

MA35D1 OTA Update by Linux Stage

NuMicro® 64/32位系列微控制器範例程式碼介紹

文件資訊

應用簡述	本範例程式說明如何於 MA35D1 系列開發平台下在 Linux 進行 OTA 更新。
BSP 版本	Linux-5.10.x
開發平台	NuMaker-IoT-MA35D1

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

此範例程式說明文件提供使用「swupdate」應用程式在MA35D1的多個儲存裝置上更新Linux Kernel、設備樹以及根檔案系統。儲存裝置包括NAND、SPI NAND、SD卡和eMMC。更新的過程涉及生成包含韌體更新的打包文件以及描述韌體更新的儲存地址資訊的標頭檔案。

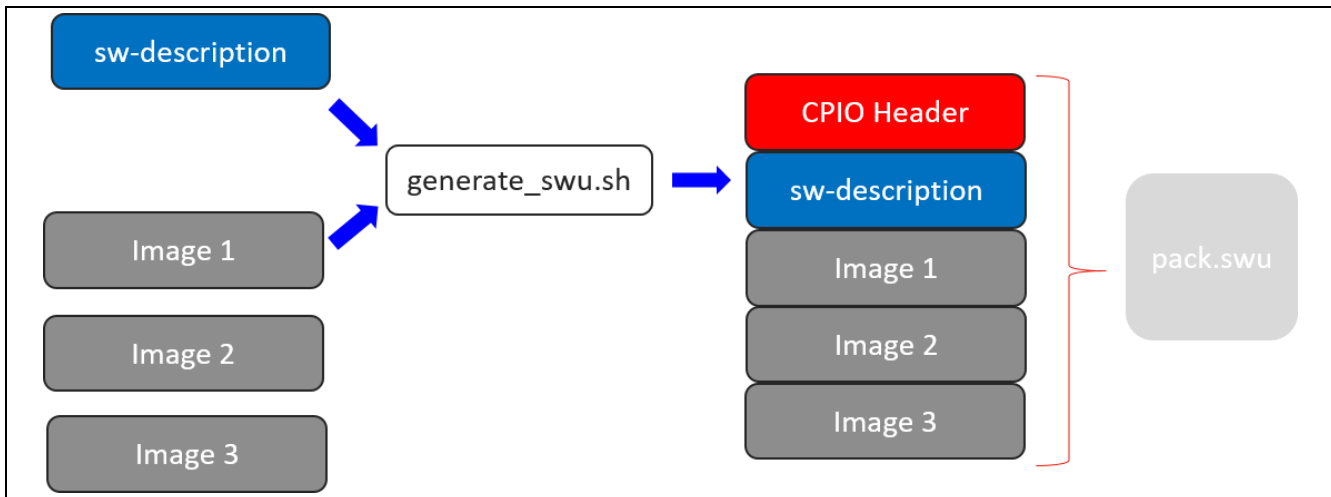


圖 1-1 更新包生成流程

為了進行韌體更新，首先需要生成包含已更新的Kernel、設備樹和根檔案系統的打包文件。可以經由腳本（`generate_swu.sh`）將更新的文件（Image1、Image2、Image3...）合併為一個文件來完成。生成打包文件後，下一步是創建一個標頭文件，該文件包含每個韌體更新的儲存地址信息。這個標頭文件通常被命名為「sw-description」，並包含以下信息：

- 韌體的版本
- 硬體相容性資訊
- 每個韌體更新的文件資訊，包括檔案名稱、路徑和儲存位置

主機端生成pack.swu文件後，打開網頁瀏覽器，連接到運行在MA35D1開發板上的swupdate網頁伺服器。然後將pack.swu文件拖放到swupdate網頁上，swupdate將開始接收該文件並解析sw-description以將相應的文件更新到對應的儲存位置。

