IAC-IMX8MM-KIT Linux User Manuel

Ver：1.0
Apr,2020

QIYANG TECHNOLOGY CO., LTD

Catalogue

Version Update

Version	Hardware Platform	Description	Date	Author
1.0	IAC-IMX8MM-CM	First Version, Version Published	2020-04-15	Zhujh

Preface

It is honored that IAC-IMX8MM-CM is selected for using. As for Linux part, there is four manuals in total, including, *IAC-IMX8-CM User Manual.PDF, IAC-IMX8-CM Hardware Manual.PDF, IAC-IMX8-CM Linux Module Specification And Test Manual.PDF.& IAC-IMX8-CM System Image Flashing Manual.*

About the mainboard testing, you can refer to the *IAC-IMX8-CM Linux Functional Description And Testing Manual.PDF.*

Image flashing ,please refer to *IAC-IMX8-CM System Image Flashing Manual.PDF.*

Please read *IAC-IMX8-CM Hardware Manual* carefully before testing the mainboard !

Company Profile

Zhejiang Qiyang Intelligent Technology Co., Ltd., established in 2007, which locates in Hangzhou, Zhejiang, PRC. It is a high-end technological enterprise that specializes in exploitation, fabrication, and selling embedded computer mainboards .With 10 years of experiences, Qiyang has established the completed service chain from the design concept to mass production successfully. Product Line includes:Cirrus Logic EP93XX, ARM9 Mainboard, ATMEL AT91SAM926x Mainboard, FreeScale I.MX Mainboard, TI Davinci audio/video Mainboard, etc. They run Linux2.4/2.6,WinCE5.0/6.0 OS, and Qiyang also provide the embedded system. Application area includes : Industrial Control, Data Acquisition, Info-communication, Medical Device, Video Surveillance, In-car Entertainment, etc.Customer demand is the driving force of the development of Qiyang , Qiyang will continue to improve themselves.

Tel：0571-87858811，87858822
Fax：0571-87858822
Technical Support E-MAIL：support@qiyangtech.com
Website： http://www.qiyangtech.comAddress： 3rd floor, Building A, WSCG Building, NO.6 Xiyuan 8th Road, Sandun Town, Xihu District, Hangzhou City, Zhejiang ,PRC .310030
Postal Code：310012

Ⅰ.Preparation

◆ A set of PC installed Windows 7 or Windows10;

◆ A set of PC installed virtual machine, the virtual machine is Linux OS (Ubuntu or other Linux distribution OS), if the Virtual Machine doesn't install Linux OS, please install the Ubuntu14.04 by refer to Virtual Machine Ubuntu Installation Manual, this manual introduces how to install Ubuntu 14.04 in Windows 10 OS.

◆ Serial Connection: Connect J18 on development board with serial port on PC via serial cable.

◆ Network Connection: Connect J11 with LAN port on PC via LAN cable.

◆ USB Connection: Connect USB Device (J13) on development board with USB on PC via USB cable.

◆ UART setting: Open terminal communication software or putty.exe, select the parameters of the serial port: Baud Rate:(115200), Data Bit:(8-bit), Stop Bit:(1-bit), Parity Bit:(None), Data Flow Control:(None)

Ⅱ.Flashing Linux System Image File

IAC-IMX8MM-CM has the exclusive flashing tool[uuu.exe], please select the most suitable boot mode by referring to the below manual:

IAC-IMX8-MB System Image Flashing Manual

Ⅲ.Functional Description & Testing

The board's file system has integrated test programs, please find out the relative test program in directory [/usr/test].

IAC-IMX8MM-CM Linux Functional Description And Testing Manual.PDF.

Ⅳ.To build embedded Linux development environments

4.1. Install cross-compiler

Install Linux kernel and the cross compiler for other application programs required in PC Ubuntu 14.04(64-bit). The finished cross-compiler's tool chain for the development board are already in

folder.

[$ ./fsl-imx-wayland-glibc-x86_64-fsl-image-qt5-validation-imx-aarch64-toolchain-4.14-sumo.sh]

Input installation path, then press [Enter], adopts default path.

