NuMicro® Family

Arm® Cortex®-A35-based Microprocessor

MA35D1 NAND Flash Modification Guide

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

Table of Contents

1 OVERVIEW 3

2 NAND FLASH RELATIVE FILES 4

2.1 NuWriter 4

2.2 TF-A 6

2.3 U-Boot 7

2.4 Linux 11

2.5 Rootfs 12

2.6 Build Flow 14

3 REVISION HISTORY 15

1 OVERVIEW

The MA35D1 NAND Flash Interface supports 2K bytes, 4 Kbytes and 8 Kbytes page size. It only supports ECC8, ECC12, and ECC24. Users can select different size NAND Flash for their emulation board. All images in NAND Flash should block align.

The MA35D1 NuMaker board uses Winbond 128MB NAND Flash. This is a ECC8 2KB page size NAND Flash. The MA35D1 BSP NAND default setting is 2KB page size. Users can select suitable NAND Flash for their emulation board and modify NAND setting by reference to this document.

This document describes how to build a pack image for specific NAND Flash. The image includes TF-A, U-Boot and Linux. Using MA35D1 NuWriter can program the pack image to NAND. Changing the power-on-setting to NAND boot can boot from NAND Flash and finally into Linux shell.

2 NAND FLASH RELATIVE FILES

There are four NAND Flash related groups should be modified – NuWriter, TF-A, U-Boot and Linux. All images should be aligned with NAND Flash block. NAND Flash block 0 ~ block 3 are used by the header that is proprietarily defined by Nuvoton. Other images should start from block 4. The header includes information of images and is used by MA35D1 IBR to load the first executable image.

2.1 NuWriter

Two json files need to be modified. They are header-nand.json and pack-nand.json. For more details, please refer to *MA35D1 NuWriter User Manual*.

In buildroot: the file path is

buildroot/board/nuvoton/ma35d1/nuwriter/header-nand.json

buildroot/board/nuvoton/ma35d1/nuwriter/pack-nand.json

In yocto: the file path is

build/tmp-glibc/deploy/images/ma35d1-iot/nuwriter/header-nand.json

build/tmp-glibc/deploy/images/ma35d1-iot/nuwriter/pack-nand.json

- header-nand.json –

  For a 128 MB NAND Flash with page size 2 KB, ECC8, 64 pages per block, and block size is 128 KB (0x20000). The NAND memory map should be as follows.

```
"image":
[
    {
      "offset": "0xC0000",  ← block 6
      "loadaddr": "0x28023000",
      "type": "3",
      "file": "bl2-ma35d1.dtb"
    },
    {
      "offset": "0xE0000",  ← block 7
      "loadaddr": "0x28000000",
      "type": "4",
      "file": "bl2-ma35d1.bin"
    }
]
```

  For a 32 GB NAND Flash with page size 8 KB, ECC8, 128 pages per block, and block size is 1 MB (0x100000). The NAND memory map should be as follows.

```
"image":
[
    {
      "offset": "0x600000",  ← block 6
      "loadaddr": "0x28023000",
      "type": "3",
      "file": "bl2.dtb"
    },
    {
      "offset": "0x700000",  ← block 7
      "loadaddr": "0x28000000",
      "type": "4",
      "file": "bl2.bin"
    }
]
```

- pack-nand.json –

  For a 128 KB NAND Flash with 2 KB page size, ECC8, 64 pages per block, and block size is 128 KB (0x20000). The NAND memory map should be as follows.

```
        {
          "offset": "0xC0000",  ← block 6
          "file": "bl2-ma35d1.dtb",
          "type": 0
        },
        {
          "offset": "0xE0000",  ← block 7
          "file": "bl2-ma35d1.bin",
          "type": 0
        },
        {
          "offset": "0x100000",  ← block 8
          "file": "fip.bin-nand",
          "type": 1
        },
        {
          "offset": "0x300000",  ← block 24
          "file": "u-boot-initial-env.bin-nand",
          "type": 1
        },
        {
          "offset": "0x3C0000",  ← block 30
          "file": "Image.dtb",
          "type": 1
        },
        {
          "offset": "0x400000",  ← block 32
          "file": "Image",
          "type": 1
        },
        {
          "offset": "0x1C00000",  ← block 224
          "file": "rootfs.ubi",
          "type": 1
        }
```

