MA35D1 Linux A/B 分区 OTA 更新机制

NuMicro® 64 位系列微控制器示例代码介绍

文件信息

应用简述	本示例程序说明如何于 MA35D1 系列开发平台下在 Linux 进行 OTA 更新 AB 分区的 rootfs。
BSP 版本	Linux-5.10.x
开发平台	NuMaker-HMI-MA35D1 V1.0

The information described in this document is the exclusive intellectual property of Nuvoton Technology Corporation and shall not be reproduced without permission from Nuvoton.

Nuvoton is providing this document only for reference purposes of NuMicro microcontroller and microprocessor based system design. Nuvoton assumes no responsibility for errors or omissions.

All data and specifications are subject to change without notice.

For additional information or questions, please contact: Nuvoton Technology Corporation.

www.nuvoton.com

1. 概述

OTA 原文为 Over-the-Air，在科技领域中泛指为在在线通过网络(WiFi、有线网络)对设备的韧体、软件或数据进行远程更新。此示例程序说明文档提供使用「swupdate」应用程序进行根文件系统(root filesystem：rootfs)的更新，其余通过 swupdate 更新 Kernel 与设备树的应用示例请参考 EC_MA35D1_OTA_Update_by_Linux_Stage_V1.00。

因为 swupdate 是基于根目录系统运作时而运行的应用程序，当需要更新根目录系统时，如果直接调用 swupdate 更新根目录系统本身是危险的，因为更新根目录系统时，无法预期那些文件是否是正在被根目录系统存取的，若贸然存取会有一个文件同时被两个程序重复存取而导致系统异常的风险，因此更新根目录系统时，事前规划出两个独立的根目录系统区域是必要且合理的，以下称为此两个独立的根目录系统为 A partition (rootfs area A) 与 B partition (rootfs area B) 。

1.1 环境配置

MA35D1 环境架设，产生镜像文件可参考新唐 MA35D1 MA35D1 Evaluation Board Quick Start。

本示例的核心在于先在 kernel 切出两块 rootfs partition，再利用 swupdate 应用过程调用 Shell Script 更新另一块没有被挂载(mounting) 状态的 rootfs 根目录系统与自动更新 uboot 参数，并在系统 rebooting 过程，自动挂载到刚才更新好的 rootfs 区域完成在线根目录文件系统更新，MA35D1 在 kernel 切出两块 rootfs partition 的排列如图 1-1 所示，以下的 AB 分区更新流程图如图 1-2 所示。