Input [Y], confirm

```
./fsl-imx-wayland-glibc-x86_64-fsl-image-qt5-validation-imx-aarch64-toolchain-4.14-sumo.sh
NXP i.MX Release Distro SDK installer version 4.14-sumo
==================================================
Enter target directory for SDK (default: /opt/fsl-imx-wayland/4.14-sumo):
You are about to install the SDK to "/opt/fsl-imx-wayland/4.14-sumo". Proceed[Y/n]? Y
[sudo] password for zhujh:
Extracting SDK.....
.....
.....
.....done
Setting it up...done
SDK has been successfully set up and is ready to be used.
Each time you wish to use the SDK in a new shell session, you need to source the environment setup scr
ipt e.g.
 $ . /opt/fsl-imx-wayland/4.14-sumo/environment-setup-aarch64-poky-linux
```

4.2 NFS Network File System

NFS(Network File System), it makes the different hosts to visit the files which shared from the remote NFS server via network. Thus, it makes the hosts as the NFS server, we can visit the files from the hosts on the development board via network. In embedded OS, the NFS makes the developing on application programs more convenient, the specific methods as below:

1. Install

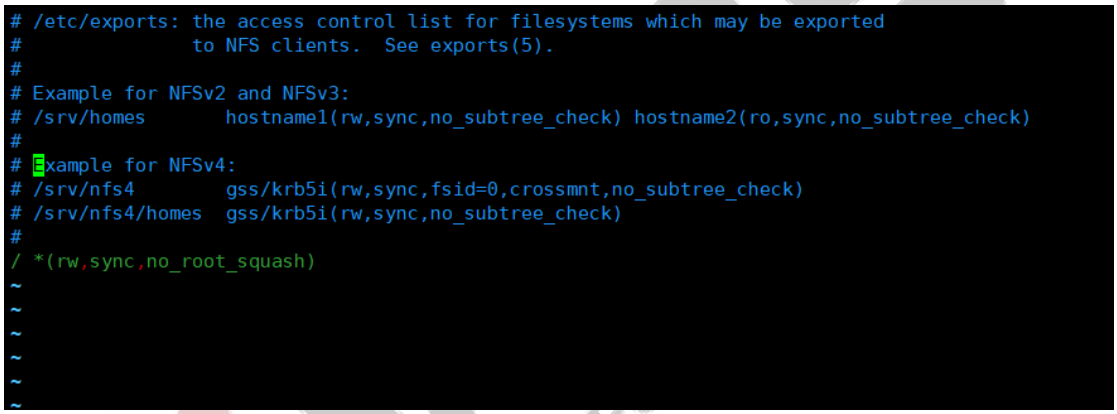
[$sudo apt-get install nfs-kernel-server]

2. Create the shared directory, modify at the file's ending [/etc/exports]

   [$sudo vi /etc/exports]

   Add the below command at the file's ending:

   [/ *(rw,sync,no_root_squash)]



   Thus, it makes the host's root directory as the shared directory.

3. Boot NFS server, reboot [nfs-kernel-server]:

   [$ sudo /etc/init.d/nfs-kernel-server restart]

4. Locally verify NFS

   [$ sudo mount local:/ /mnt]

   [$ ls /mnt/]

You can see the content in root directory in [/mnt] directory, it means

the NFS configuration is correct.

Ⅴ.Test Code Compilation

In [Test Code/Source Code/CD-ROM] directory, it provides source code for the relative interfaces, it could be compiled as required. All operations should be executed after the system installed cross-compiler and be added into the environment variables. If without installing cross-compiler, then please refer to Chapter 4.1 to install cross compiler.

Select the relative source code which needs compilation, here, take a sample of Buzzer.

Copy[Buzzer_test/Test Code/Source Code/CD-ROM] to directory [Ubuntu 14.04], open Linux terminal, enter into [buzzer_test] directory, execute[make]command, you will get the executable file [buzzer_test] in this directory after compiling.

[$ cd buzzer_test/]

[$ ls]

```
buzzer_test.c  Makefile
```

[$ source /opt/fsl-imx-wayland/4.14-sumo/environment-setup-aarch64-poky-linux]

[$ make]

Ⅵ. U-Boot Compilation

CD-ROM provide the compiled boot image and source code. Suggest you to use the boot image in CD-ROM as the bootstrap program. If needs to recompile the bootstrap program, please follow steps as shown. Before executing the commands, please confirm cross-compiler is installed correctly and add it into environment variables. If not, please refer to Chapter 4.1 of this manual to install.