  For a 32 GB NAND Flash with 8 KB page size, ECC8, 128 pages per block, and block size is 1 MB (0x100000). The NAND memory map should be as follows.

```
        {
          "offset": "0x600000",  ← block 6
          "file": "bl2.dtb",
          "type": 128
        },
        {
          "offset": "0x700000",  ← block 7
          "file": "bl2.bin",
          "type": 128
        },
        {
          "offset": "0x800000",  ← block 8
          "file": "fip.bin-nand",
          "type": 1
        },
        {
          "offset": "0xA00000",  ← block 10
          "file": "uboot-env.bin-nand",
          "type": 1
        },
```

```
        {
          "offset": "0xB00000",  ← block 11
          "file": "Image.dtb",
          "type": 1
        },
        {
          "offset": "0xC00000",  ← block 12
          "file": "Image",
          "type": 1
        },
        {
          "offset": "0x1C00000",  ← block 28
          "file": "rootfs.ubi",
          "type": 1
        }
```

2.2 TF-A

Device tree of TF-A should be modified.

In buildroot: the file path is

buildroot/output/build/arm-trusted-firmware-custom/fdts/ma35d1.dtsi

In yocto: the path is

build/tmp-glibc/work/ma35d1_iot-poky-linux/tf-a-ma35d1/2.3.r1-r0/git/fdts/ma35d1.dtsi

Use 'bitbake tfa-a-ma35d1 –c devshell' can get the path.

- The following content defines a 128 MB NAND Flash with page size 2 KB, ECC8, oob 64 bytes, 64 pages per block, and total 1024 blocks. The FIP image offset should be 0x100000.

```
/* nand information */
nand-ecc-strength = <8>;
nand-ecc-step-size = <512>;
nand-page-size = <2048>;
nand-oob-size = <64>;
nand-page-count = <64>;
nand-block-count = <1024>;
nand-image-offset = <0x100000>;        // fip image offset
```

- The following content defines a 4 GB NAND Flash with page size 8 KB, ECC8, oob 448 bytes, 128 pages per block, and total 4096 blocks. The FIP image offset should be 0x800000.

```
/* nand information */
nand-ecc-strength = <8>;
nand-ecc-step-size = <512>;
nand-page-size = <8192>;
nand-oob-size = <448>;
nand-page-count = <128>;
nand-block-count = <4096>;
nand-image-offset = <0x800000>;
```

2.3 U-Boot

NAND defconfig needs to be modified. Yocto modify ma35d1_nand_defconfig directly. But buildroot need set menuconfig.

In yocto: the path is

build/tmp-glibc/work/ma35d1_iot-poky-linux/u-boot-ma35d1/2020.07-r0/git/configs/ma35d1_nand_defconfig

Use 'bitbake u-boot-ma35d1 –c devshell' can get the path.

- For a 128 MB NAND Flash with 2 KB page size and 128 KB block size. The setting can be as follows.

```
…
       CONFIG_ENV_OFFSET=0x300000
…
       CONFIG_MTDPARTS_DEFAULT="mtdparts=nand0:3m(uboot),768k(uboot-
env),256k(device-tree),24m(kernel),100m(rootfs)"
```

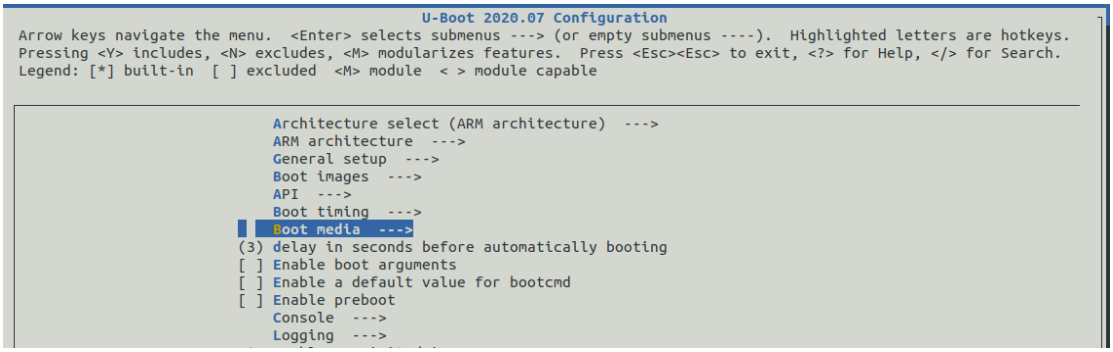
- For a 32 GB NAND Flash with 8 KB page size and 1 MB block size. The setting can be as follows.

```
…
       CONFIG_ENV_OFFSET=0xA00000
…
       CONFIG_MTDPARTS_DEFAULT="mtdparts=nand0:10m(uboot),1m(uboot-env),1m(device-
tree),16m(kernel),100m(rootfs)"
```