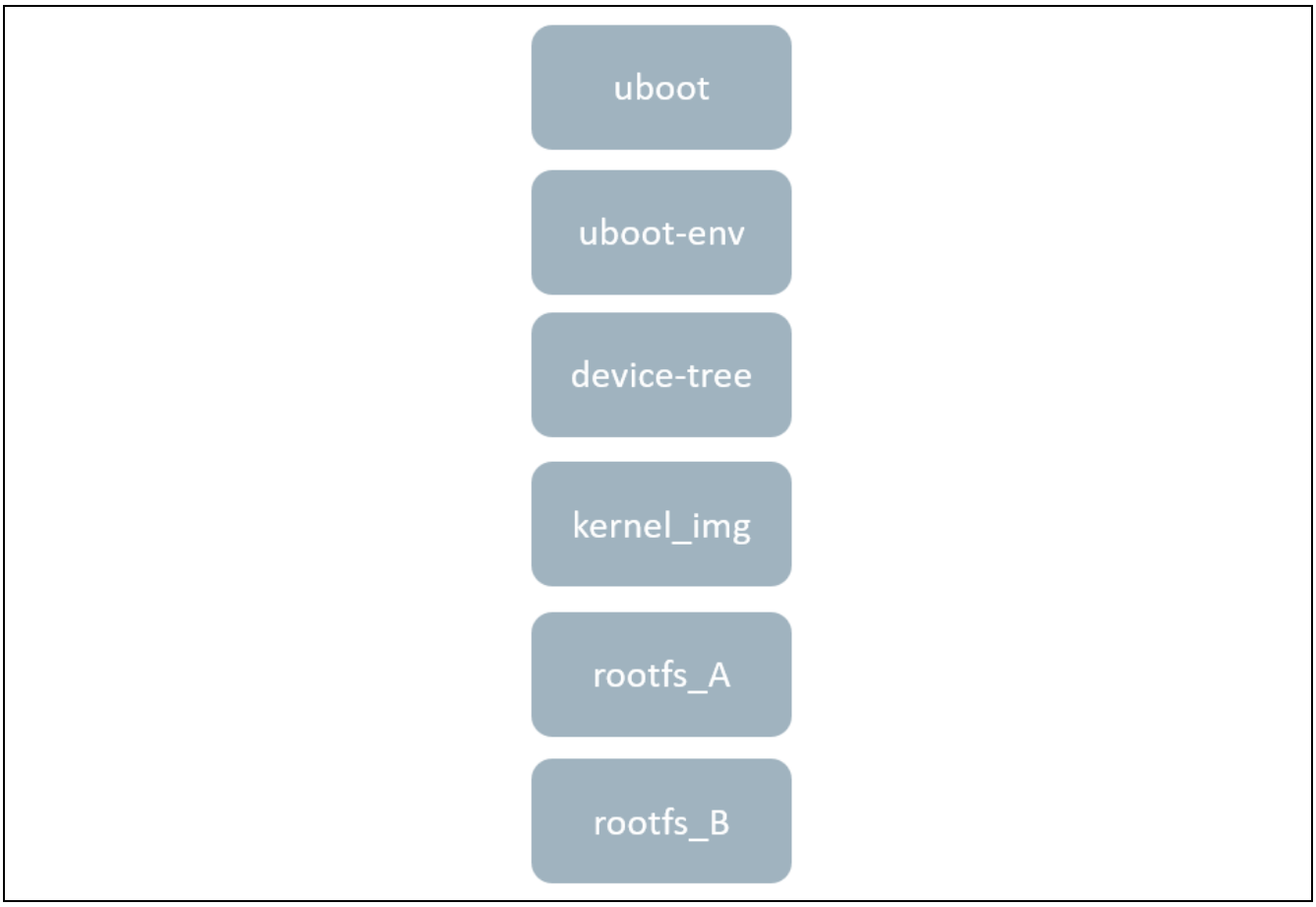


图 1-1 MA35D1 Kernel Partition 排列方式

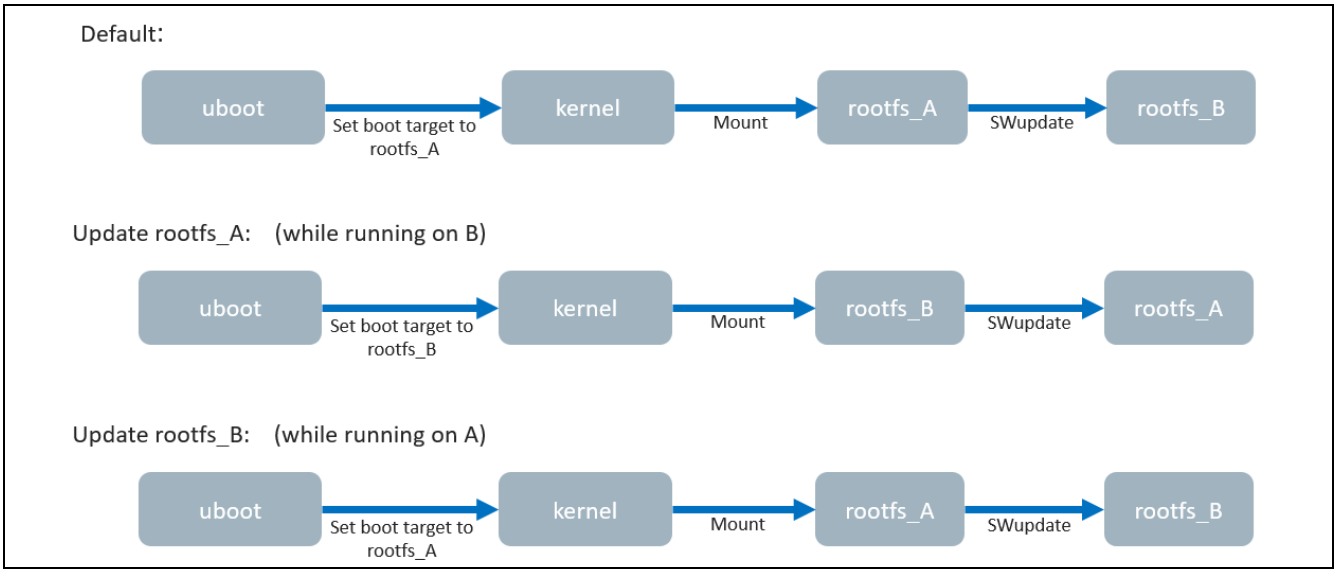


图 1-2 swupdate 更新 rootfs 流程图

2. 代码说明

2.1 eMMC/SD card 升级方式

以下为 eMMC/SD card AB Rootfs 更新流程：

1. 请先按照 EC_MA35D1_OTA_Update_by_Linux_Stage_V1.00 完成 swupdate 在 Buildroot_2024 上面的勾选。

   请修改 /package/swupdate/swupdate.config，并增加以下设置：

   CONFIG_SHELLSCRIPTHANDLER=y
   CONFIG_UBOOT_ENV=y

2. 修改完成后，请执行 swupdate rebuild，如图 2-1 所示。

```
$ make swupdate-rebuild
$ make
```



图 2-1 修改 swupdate.config

3. 请修改 /buildroot_2024/board/nuvoton/ma35xx/nuwriter/pack-sdcard.json，增加 B partition 分区的 eMMC/SD card 空间，此示例程序 B partition 分区设置为 1 GB。

```
{
  "image": [
    …
   {
     "offset": "37748736",
     "file": "rootfs.ext4",
      "type": 0
    },
    {
      "offset": "1140850688",
      "file": "rootfs.ext4",
      "type": 0
    }
```

```
  ]
}
```

4. 在 *uboot-env.txt* 文件，新增下列的 uboot 参数，让 swupdate 更新 rootfs 时，自动带入新参数，下次重启时自动指向到新的 rootfs 并完成开机，路径位于 */buildroot_2024/board/nuvoton/ma35xx/uboot-env.txt*。

   此 *uboot-env.txt* 同时也加入了正常开机到 kernel 时，会自动维持 rootfs_active_count 为 3，如果连续开机三次失败后，会自动跳转到 B 分区的机制，B 分区也是正常开机到 kernel 时，会自动维持 rootfs_active_count 为 3，如果连续开机三次失败后，会自动跳转到 A 分区，eMMC/SD card 与 SPI-NAND、NAND 皆可共享此 *uboot-env.txt*。

```
bootcmd=run check_boot_count; bootcmd=run set_mmc_part; run distro_bootcmd
bootdelay=3
baudrate=115200
   .
   .
kernelmem=256M
rootfs_active=a
mmc_part=1
mmc_block=mmcblk0p${mmc_part}
set_mmc_part=if test "${rootfs_active}" = "b"; then setenv mmc_part 2; else setenv
mmc_part 1; fi; setenv mmc_block mmcblk0p${mmc_part}
   .
   .
rootfs_active_count=3
check_boot_count=if test "${rootfs_active_count}" = "0"; then echo "Boot failed 3 times!
Rolling back..."; if test "${rootfs_active}" = "b"; then run switch_to_a; else run
switch_to_b; fi; setenv rootfs_active_count 3; saveenv; else if test
"${rootfs_active_count}" = "3"; then setenv rootfs_active_count 2; saveenv; elif test
"${rootfs_active_count}" = "2"; then setenv rootfs_active_count 1; saveenv; elif test
"${rootfs_active_count}" = "1"; then setenv rootfs_active_count 0; saveenv; fi; fi
   .
   .
```