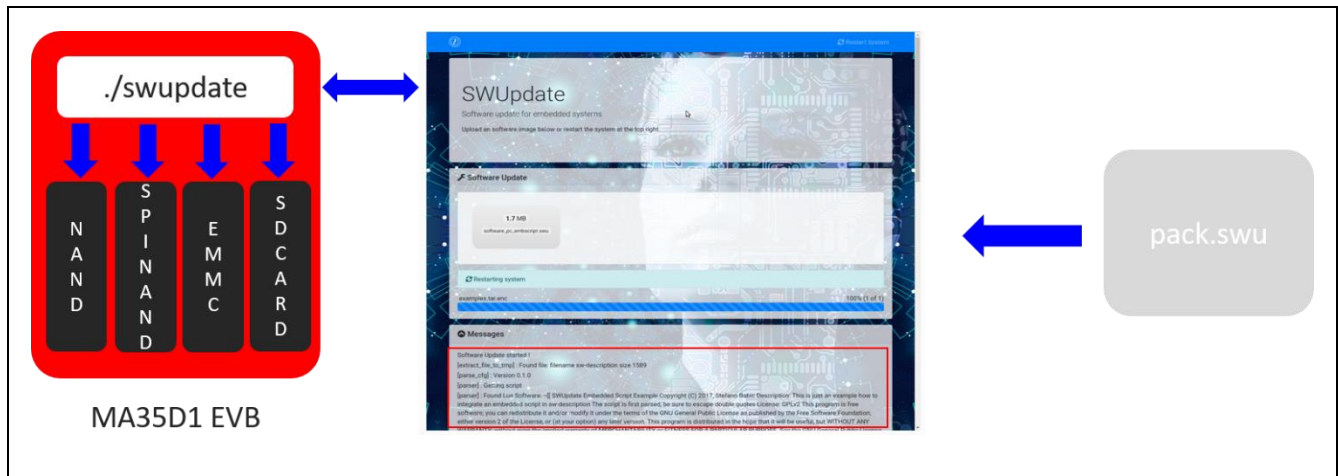


圖 1-2 swupdate Web Server 更新框架

總體而言，使用「swupdate」應用程式在多個儲存設備上進行韌體更新涉及生成韌體更新的打包文件，並創建一個帶有每個更新的儲存地址信息的標頭文件。擁有這些文件後，可以使用「swupdate」應用程式對目標設備進行韌體更新。

1.1 環境設定

MA35D1環境架設，產生映像檔可參考新唐 [MA35D1 Evaluation Board Quick Start](#)。

● Buildroot

於 Buildroot 的 menuconfig 中啟用「swupdate」所需的套件，請按照以下步驟進行：

1. 透過執行 make menuconfig 命令來打開 Buildroot 的配置菜單。
2. 在套件選擇菜單中，選擇相應的選項並按下空格鍵來啟用每個套件。可以使用 / 鍵按名稱搜索套件。
 - a. 啟用 **BR2_PACKAGE_UBOOT_TOOLS_FWPRINTENV** 以包含 U-Boot Tools fw_printenv 套件。
 - b. 啟用 **BR2_PACKAGE_MTD** 以包含 MTD 套件。
 - c. 啟用 **BR2_PACKAGE_SWUPDATE** 以包含 swupdate 套件。
 - d. 啟用 **BR2_PACKAGE_MMC_UTILS** 以包含 mmc-utils 套件。

3. 啟用所有所需的套件後，退出 menuconfig 並保存配置。
4. 通過 make 命令使用更新的配置來構建 Buildroot 系統。

編譯完成後，使用 NuWriter 將映像燒錄到 MA35D1 開發板

1.2 使用 sw-description 為不同儲存類型創建韌體更新包

sw-description 告知 swupdate 更新文件的位置。此章節將說明如何生成 sw-description 文件的過程，考慮到特定的存儲設備（NAND/SPI NAND/SD卡/eMMC）。此外，也提供將 sw-description 文件與韌體映像打包在一起的範例，實現完整的韌體更新包。