In buildroot should set the menuconfig, as follows.

```
$ make uboot-menuconfig
```

Set boot from NAND Flash. Select 'Boot media'.



Select 'Support for booting from NAND flash'.

```
                                    Boot media
Arrow keys navigate the menu.  <Enter> selects submenus ---> (or empty submenus ----).  Highlighted letters are hotkeys.
Pressing <Y> includes, <N> excludes, <M> modularizes features.  Press <Esc><Esc> to exit, <?> for Help, </> for Search.
Legend: [*] built-in  [ ] excluded  <M> module  < > module capable

                        [*] Support for booting from NAND flash
                        [ ] Support for booting from ONENAND
                        [ ] Support for booting from QSPI flash
                        [ ] Support for booting from SATA
                        [ ] Support for booting from SD/EMMC
                        [ ] Support for booting from SPI flash
```

Set the U-Boot environment. Select ‘Environment’.

```
                                 U-Boot 2020.07 Configuration
Arrow keys navigate the menu.  <Enter> selects submenus ---> (or empty submenus ----).  Highlighted letters are hotkeys.
Pressing <Y> includes, <N> excludes, <M> modularizes features.  Press <Esc><Esc> to exit, <?> for Help, </> for Search.
Legend: [*] built-in  [ ] excluded  <M> module  < > module capable

                        ^(-)
                            Boot media  --->
                        (3) delay in seconds before automatically booting
                        [ ] Enable boot arguments
                        [ ] Enable a default value for bootcmd
                        [ ] Enable preboot
                            Console  --->
                            Logging  --->
                        -*- Enable raw initrd images
                        ()  Default fdt file
                        [ ] Execute Misc Init
                        [ ] add U-Boot environment variable vers
                        [*] Execute Board late init
                        [*] Display information about the CPU during start up
                        [*] Display information about the board during early start up
                        [ ] Display information about the board during late start up
                        [ ] Include bounce buffer API
                        [ ] Call get_board_type() to get and display the board type
                            Start-up hooks  --->
                            Security support  ----
                            Update support  --->
                            Blob list  --->
                            SPL / TPL  --->
                            Command line interface  --->
                            Partition Types  --->
                            Device Tree Control  --->
                        (dtc) Path to dtc binary for use within mkimage
                            Environment  --->
                        [*] Networking support  --->
                            Device Drivers  --->
                            File systems  --->
                            Library routines  --->
                        [ ] Unit tests  ----
```

Set the environment offset. For a 2 KB page size example, offset is 0x300000. The environment size is fixed at 64KB (Do ***NOT*** modify).

```
                                    Environment
Arrow keys navigate the menu.  <Enter> selects submenus ---> (or empty submenus ----).  Highlighted letters are hotkeys.
Pressing <Y> includes, <N> excludes, <M> modularizes features.  Press <Esc><Esc> to exit, <?> for Help, </> for Search.
Legend: [*] built-in  [ ] excluded  <M> module  < > module capable

                        [ ] Environment is not stored
                        [ ] Environment in EEPROM
                        [ ] Environment is in a FAT filesystem
                        [ ] Environment is in a EXT4 filesystem
                        [ ] Environment in flash memory
                        [ ] Environment in an MMC device
                        [*] Environment in a NAND device
                        [ ] Environment in a non-volatile RAM
                        [ ] Environment is in OneNAND
                        [ ] Environment is in remote memory space
                        [ ] Environment is in SPI flash
                        [ ] Environment in a UBI volume
                        [ ] Enable redundant environment support
                        (0x300000) Environment offset
                        (0x10000) Environment Size
                        [ ] Relocate gd->env_addr
                        [ ] Create default environment from file
                        [ ] Add run-time information to the environment
                        [ ] Block forced environment operations
```