5. 请更新 /buildroot_2024/board/nuvoton/ma35xx/post-image.sh，更新完成后，通过 NuWriter 刻录时，默认会多切出 B partition 分区 (A partition 分区：244 MB，B partition 分区：1 GB)。

6. 新增 *fw_env.config* 文件，并请放置在 */output/target/etc/fw_env.config*。此配置文件提供 linux 下的 fw_printenv 与 fw_setenv 使用，使其能正确存取 uboot environment。以下设置表示 uboot environment 存储于 */dev/mmcblk0*，起始 offset 为 0x40000，environment size 为 0x10000。

```
/dev/mmcblk0 0x40000 0x10000
```

7. 修改完成后，请执行 kernel rebuild 并完成目标版刻录。

```
$ make arm-trusted-firmware-rebuild uboot-rebuild linux-rebuild
$ make
```

8. 正常开机到 kernel 后，请执行 boot_success.sh，让 rootfs_active_count 变量保持为 3，而不会进入重启倒数判断。

```
MA35D1# ./boot_success.sh
```

9. 更新 *sw-description* 与 *generate_swu.sh* 为 eMMC/SD card 版本。

```
{
  "software": {
    "version": "0.1.0",
    "description": ["Linux kernel and dtb and rootfs update for MA35 eMMC/SDcard"],
    "hardware-compatibility": ["1.0", "1.2", "1.3"],
    "scripts": [
      {
        "filename": "prepare_ab_update.sh",
        "type": "preinstall"
      },
      {
        "filename": "finish_ab_update.sh",
        "type": "postinstall"
      }
    ],
    "images": [
      {
        "version": "5.10",
        "name": "nand-kernel device tree",
        "filename": "Image.dtb",
        "device": "/dev/mmcblk0",
        "type": "raw",
        "offset": 2883584
      },
      {
        "version": "5.10",
        "name": "nand-kernel",
        "filename": "Image",
        "device": "/dev/mmcblk0",
        "type": "raw",
        "offset": 3145728
      },
      {
        "version": "5.10",
        "name": "rootfs_update",
        "filename": "rootfs.ext2",
        "device": "/tmp/update_rootfs",
        "type": "raw"
      }
    ]
  }
}
```

generate_swu.sh

```
# Assigns the value "sw-description Image" to the variable FILES, which contains a space-separated list of filenames to be included in the container
FILES="sw-description Image Image.dtb rootfs.ext2 prepare_ab_update.sh finish_ab_update.sh"
```

10. 请将 rebuild 成功的 Image、*Image.dtb*、*rootfs.ext2* 与 *prepare_ab_update.sh* 脚本、*finish_ab_update.sh* 脚本、*sw-description* 放在同一个文件夹中，并执行 *generate_swu.sh*，执行完成后会得到 *pack_1.0.swu* 镜像文件，如图 2-2 所示。

```
ubuntu:~$ ./generate_swu.sh
```

```
nuvoton@ubuntu:~/                                                    /eMMC_SDcard$ ./generate_swu.s
sw-description
Image
Image.dtb
rootfs.ext2
prepare_ab_update.sh
finish_ab_update.sh
519199 blocks
nuvoton@ubuntu:~/                                                    /eMMC_SDcard$ ls -l
total 519228
-rwxrw-rw- 1 nuvoton nuvoton      1401 May 12 02:39 finish_ab_update.sh
-rwxrwxr-x 1 nuvoton nuvoton      1079 May 12 03:21 generate_swu.sh
-rw-r--r-- 1 nuvoton nuvoton  14123520 May 12 03:04 Image
-rwxr-xr-x 1 nuvoton nuvoton     43259 May 12 03:04 Image.dtb
-rw-rw-r-- 1 nuvoton nuvoton 265829888 May 12 05:08 pack_1.0.swu
-rwxrw-rw- 1 nuvoton nuvoton      1618 May 12 02:36 prepare_ab_update.sh
-rw-r--r-- 1 nuvoton nuvoton 251658240 May 12 03:04 rootfs.ext2
-rw-rw-r-- 1 nuvoton nuvoton       962 May 12 03:22 sw-description
-rw-rw-r-- 1 nuvoton nuvoton      1333 May 19  2023 sw-description 2021
nuvoton@ubuntu:~/                                                    /eMMC_SDcard$
```

图 2-2 产生 pack 镜像文件

2.2 SPI-NAND 升级方式

1. 请先按照 EC_MA35D1_OTA_Update_by_Linux_Stage_V1.00 完成 swupdate 在 Buildroot_2024 上面的勾选。

2. 请参考 eMMC/SD card 更新 swupdate，修改 */package/swupdate/swupdate.config*。

3. 修改完成后，请执行 swupdate rebuild。

```
$ make swupdate-rebuild
$ make
```

4. 请修改 uboot 内容(uboot-menuconfig)，将 uboot-env 的空间调整成 1 MB，为了让 AB partition 的更新顺利对齐，将 rootfs 修改为 200 MB。

```
$ make uboot-menuconfig

   Command line interface  --->
             Filesystem commands  --->
   mtdparts=spi-nand0:3m(uboot),1m(uboot-env),256k(device-tree),24m(kernel),200m(rootfs)
```

```
$ make
```