1.2.1 MA35D1 Storages Partition Layout

1.2.1.1 NAND

MA35D1 的 NAND partition layout 可以在 [ma35d1.dtsi](#) 找到

```
nand:nand@401A0000 {
    ...
    partitions {
        compatible = "fixed-partitions";
        #address-cells = <1>;
        #size-cells = <1>;

        uboot@0 {
            label = "nand-uboot";
            reg = <0x00000000 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 0x6400000>;
        };
    };
};
```

若要在 sw-description 描述更新 MA35D1 NAND 分區的內核，可以參考以下內容：

```
software =
{
    version = "0.1.0";
    description = "Linux kernel update in NAND";
    images: (
        {
            version = "5.10";
            name = "nand-kernel";
            filename = "Image";
            device = "/dev/mtdblock3";
            type = "raw";
        }
    );
}
```

此範例中，"Image" 是內核映像檔案的文件名，"/dev/mtdblock3" 是實際內核映像檔案的 mtdblock

1.2.1.2 SPI NAND

MA35D1 的 SPI NAND partition layout 可以在 [ma35d1.dtsi](#) 找到

```
qspi0: spi@40680000 {
    flash: flash@0 {
        compatible = "spi-nand";
        reg = <0 0x0 0x2000000>;
        spi-ax-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 0xc0000>;
        };
        device-tree@3c0000 {
            label = "spinand-device-tree";
            reg = <0x3c0000 0x40000>;
        };
        kernel@400000 {
            label = "spinand-kernel";
            reg = <0x400000 0x1800000>;
        };
        rootfs@1c00000 {
            label = "spinand-rootfs";
            reg = <0x1c00000 0x6400000>;
        };
    };
    ...
};
```

若要在 sw-description 描述更新 MA35D1 SPI NAND 分區的內核，可以參考以下內容：

```
software =
{
    version = "0.1.0";
    description = " Linux kernel update in SPI NAND";
    images: (
        {
            version = "5.10";
            name = "nand-kernel";
            filename = "Image";
            device = "/dev/mtdblock8";
            type = "raw";
        }
    );
}
```

此範例中，"Image" 是內核映像檔的檔名，"/dev/mtdblock8" 是指向實際內核映像檔的 MTD 區塊。

請注意，MTD (Memory Technology Device) 區塊編號可能會根據設備樹的配置而有所不同。對於具有 NAND 和 SPI NAND 連接的 NuMaker-IoT-MA35D1，MTD 區塊的編號如下：

NAND：MTD 區塊的編號為 mtdblock0 到 mtdblock4。

SPI NAND：MTD 區塊的編號為 mtdblock5 到 mtdblock9。

需參考設備樹配置或目標設備來確定每個存儲類型的正確 MTD 區塊編號。這些資訊將用於 sw-description 檔案中，以指定用於韌體更新的正確分區。

1.2.1.3 eMMC/SDcard

MA35D1 的 eMMC/SDcard partition layout 可以在 [pack-sdcard.json](#) 找到

```
{
  "image": [
    {
      "offset": "0x2c0000",
      "file": "Image.dtb",
      "type": 0
    },
    {
      "offset": "0x300000",
      "file": "Image",
      "type": 0
    },
    {
      "offset": "37748736",
      "file": "rootfs.ext4",
      "type": 0
    }
  ]
}
```

若要在 sw-description 描述更新 MA35D1 eMMC/SDcard 分區的內核，可以參考以下內容：

```
software =
{
    version = "0.1.0";
    description = " Linux kernel update in eMMC/SDcard";
    images: (
        {
            filename = "Image";
            device = "/dev/mmcblk0";
            offset = "0x300000";
        }
    );
}
```