Source Code：

[imx-atf_imx_4.14.98_2.0.0_ga.tar.gz]

[imx-mkimage_imx_4.14.98_2.0.0_ga.tar.g]

[uboot-imx_v2018.03_4.14.98_2.0.0_ga.tar.gz]

Unzip u-boot source code zipped file in CD-ROM:

[$tar zxvf uboot-imx_v2018.03_4.14.98_2.0.0_ga.tar.gz]

[$tar zxvf imx-atf_imx_4.14.98_2.0.0_ga.tar.gz]

[$tar zxvf imx-mkimage_imx_4.14.98_2.0.0_ga.tar.gz]

6.1. Compile imx_v2018.03_4.14.98_2.0.0_ga

[$ cd imx_v2018.03_4.14.98_2.0.0_ga/]

[$source/opt/fsl-imx-wayland/4.14-sumo/environment-setup-aarch64-poky-linux]

[$ make imx8mm_qiyang_defconfig]

[$ make -j40]

Generate the files after compiling:

[u-boot-nodtb.bin]

[spl/u-boot-spl.bin]

[arch/arm/dts/fsl-imx8mm-qiyang.dtb]

6.2. Compile imx-atf

[$ cd imx-atf/]

[$ source /opt/fsl-imx-wayland/4.14-sumo/environment-setup-aarch64-poky-linux]

[$ LDFLAGS="" make PLAT=imx8mm]

Generate the files after compiling:

[build/imx8mm/release/bl31.bin]

6.3 Compile imx-mkimage

Compile [*imx-mkimage]* after [*imx-atf]* and[*imx_v2018.03_4.14.98_2.0.0_ga]* is compiled.

[$ cd imx-mkimage/

[$ source /opt/fsl-imx-wayland/4.14-sumo/environment-setup-aarch64-poky-linux]

[$ cp ../imx_v2018.03_4.14.98_2.0.0_ga/u-boot-nodtb.bin iMX8M/]

[$ cp ../imx_v2018.03_4.14.98_2.0.0_ga/spl/u-boot-spl.bin iMX8M/]

[$ cp ../imx_v2018.03_4.14.98_2.0.0_ga/arch/arm/dts/fsl-imx8mm-qiyang.dtb iMX8M/]

[$ cp ../imx_v2018.03_4.14.98_2.0.0_ga/tools/mkimage ./mkimage_uboot]

[$ cp ../imx-atf/build/imx8mm/release/bl31.bin iMX8M/]

[$ make SOC=iMX8MM flash_evk]

Generate the files after compiling:

[iMX8M/flash.bin]

[flash.bin] is the uboot image file which would be burn into development kit.

Ⅶ.Kernel Compilation

CD-ROM has the configured kernel source code file:[*kernel_imx_4.14.98_2.0.0_ga.tar.gz]*

Copy the zipped file to directory Ubuntu 14.04. User can configure the kernel freely. Before compiling kernel, please make sure the cross-compiler have installed into the system correctly.

[$ tar zxvf kernel_imx_4.14.98_2.0.0_ga.tar.gz]

[$ cd imx_4.14.98_2.0.0_ga/]

[$ source /opt/fsl-imx-wayland/4.14-sumo/environment-setup-aarch64-poky-linux]

[$ make defconfig]

[$ LDFLAGS="" CC="$CC" make -j40]

Generate the files after compiling:

[arch/arm64/boot/Image]

[arch/arm64/boot/dts/freescale/fsl-imx8mm-qiyang.dtb]

Ⅷ.Application Development

8.1 Hello World

You can develop the application program on PC freely. Take *Hello World* as an example. Firstly, write a program as shown:

```
#include <stdio.h>

int main(void)
```

```
{
    printf("Hello World ! \n");
    return 0;
}
```

Save in the [*hello.c*] file, we use command as shown to cross-compile it.

8. 2. Cross-Compile

To make the program run smoothly on the development kit, the application should be compiled by the installed cross-compiler. Put files mentioned above to the directory of Ubuntu 14.04, we can compile with the below commands:

```
[$ source /opt/fsl-imx-wayland/4.14-sumo/environment-setup-aarch64-poky-linux]
[$ $CC -o hello hello.c]
```

After compiling, there is a binary file [*Hello*] in the current directory which can be executed in ARM platform.