5. 请修改 */buildroot_2024/output/build/linux-custom/arch/arm64/boot/dts/nuvoton/ma35d1.dtsi*，修改 spinand-uboot-env，调整到 0x100000 (1 MB)，并增加 B partition 分区(spinand-rootfs-b)的 SPI-NAND 空间，此示例程序 B partition 分区设置为 200 MB。

```
  flash: flash@0 {
      compatible = "spi-nand";
      reg = <0 0x0 0x2000000>;
      spi-max-frequency = <50000000>;
      #address-cells = <1>;
      #size-cells = <1>;

      uboot@0 {
        label = "spinand-uboot";
        reg = <0x0 0x300000>;
      };
      uboot-env@300000 {
        label = "spinand-uboot-env";
        reg = <0x300000 0x100000>;
      };
      device-tree@400000 {
        label = "spinand-device-tree";
        reg = <0x400000 0x40000>;
      };
      kernel@440000 {
        label = "spinand-kernel";
        reg = <0x440000 0x1800000>;
      };
      rootfs@1c00000 {
        label = "spinand-rootfs";
        reg = <0x1c00000 0xC800000>;
      };
      rootfs-b@E400000 {
        label = "spinand-rootfs-b";
        reg = <0xE400000 0xC800000>;
      };

    };
```

6. 在 uboot-env.txt 文件，新增下列的 uboot 参数，让 swupdate 更新 rootfs 时，自动带入新参数，下次重启时自动指向到新的 rootfs 并完成开机，如果是 MA35D1-SOM，开机会同时开启 NAND driver 与 SPI-NAND Driver。这里的示范示例是 spinand_ubiblock 就设置为 9，spinand_uboot_para_blk 设置为 6。

   如果开机没有包含 NAND driver 只有单独开启 SPI-NAND Driver，此时 spinand_ubiblock 开机就设置为 4，spinand_uboot_para_blk 设置为 1，路径位于 */buildroot_2024/board/nuvoton/ma35xx/uboot-env.txt*。

此 *uboot-env.txt* 同时也加入了正常开机到 kernel 时，会自动维持 rootfs_active_count 为 3，如果连续开机三次失败后，会自动跳转到 B 分区的机制，B 分区也是正常开机到 kernel 时，会自动维持 rootfs_active_count 为 3，如果连续开机三次失败后，会自动跳转到 A 分区，eMMC/SD card 与 SPI-NAND、NAND 皆可共享此 *uboot-env.txt*。

```
bootcmd=run check_boot_count; run distro_bootcmd
bootdelay=3
baudrate=115200

   .
   .
rootfs_active=a
spinand_ubiblock=9
nand_ubiblock=4
spinand_uboot_para_blk=6
switch_to_a=setenv rootfs_active "a"; setenv nand_ubiblock 4; setenv spinand_ubiblock 9
switch_to_b=setenv rootfs_active "b"; setenv nand_ubiblock 5; setenv spinand_ubiblock 10
rootfs_active_count=3
check_boot_count=if test "${rootfs_active_count}" = "0"; then echo "Boot failed 3 times!
Rolling back..."; if test "${rootfs_active}" = "b"; then run switch_to_a; else run
switch_to_b; fi; setenv rootfs_active_count 3; saveenv; else if test
"${rootfs_active_count}" = "3"; then setenv rootfs_active_count 2; saveenv; elif test
"${rootfs_active_count}" = "2"; then setenv rootfs_active_count 1; saveenv; elif test
"${rootfs_active_count}" = "1"; then setenv rootfs_active_count 0; saveenv; fi; fi    .
   .
   .
```

7. 新增 *fw_env.config* 文件，并请放置在 */output/target/etc/fw_env.config*。此配置文件提供 linux 下的 fw_printenv 与 fw_setenv 使用，使其能正确存取 uboot environment。
   以下设置表示 uboot environment 储存于 */dev/ubi1_0*，起始 offset 为 0x00000，environment size 为 0x10000，erase size 为 0x1F000。

```
/dev/ubi1_0    0x00000  0x10000  0x1F000
```

8. 修改完成后，请执行 kernel rebuild 并完成目标版刻录。

```
$ make arm-trusted-firmware-rebuild uboot-rebuild linux-rebuild
$ make
```

9. 正常开机到 kernel 后，请执行 *boot_success.sh*，让 rootfs_active_count 变量保持为 3，而不会进入重启倒数判断。

```
MA35D1# ./boot_success.sh
```

10. 更新 *sw-description* 与 *generate_swu.sh* 为 SPI-NAND 版本。

```
{
  "software": {
    "version": "0.1.0",
```

```
    "description": ["Linux kernel and dtb and rootfs update for MA35 SPI-NAND"],
    "hardware-compatibility": ["1.0", "1.2", "1.3"],
    "scripts": [
      {
        "filename": "prepare_ab_update.sh",
        "type": "preinstall"
      },
      {
        "filename": "finish_ab_update.sh",
        "type": "postinstall"
      }
    ],
    "images": [
      {
        "version": "5.10",
        "name": "spinand-kernel device tree",
        "filename": "Image.dtb",
        "device": "/dev/mtdblock2",
        "type": "raw"
      },
      {
        "version": "5.10",
        "name": "spinand-kernel",
        "filename": "Image",
        "device": "/dev/mtdblock3",
        "type": "raw"
      },
      {
        "version": "5.10",
        "name": "rootfs_update",
        "filename": "rootfs.ubi",
        "device": "/tmp/update_rootfs",
        "type": "raw"
      }
    ]
  }
}
```

generate_swu.sh

```
# Assigns the value "sw-description Image" to the variable FILES, which contains a space-separated list of filenames to be included in the container
FILES="sw-description Image Image.dtb rootfs.ubi prepare_ab_update.sh finish_ab_update.sh"
```