1.2.2 韌體映像的封裝

為了將sw-description文件和其他韌體更新文件封裝成單一的包，請按照以下步驟進行操作：

1. 準備必要的文件：確保將 sw-description 文件以及其他韌體文件（如內核映像、設備樹二進制文件和根文件系統映像）放在同一個文件夾中。
2. 創建封裝腳本：開發一個腳本來處理封裝過程。該腳本從指定的文件夾中收集文件並將它們封裝成單一的文件。以下是一個示例：

```
#!/bin/sh
CONTAINER_VER="1.0"
PRODUCT_NAME="pack"
FILES="sw-description Image"
for i in $FILES;do
    echo $i;done | cpio -ov -H crc > ${PRODUCT_NAME}_${CONTAINER_VER}.swu
```

此腳本會將 sw-description 和 Image 文件打包成一個名為 pack.swu 的 cpio 格式文件

3. 執行打包腳本：運行打包腳本來創建 pack.swu 文件，該文件包含了壓縮的韌體更新文件和 sw-description 文件。

將sw-description文件和其他韌體更新文件放入同一個文件夾中，運行打包腳本，就可以生成 pack.swu文件。該文件是一個完整的韌體更新包，可以通過swupdate應用程序輕鬆傳輸和應用。

2. 代碼介紹

2.1 更新 Rootfs 中的文件/用戶應用程式

```
/* sw-description for updating files/upser applicaitons */

/*denotes the beginning of the software section of the sw-description file*/
software =
{
    /*specifies the version number of the software to be updated*/
    version = "0.1.0";
    /*specifies the hardware versions that are compatible with this software version*/
    hardware-compatibility = [ "1.0", "1.2", "1.3" ];

    files: (
        /* single copy mode */
        {
            /*the name of the file/user application to be updated*/
            filename = "target.txt";
            /*specifies the path where the file should be installed on the target system*/
            path = "/home/target.txt"
        },
    );
};
}
```

2.2 更新內核和設備樹

2.2.1 NAND

```
/* sw-description for updating Kernel and device tree in NAND */

/* Denotes the beginning of the software section of the sw-description file */
software =
{
    /* Specifies the version number of the software to be updated */
    version = "0.1.0";
    /* Provides a brief description of the software update */
    description = "Linux kernel and dtb update for MA35 NAND";
    /* Specifies the hardware versions that are compatible with this software version */
    hardware-compatibility = [ "1.0", "1.2", "1.3" ];
    /* Begins the block for specifying the firmware update images */
    images: (
        {
            /* Specifies the version of the image. */
            version = "5.10";
            /* Provides a name or description for the image */
            name = "nand-kernel device tree";
            /* Specifies the filename of the image */
            filename = "Image.dtb";
            /* Specifies the device or storage location where the image will be
updated */
            device = "/dev/mtdblock2";
            /* Defines the type of the image file */
            type = "raw";
        },
        {
            /* Specifies the version of the image. */
            version = "5.10";
            /* Provides a name or description for the image */

```



```

        name = "nand-kernel";
        /* Specifies the filename of the image */
        filename = "Image";
        /* Specifies the device or storage location where the image will be updated
*/
        device = "/dev/mtdblock3";
        /* Defines the type of the image file */
        type = "raw";
    }
);
}

```

2.2.2 SPI NAND

```

/* sw-description for updating Kernel and device tree in SPI NAND */
/* Denotes the beginning of the software section of the sw-description file */
software =
{
    /* Specifies the version number of the software to be updated */
    version = "0.1.0";
    /* Provides a brief description of the software update */
    description = "Linux kernel and dtb update for MA35 SPI NAND";
    /* Specifies the hardware versions that are compatible with this software version */
    hardware-compatibility = [ "1.0", "1.2", "1.3" ];
    /* Begins the block for specifying the firmware update images */
    images: (
        {
            /* Specifies the version of the image. */
            version = "5.10";
            /* Provides a name or description for the image */
            name = "spinand-kernel device tree";
            /* Specifies the filename of the image */
            filename = "Image.dtb";
            /* Specifies the device or storage location where the image will be updated
*/
            device = "/dev/mtdblock7";
            /* Defines the type of the image file */
            type = "raw";
        },
        {
            /* Specifies the version of the image. */
            version = "5.10";
            /* Provides a name or description for the image */
            name = "spinand-kernel";
            /* Specifies the filename of the image */
            filename = "Image";
            /* Specifies the device or storage location where the image will be updated
*/
            device = "/dev/mtdblock8";
            /* Defines the type of the image file */
            type = "raw";
        }
    );
}