8.3. Run Application Program

As what we described above, application can be operated on the dashboard. Next let us introduce how to run an application on mainboard.

Here are two methods executed: Mount NFS server and Add Into System files.

We can use the NFS server which is already done. After mounting the NFS server of the host, the files of host can be operated on the development kit like copying files, run the programme and so on. It is much convenient to test and develop. The detailed steps shown as follow:

1.Make sure the development kit is connected with PC with cable, and open the NFS server on PC.

2. Set development kit IP and PC IP at the same network segment. For example:

PC IP ： 192.168.1.75

Target board IP： 192.168.1.203

Network Marsk: 255.255.255.0

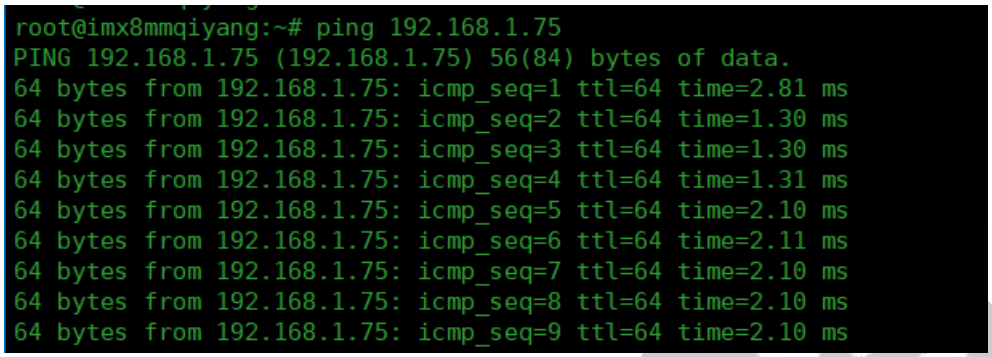
Broadcast IP: 192.168.1.255

3.Test the network connect

Ping the host on the development kit, text command on the Hyper

Terminal as shown:

ping 192.168.1.75



Ping development kit on the host as the same way. If the host and development kit can ping each other smoothly, the network is connected.

4.Mount NFS server of the host

Text the command in the Hyper Terminal as shown:

mount -o nolock 192.168.1.75:/ /mnt/

cd /mnt/

After mounting correctly, there is a root directory in directory [/mnt] in the development kit. By this way, it is convenient for testing and developing. Until the program works it out, it can be added into files system and flashed into eMMc which can avoid flashing the program into eMMC constantly.

8.4. Add Application Program Into File System

After testing the application, it can be added to file systems directly. So that you can flash it and flash the file system into eMMc. Next, it comes to introduce how to add application to root file system. Copy the root file system source code to Ubuntu 14.04 from CD-ROM and unzip it,

Firstly, create a new folder *filesystem* , the unzipped files are also in there.

[$ mkdir filesystem]

Copy the zipped file system [*rootfs.tar.bz2*] in CD-ROM to the directory of the filesystem.

[$ tar jxvf rootfs.tar.bz2] Operate at the unzipped filesystem

[$ ls]

You can add your application into the directory as shown.

Before zipping the files in this folder, please delete the original files named [*rootfs.tar.bz2*]

[$ rm -rf rootfs.tar.bz2]

[$ tar -jcvf rootfs.tar.bz2 -R *]

After zipping, there is a new file *[rootfs.tar.bz2]* generated in the current folder.

Ⅸ.Conclusion

Information above might be not detailed enough. If there is any technical problem or suggestions，please contact us：supports@qiyangtech.com，also can login our company's forum：http://www.qiytech.com/bbs/，More information，please contact our sales: sales@qiyangtech.com，or login: http://www.qiytech.com/index.html。

Zhejiang Qiyang Intelligent Technology Co., Ltd

Tel: 86-571-87858811 / 87858822

Fax: 86-571-89935912

Technology Support：86-571-89935913

E-MAIL: supports@qiyangtech.com

Website: http://www.qiytech.com

Address:3rd floor, Building A, WSCG Building, NO.6 Xiyuan 8th Road, Sandun Town, Xihu District, Hangzhou City, Zhejiang ,PRC

Post Code：310030