11. 请将 rebuild 成功的 Image、*Image.dtb*、*rootfs.ubi* 与 *prepare_ab_update.sh* 脚本、*finish_ab_update.sh* 脚本、*sw-description* 放在同一个文件夹中，并执行 *generate_swu.sh*，执行完成后会得到 *pack_1.0.swu*，如图 2-3 所示。

```
ubuntu:~$ ./generate_swu.sh
```

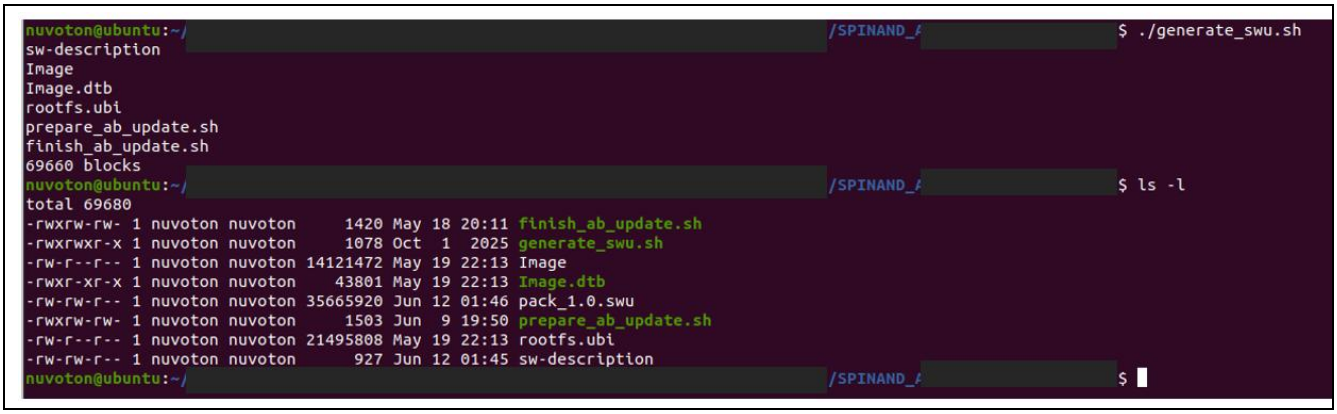


图 2-3 产生 pack 镜像文件

2.3 NAND 升级方式

1. 请先按照 EC_MA35D1_OTA_Update_by_Linux_Stage_V1.00 完成 swupdate 在 Buildroot_2024 上面的勾选。

2. 请参考 eMMC/SD card 更新 swupdate，修改 */package/swupdate/swupdate.config*。

3. 修改完成后，请执行 swupdate rebuild。

```
$ make swupdate-rebuild
$ make
```

4. 请修改 */buildroot_2024/output/build/linux-custom/arch/arm64/boot/dts/nuvoton/ma35d1.dtsi*，增加 B partition 分区(nand-rootfs-b)的 NAND 空间，此示例程序 B partition 分区设置为 200 MB。

```
  partitions {
      compatible = "fixed-partitions";
      #address-cells = <1>;
      #size-cells = <1>;

      uboot@0 {
        label = "nand-uboot";
        reg = <0x0000000 0x300000>;
        read-only;
      };
      uboot-env@300000 {
        label = "nand-uboot-env";
        reg = <0x300000 0xC0000>;
      };
      device-tree@3c0000 {
        label = "nand-device-tree";
        reg = <0x3C0000 0x40000>;
      };
      kernel@400000 {
        label = "nand-kernel";
        reg = <0x400000 0x1800000>;
      };
      rootfs@1c00000 {
        label = "nand-rootfs";
```

```
        reg = <0x1C00000 0xC800000>;
      };
      rootfs-b@E400000 {
        label = "nand-rootfs-b";
        reg = <0xE400000 0xC800000>;
      };

    };
```

5. 在 *uboot-env.txt* 文件，新增下列的 uboot 参数，让 swupdate 更新 rootfs 时，自动带入新参数，下次重启时自动指向到新的 rootfs 并完成开机。这里的示范示例是 nand_ubiblock 初始设置为 4、rootfs_active 为 a，路径位于 */buildroot_2024/board/nuvoton/ma35xx/uboot-env.txt*。

   此 *uboot-env.txt* 同时也加入了正常开机到 kernel 时，会自动维持 rootfs_active_count 为 3，如果连续开机三次失败后，会自动跳转到 B 分区的机制，B 分区也是正常开机到 kernel 时，会自动维持 rootfs_active_count 为 3，如果连续开机三次失败后，会自动跳转到 A 分区，eMMC/SD card 与 SPI-NAND、NAND 皆可共享此 *uboot-env.txt*。

```
bootcmd=run check_boot_count; run distro_bootcmd
bootdelay=3
baudrate=115200

   .
   .
rootfs_active=a
nand_ubiblock=4
switch_to_a=setenv rootfs_active "a"; setenv nand_ubiblock 4; setenv spinand_ubiblock 9
switch_to_b=setenv rootfs_active "b"; setenv nand_ubiblock 5; setenv spinand_ubiblock 10
rootfs_active_count=3
   .
   .
   .

check_boot_count=if test "${rootfs_active_count}" = "0"; then echo "Boot failed 3 times!
Rolling back..."; if test "${rootfs_active}" = "b"; then run switch_to_a; else run
switch_to_b; fi; setenv rootfs_active_count 3; saveenv; else if test
"${rootfs_active_count}" = "3"; then setenv rootfs_active_count 2; saveenv; elif test
"${rootfs_active_count}" = "2"; then setenv rootfs_active_count 1; saveenv; elif test
"${rootfs_active_count}" = "1"; then setenv rootfs_active_count 0; saveenv; fi; fi
   .
   .
```