```

2.2.3 eMMC/SDcard

```
/* sw-description for updating Kernel and device tree in eMMC/SDcard */

/* Denotes the beginning of the software section of the sw-description file */
software =
{
    /* Specifies the version number of the software to be updated */
    version = "0.1.0";
    /* Provides a brief description of the software update */
    description = "Linux kernel and dtb update for MA35 eMMC/SDcard";
    /* Specifies the hardware versions that are compatible with this software version */
    hardware-compatibility = [ "1.0", "1.2", "1.3" ];
    /* Begins the block for specifying the firmware update images */
    images: (
        {
            /* Specifies the filename of the image */
            filename = "Image.dtb";
            /* Specifies the device or storage location where the image will be updated */
            device = "/dev/mmcblk0";
            /* Specifies the offset at which the image will be written in the storage */
            offset = "0x2c0000";
        },
        {
            /* Specifies the filename of the image */
            filename = "Image";
            /* Specifies the device or storage location where the image will be updated */
            device = "/dev/mmcblk0";
            /* Specifies the offset at which the image will be written in the storage */
            offset = "0x300000"
        }
    );
}
```

2.3 韌體打包腳本

該腳本從指定的文件夾中收集文件並將它們打包成一個單獨的文件。

```
# generate_swu.sh
# Indicating that the script should be executed using the shell interpreter specified
#!/bin/sh
# Assigns the value "1.0" to the variable CONTAINER_VER, which represents the version of
the container being created
CONTAINER_VER="1.0"
# Assigns the value "pack" to the variable PRODUCT_NAME, which represents the name of the
product or container being created
PRODUCT_NAME="pack"
# 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"
# Starts a loop that iterates over each element in the FILES variable
# | cpio -ov -H crc: Pipes the output of the loop to the cpio command, which creates a
cpio archive. The -ov options specify that the output should be in verbose mode and
overwrite existing files. The -H crc option specifies the header format to use, in this
case, CRC.
for i in $FILES;do
    echo $i;done | cpio -ov -H crc > ${PRODUCT_NAME}_${CONTAINER_VER}.swu
```

3. 軟體與硬體需求

3.1 軟體需求

- BSP 版本
 - Linux-5.10.x

3.2 硬體需求

- 電路元件
 - NuMaker-IoT-MA35D1

4. 目錄資訊

📁 EC_MA35D1_OTA_Update_by_Linux_Stage_V1.00

📁 SampleCode generate_swu.sh

📁 eMMC-SDcard sw-description

📁 NAND sw-description

📁 SPINAND sw-description

圖 4-1 目錄資訊

5. 範例程式執行

於主機端打包映像:

要在主機端進行映像打包，需要將 `sw-description`、`Image`、`Image.dtb` 和 `generate.sh` 文件放在同一個文件夾中。這樣可以確保在 PC 主機上的打包過程中所有所需的文件都可以訪問到:

```
@ubuntu:~$ ls -l
total 10044
-rwxrwx-rw- 1 196 May  9 14:18 generate_swu.sh
-rwxrwx-rw- 1 10232320 Oct 31 2022 Image
-rwxrwx-rw- 1 37525 Oct 31 2022 Image.dtb
-rwxrwx-rw- 1 607 Oct 31 2022 sw-description
```

圖 5-1 映像檔的資料夾內容(1)

執行 `generate_swu.sh` 腳本後，最終的映像套件將在當前資料夾建立。該腳本負責執行必要的步驟以生成套件，這可能包括文件操作、壓縮或其他操作。當腳本成功執行後，將在當前目錄中找到生成的映像套件。

