
5 Appendix A References	11

Table Index

Table 1: Applicable Modules.....	6
Table 2: Terms and Abbreviations	11

Figure Index

Figure 1: Add Startup Program under <i>/etc/init.d/</i>	8
Figure 2: Create a Soft Link	8
Figure 3: Add Startup Program under <i>/etc/init.d/</i>	9
Figure 4: Add Startup Program with SDK	10

1 Startup Sequence for Linux Applications

After the kernel starts the init process, the init process will parse the */etc/inittab* file. The init program used is *sysvinit* and the init startup process is performed in the sequence of *init(sysvinit)→/etc/init.d/rcS→/etc/init.d/rc*. After the rc script starts up, the startup scripts under the */etc/rcS.d/* directory will be executed in sequence.

In principle, startup programs can be added to any script of *init*, *rcS* and *rc*. But as Linux provides a set of methods for adding startup programs, it is recommended to take these methods which will be introduced in the following chapters.

1.1. Applicable Modules

Table 1: Applicable Modules

Module Series	Module
EC2x series QuecOpen	EC25 series QuecOpen
	EC21 series QuecOpen
	EC20 R2.1 QuecOpen

2 Add Linux Auto-startup Programs

Create a soft link under the `/etc/rc[0-6].d/` directory which points to the control script under the `/etc/init.d/` directory. The naming rule of files under directory `/etc/rc[0-6].d/` is S/K + nn + script, for example S45test.sh in **Figure 2**.

- 0–6 indicates the running level of Linux operating system. Check the current running level by executing `runlevel`. rc5 is used by default.
- S/K: S indicates the script to be executed after power-on and K indicates the script to be executed after power-off.
- nn, whose range is 0–100, indicates priority. The bigger the number is, the lower the priority will be.
- script: the file name of the script which the soft link points to.

3 Methods of Adding with Linux Console

3.1. Method One

If `/etc` is not a tmpfs and is readable & writable, create a soft link with command `ln` according to the naming rule in **Chapter 2**.

1. Under `/etc/init.d/`, add the program to be executed after power-on. The following figure shows an example with `test.sh`.

```
root@mdm9607-perf:/etc/init.d# ls -l test.sh
-rwxrwxrwx 1 root root 18 Jan 6 17:58 test.sh
```

Figure 1: Add Startup Program under `/etc/init.d/`

2. Create a soft link under `rc5` directly with command `ln`. As shown in the following figure, `test.sh` is the startup script to be executed after power-on, and its priority level is 45.

```
root@mdm9607-perf:/etc/init.d# ln -sv test.sh ../rc5.d/S45test.sh
```

Figure 2: Create a Soft Link

3.2. Method Two

If `/etc` is not a tmpfs and is readable & writable, create a soft link with command `update-rc.d` according to the naming rule in **Chapter 2**.

1. Under `/etc/init.d/`, add the program to be executed after power-on. The following figure shows an example with `test.sh`.

```
root@mdm9607-perf:/etc/init.d# ls -l test.sh
-rwxrwxrwx 1 root root 18 Jan 6 17:58 test.sh
```

Figure 3: Add Startup Program under `/etc/init.d/`

2. Create a soft link directly with command `update-rc.d`:

```
update-rc.d test.sh defaults 45
```

3.3. Method Three

If `/etc` is a tmpfs, add the program to be executed after power-on to `/data/boot/`, and name it according to the naming rule (S|K + nn + script) in **Chapter 2**. Then this program will be copied by other scripts from `/data/boot/` to `/etc/`, which will avoid loss of files under `/etc/` after rebooting (SDK upgrade is required for this method).

4 Method of Adding with SDK Packaging

As the system uses rc5 to boot by default, users can follow the steps below to add startup programs with SDK packaging.

1. Copy the program to be executed after power-on to the directory `/ql-ol-rootfs/etc/init.d/`.
2. Execute the following command:

```
cd ql-ol-rootfs/etc/rc5.d
```

3. Create a soft link:

```
ln -vsf ../init.d/test.sh S45test.sh
```

4. Build rootfs.

The following figure shows an example of adding startup program test.sh with SDK and its priority level is 45.

```
will@server2:/home/share/francis/wps/ql-ol-sdk$ ls ql-ol-rootfs/etc/init.d/test.sh
ql-ol-rootfs/etc/init.d/test.sh
will@server2:/home/share/francis/wps/ql-ol-sdk$ cd ql-ol-rootfs/etc/rc5.d/
will@server2:/home/share/francis/wps/ql-ol-sdk/ql-ol-rootfs/etc/rc5.d$ ln -vsf ../init.d/test.sh S45test.sh
'S45test.sh' -> '../init.d/test.sh'
```

Figure 4: Add Startup Program with SDK

5 Appendix A References

Table 2: Terms and Abbreviations

Abbreviation	Description
LTE	Long Term Evolution
RC	Run Control
RCS	Run Commands
SDK	Software Development Kit
TMPFS	Temporary File System