6. 新增 *fw_env.config* 文件，并请放置在 */output/target/etc/fw_env.config*。此配置文件提供 linux 下的 fw_printenv 与 fw_setenv 使用，使其能正确存取 uboot environment。
   以下设定表示 uboot environment 储存于 */dev/mtd1*，起始 offset 为 0x00000，environment size 为 0x10000，erase size 为 0x20000。

```
/dev/mtd1 0x0 0x10000 0x20000
```

7. 修改完成后，请执行 kernel rebuild 并完成目标版刻录。

```
$ make arm-trusted-firmware-rebuild uboot-rebuild linux-rebuild
$ make
```

8. 正常开机到 kernel 后，请执行 boot_success.sh，让 rootfs_active_count 变量保持为 3，而不会进入重启倒数判断。

```
MA35D1# ./boot_success.sh
```

9. 更新 *sw-description* 与 *generate_swu.sh* 为 NAND 版本。

```
{
  "software": {
    "version": "0.1.0",
    "description": ["Linux kernel and dtb and rootfs update for MA35 NAND"],
    "hardware-compatibility": ["1.0", "1.2", "1.3"],
    "scripts": [
      {
        "filename": "prepare_ab_update.sh",
        "type": "preinstall"
      },
      {
        "filename": "finish_ab_update.sh",
        "type": "postinstall"
      }
    ],
    "images": [
      {
        "version": "5.10",
        "name": "nand-kernel device tree",
        "filename": "Image.dtb",
        "device": "/dev/mtdblock2",
        "type": "raw"
      },
      {
        "version": "5.10",
        "name": "nand-kernel",
        "filename": "Image",
        "device": "/dev/mtdblock3",
        "type": "raw"
      },
      {
        "version": "5.10",
        "name": "rootfs_update",
        "filename": "rootfs.ubi",
        "device": "/tmp/update_rootfs",
        "type": "raw"
      }
    ]
  }
}
```

generate_swu.sh

```
# Assigns the value "sw-description Image" to the variable FILES, which contains a space-separated list of filenames to be included in the container
FILES="sw-description Image Image.dtb rootfs.ubi prepare_ab_update.sh finish_ab_update.sh"
```

10. 请将 rebuild 成功的 Image、*Image.dtb*、*rootfs.ubi* 与 *prepare_ab_update.sh* 脚本、*finish_ab_update.sh* 脚本、sw-description 放在同一个文件夹中，并执行 *generate_swu.sh*，执行完成后会得到 *pack_1.0.swu*。

```
ubuntu:~$ ./generate_swu.sh
```