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

```
@ubuntu:~$ ./generate_swu.sh
sw-description
Image
Image.dtb
20061 blocks
@ubuntu:~$ ls -l
total 20076
-rwxrwx-rw- 1 196 May  9 14:18 generate_swu.sh
-rwxrwx-rw- 1 10232320 Oct 31 2022 Image
-rwxrwx-rw- 1 37525 Oct 31 2022 Image.dtb
-rw-rw-r-- 1 10271232 May  9 14:22 Linux_kernel_dtb_update_1.0.swu
-rwxrwx-rw- 1 607 Oct 31 2022 sw-description
```

圖 5-2 映像檔的資料夾內容(2)

在 MA35 裝置上，可以通過配置設備使用 DHCP (動態主機配置協議) 從路由器獲取 IP 地址。

```
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
```

系統初始化之後，swupdate 進程會在後台運行。要列出正在運行的 swupdate 進程，可以使用 ps 命令，該命令會顯示系統上活動進程的相關信息。以下是列出 swupdate 進程的示例命令：

```
MA35D1# ps aux | grep swupdate
212 root    /usr/bin/swupdate -v -w -r /var/www/swupdate -p 8080
219 root    /usr/bin/swupdate -v -w -r /var/www/swupdate -p 8080
258 root    grep swupdate
```

要從電腦訪問在 MA35D1 設備上運行的 swupdate 網頁伺服器，可以打開一個網頁瀏覽器，然後輸入 MA35D1 設備的 IP 地址，後面加上 ":8080"。以下是具體步驟：

1. 確保電腦和 MA35D1 設備連接到同一個網路。
2. 在電腦上打開一個網頁瀏覽器（如 Edge、Chrome、Firefox 等）。
3. 在網頁瀏覽器的地址欄中，輸入 MA35D1 設備的 IP 地址，後面加上 ":8080"。例如，如果 MA35D1 設備的 IP 地址是 192.168.0.175，則輸入 "192.168.0.175:8080"。
4. 網頁瀏覽器將與在 MA35D1 設備上運行的 swupdate 網頁伺服器建立連接。

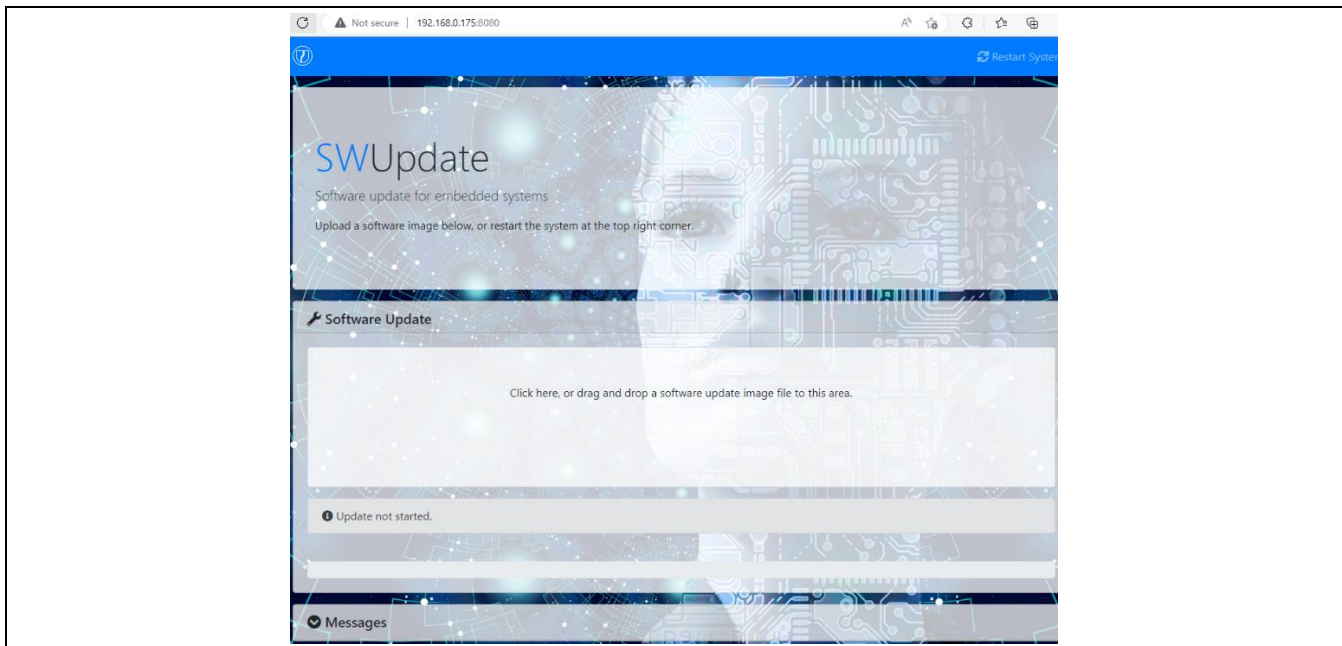


圖 5-3 swupdate 網頁端

訪問 swupdate 網頁伺服器後可執行一系列操作，例如上傳 pack.swu 韌體套件進行更新、啟動更新流程以及監控更新的進度或狀態。

要將 pack.swu 韌體套件傳輸到 MA35D1 設備上，只需在瀏覽器中的 swupdate 網頁上將

pack.swu 檔案拖放即可。

以下為操作實例畫面：