3. 软件与硬件需求

3.1 软件需求

- BSP 版本
  - Linux-5.10.x
  - Swupdate-2023.12

3.2 硬件需求

- 电路组件
  - NuMaker-HMI-MA35D1 V1.0， 如图 3-1 所示。

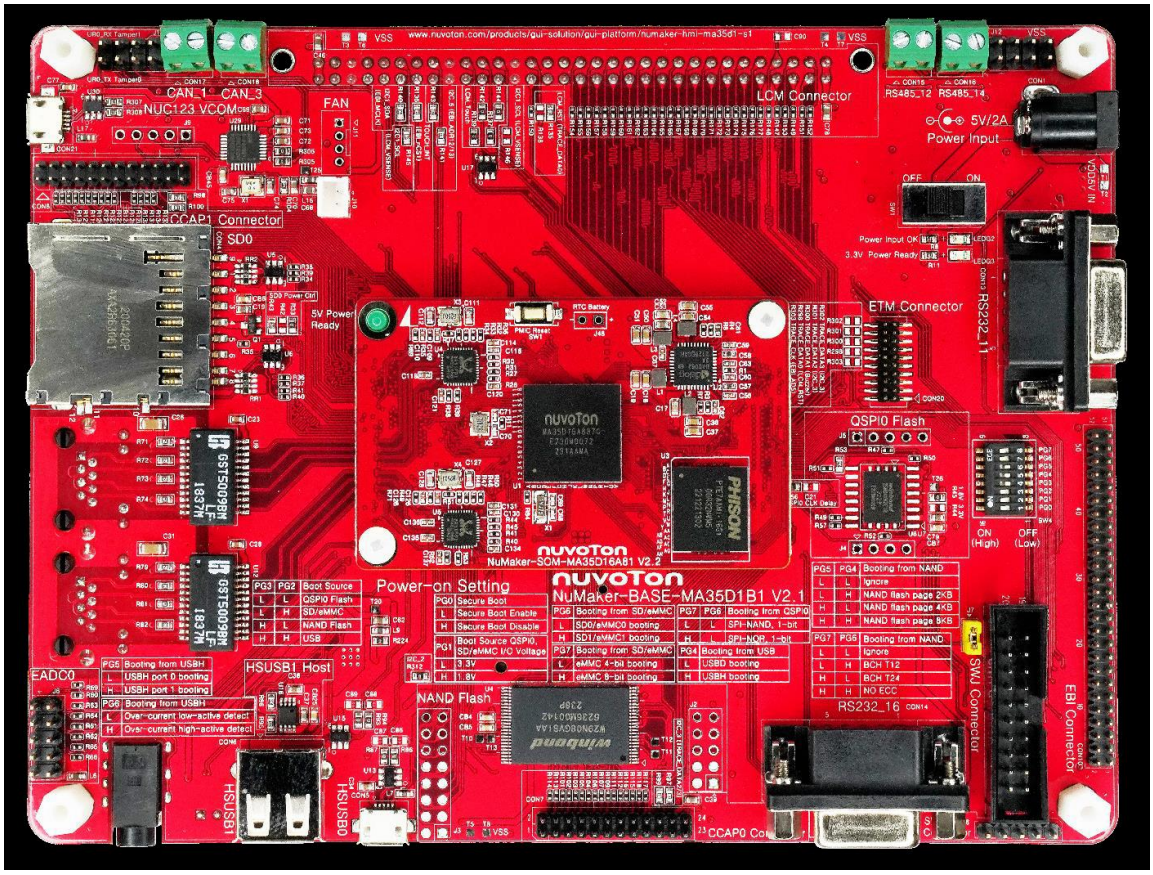


图 3-1 NuMaker-HMI-MA35D1 V1.0

4. 目录信息

目录结构如图 4-1 所示。

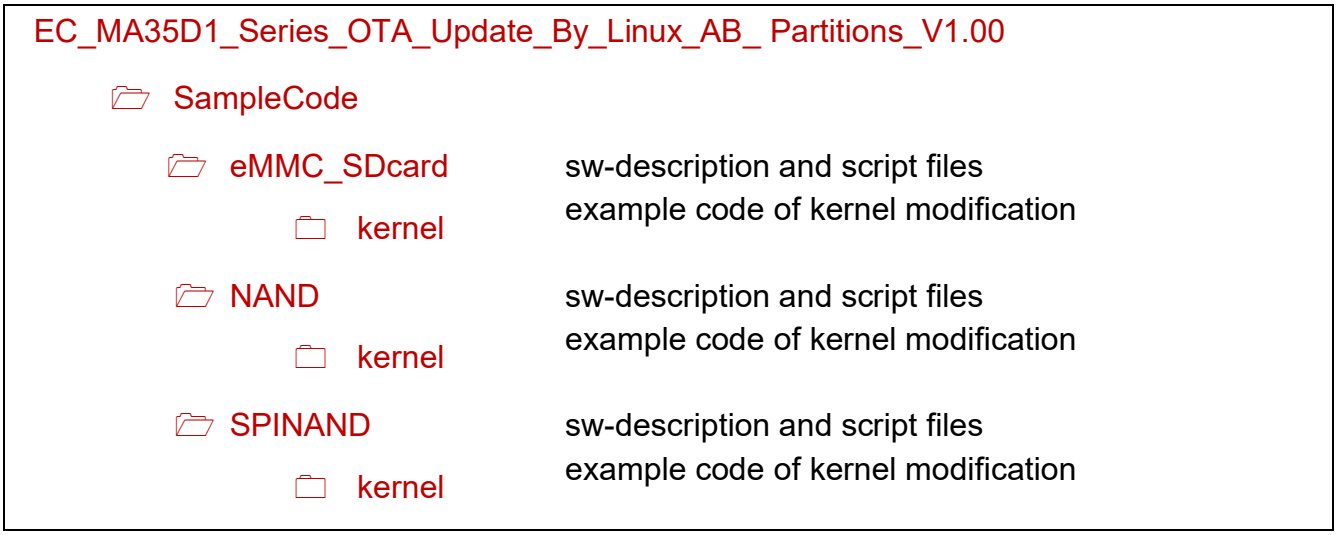


图 4-1 目录信息

5. 范例程序执行

1. 在 MA35D1 设备上，可以通过设置设备使用 DHCP（动态主机配置协议）从路由器取得 IP 地址，本示例程序获得的示例 IP 为 192.168.0.175。

```
MA35D1# ifconfig eth0 up
MA35D1# udhcpc -i eth0
udhcpc: started, v1.33.1
udhcpc: sending discover
udhcpc: sending discover
udhcpc: sending select for 192.168.0.175
udhcpc: lease of 192.168.0.175 obtained, lease time 7200
deleting routers
adding dns 192.168.0.1
```

2. 确保计算机和 MA35D1 设备连接到同一个网络后，输入 "192.168.0.175:8080"。

3. 点击“Click here, or drag and drop a software update image file to this area.”

4. 参考第二章与实际选择的更新方式，执行 *generate_swu.sh* 并产生出的 *pack_1.0.swu*，将打包好 *pack_1.0.swu* 拖拉并上传即可完成 AB 自动分区更新，eMMC/SD card 更新如图 5-1 所示， SPI-NAND / NAND 更新如图 5-2 所示。