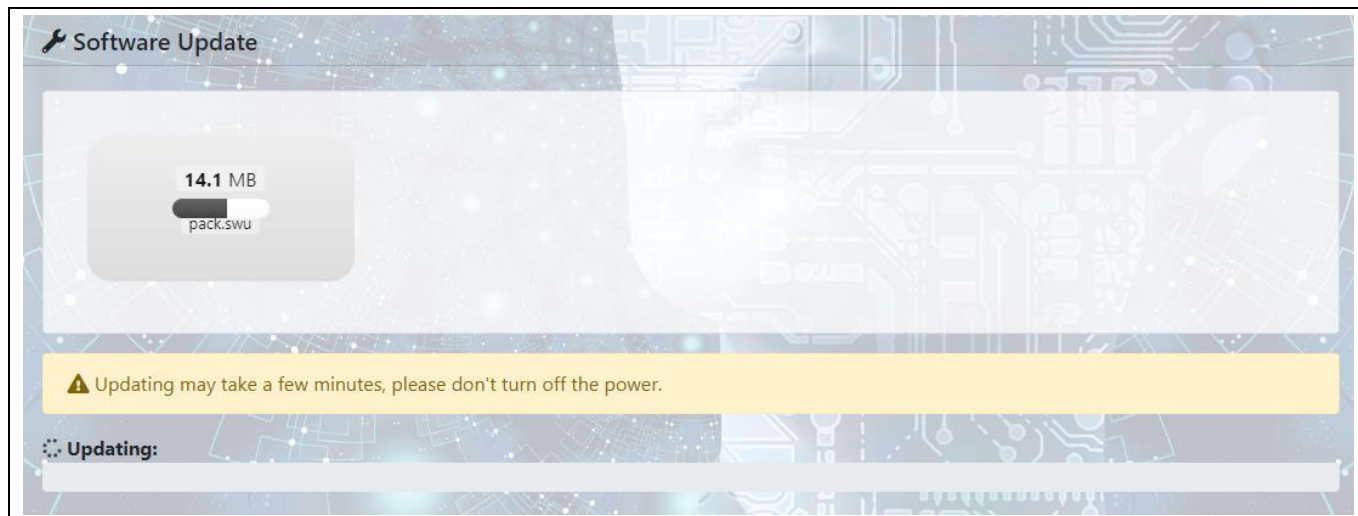


圖 5-4 網頁端更新中



圖 5-5 網頁端更新完成

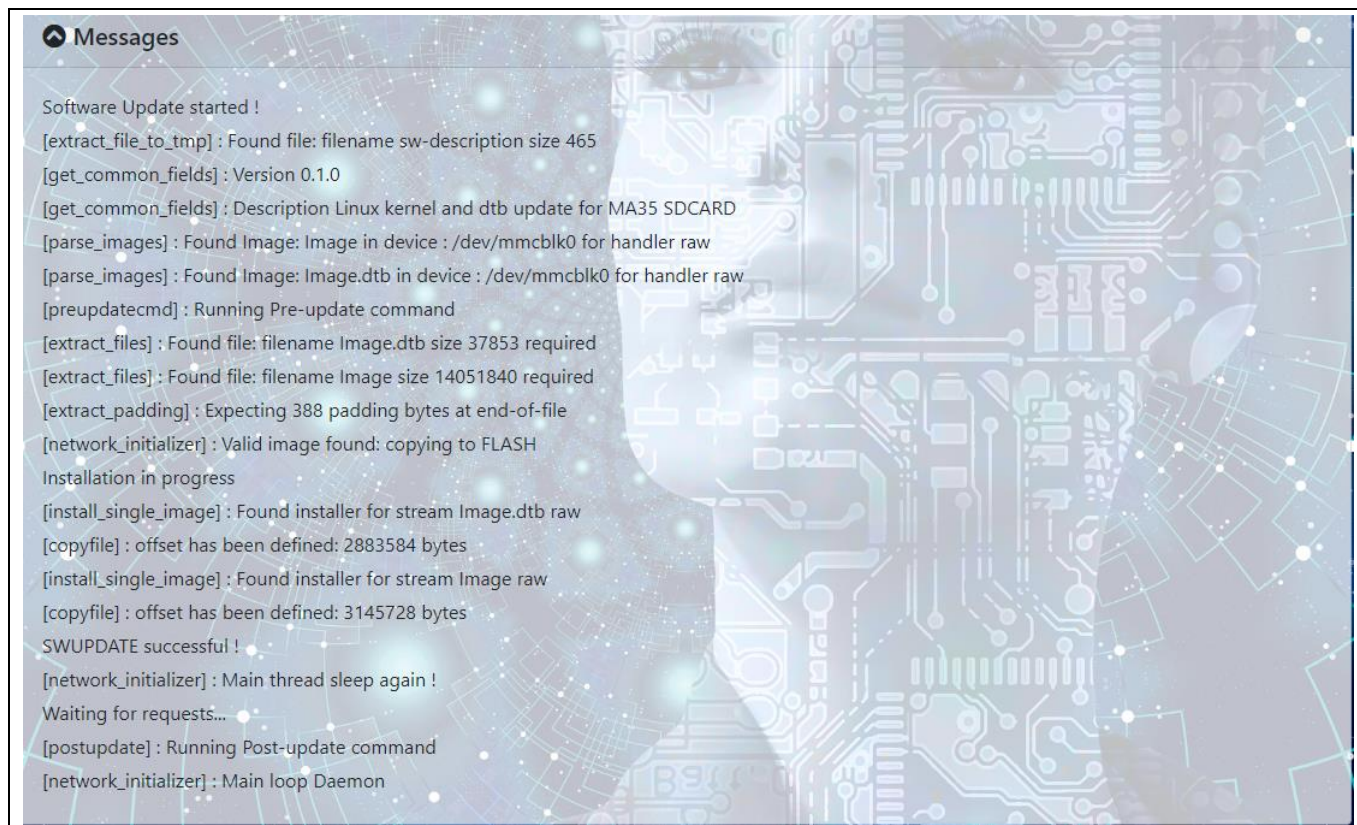


圖 5-6 網頁端更新過程紀錄

6. 修訂紀錄

Date	Revision	Description
2023.05.09	1.00	初始發布。

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.*