5. 分区自动更新后，可通过开机信息(log)查看是否更新成功，如图 5-3 所示。

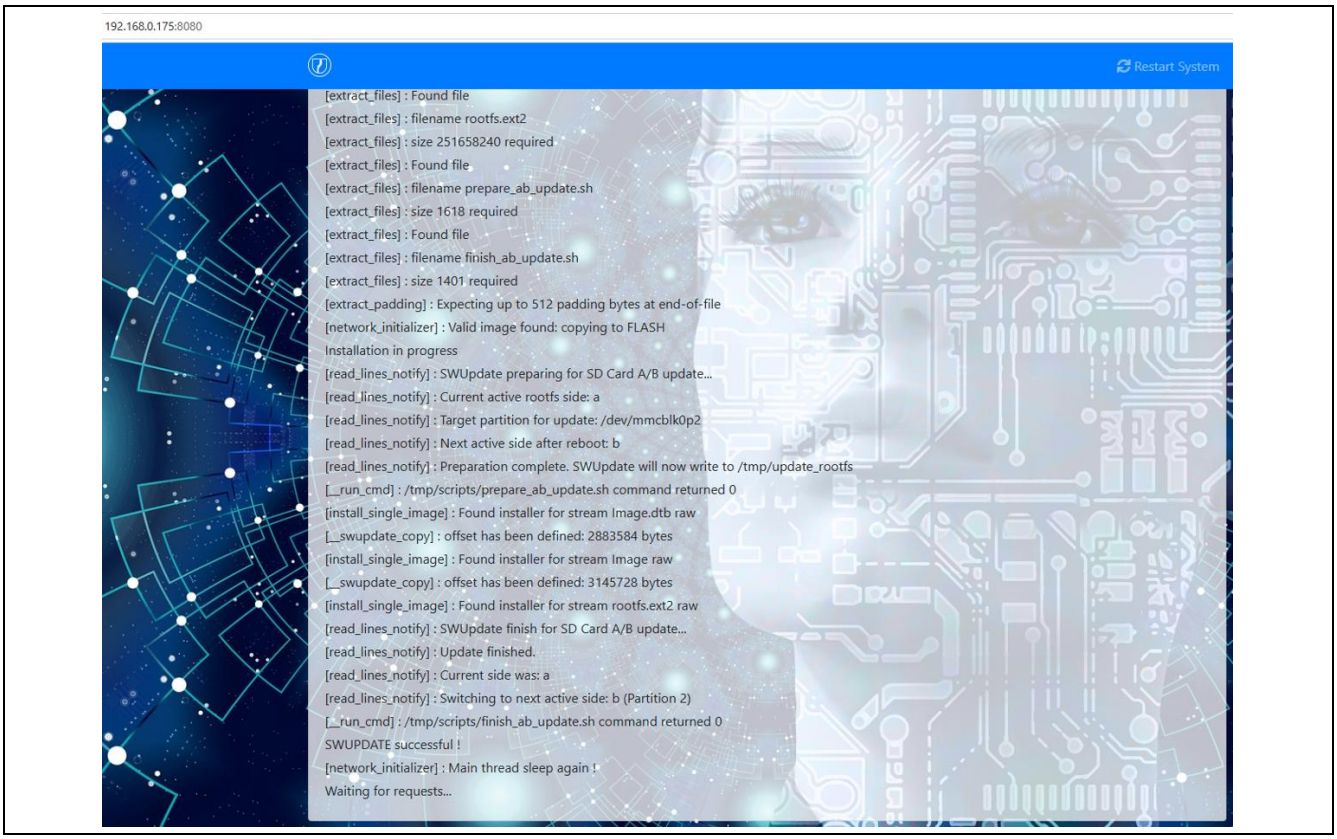


图 5-1 eMMC/SD card 通过 swupdate 更新成 B 分区 rootfs



图 5-2 SPI-NAND / NAND 通过 swupdate 更新成 B 分区 rootfs

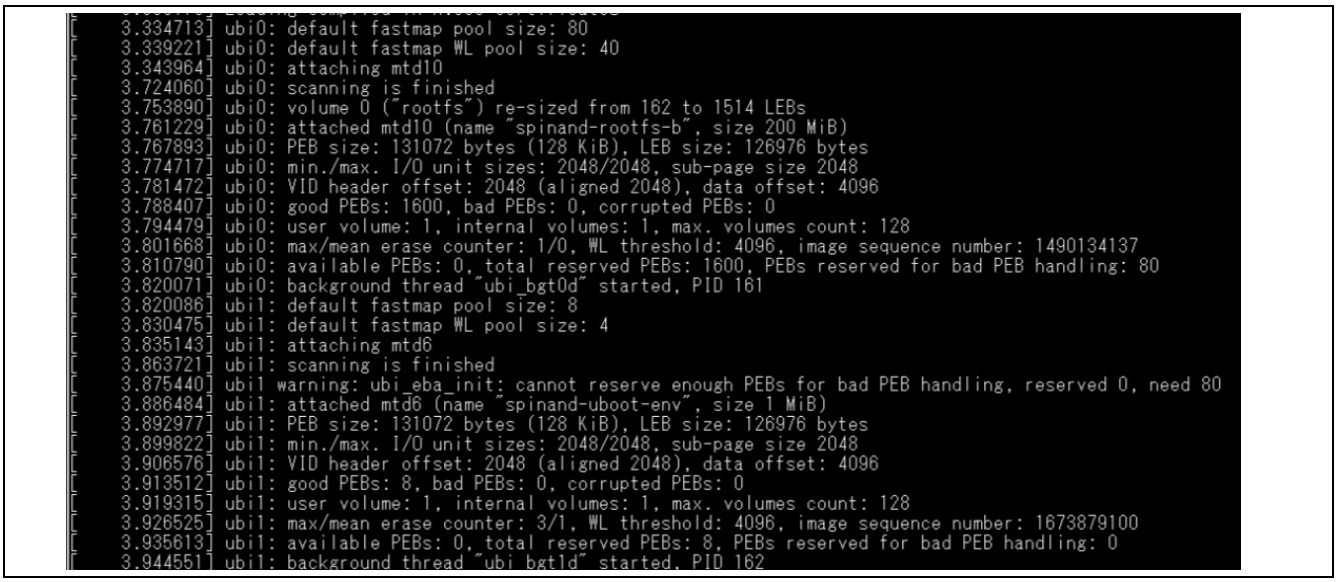


图 5-3 通过 swupdate 更新成 B 分区成功的 SPI-NAND 开机 log

6. 修订记录

Date	Revision	Description
2026.06.12	1.00	初始发布。

Important Notice

Nuvoton Products are neither intended nor warranted for usage in systems or equipment, any malfunction or failure of which may cause loss of human life, bodily injury or severe property damage. Such applications are deemed, "Insecure Usage".

Insecure usage includes, but is not limited to: equipment for surgical implementation, atomic energy control instruments, airplane or spaceship instruments, the control or operation of dynamic, brake or safety systems designed for vehicular use, traffic signal instruments, all types of safety devices, and other applications intended to support or sustain life.

All Insecure Usage shall be made at customer's risk, and in the event that third parties lay claims to Nuvoton as a result of customer's Insecure Usage, customer shall indemnify the damages and liabilities thus incurred by Nuvoton.

Please note that all data and specifications are subject to change without notice.
All the trademarks of products and companies mentioned in this datasheet belong to their respective owners.