Set the environment offset. For 8 KB page size example, offset is 0xA00000. The environment size is fixed at 64KB (Do ***NOT*** modify).

```
                                   Environment
Arrow keys navigate the menu.  <Enter> selects submenus ---> (or empty submenus ----).  Highlighted letters are hotkeys.
Pressing <Y> includes, <N> excludes, <M> modularizes features.  Press <Esc><Esc> to exit, <?> for Help, </> for Search.
Legend: [*] built-in  [ ] excluded  <M> module  < > module capable

                        [ ] Environment is not stored
                        [ ] Environment in EEPROM
                        [ ] Environment is in a FAT filesystem
                        [ ] Environment is in a EXT4 filesystem
                        [ ] Environment in flash memory
                        [ ] Environment in an MMC device
                        [*] Environment in a NAND device
                        [ ] Environment in a non-volatile RAM
                        [ ] Environment is in OneNAND
                        [ ] Environment is in remote memory space
                        [ ] Environment is in SPI flash
                        [ ] Environment in a UBI volume
                        [ ] Enable redundant environment support
                        (0xA00000) Environment offset
                        (0x10000) Environment Size
                        [ ] Relocate gd->env_addr
                        [ ] Create default environment from file
                        [ ] Add run-time information to the environment
                        [ ] Block forced environment operations
```

Set the MTD partition table. Select ‘Command line interface’.

```
                                U-Boot 2020.07 Configuration
Arrow keys navigate the menu.  <Enter> selects submenus ---> (or empty submenus ----).  Highlighted letters are hotkeys.
Pressing <Y> includes, <N> excludes, <M> modularizes features.  Press <Esc><Esc> to exit, <?> for Help, </> for Search.
Legend: [*] built-in  [ ] excluded  <M> module  < > module capable

                        ^(-)
                            Boot media  --->
                        (3) delay in seconds before automatically booting
                        [ ] Enable boot arguments
                        [ ] Enable a default value for bootcmd
                        [ ] Enable preboot
                            Console  --->
                            Logging  --->
                        -*- Enable raw initrd images
                        ()  Default fdt file
                        [ ] Execute Misc Init
                        [ ] add U-Boot environment variable vers
                        [*] Execute Board late init
                        [*] Display information about the CPU during start up
                        [*] Display information about the board during early start up
                        [ ] Display information about the board during late start up
                        [ ] Include bounce buffer API
                        [ ] Call get_board_type() to get and display the board type
                            Start-up hooks  --->
                            Security support  ----
                            Update support  --->
                            Blob list  --->
                            SPL / TPL  --->
                            Command line interface  --->
                            Partition Types  --->
                            Device Tree Control  --->
                        (dtc) Path to dtc binary for use within mkimage
                            Environment  --->
                        [*] Networking support  --->
                            Device Drivers  --->
                            File systems  --->
                            Library routines  --->
                        [ ] Unit tests  ----
```

Select ‘Filesystem commands’.

```
                                   Command line interface
Arrow keys navigate the menu.  <Enter> selects submenus ---> (or empty submenus ----).  Highlighted letters are hotkeys.
Pressing <Y> includes, <N> excludes, <M> modularizes features.  Press <Esc><Esc> to exit, <?> for Help, </> for Search.
Legend: [*] built-in  [ ] excluded  <M> module  < > module capable

                        [*] Support U-Boot commands
                        -*-   Use hush shell
                        -*-   Enable command line editing
                        -*-   Enable auto complete using TAB
                        -*-   Enable long help messages
                        (MA35D1> ) Shell prompt
                        (y) Command execution tracer
                            Autoboot options  --->
                            *** Commands ***
                            Info commands  --->
                            Boot commands  --->
                            Environment commands  --->
                            Memory commands  --->
                            Compression commands  --->
                            Device access commands  --->
                            Shell scripting commands  --->
                            Android support commands  ----
                        [*] Network commands  --->
                        [ ] ethsw
                        -*- pxe
                        [ ] wol
                            Misc commands  --->
                            TI specific command line interface  --->
                            Power commands  ----
                            Security commands  --->
                            Firmware commands  ----
                            Filesystem commands  --->
                            Debug commands  --->
                        [*] Enable UBI - Unsorted block images commands
                        [ ]   Enable rename
                        [*]   Enable UBIFS - Unsorted block images filesystem commands
```

For 2KB page size example.

```
                                   Filesystem commands
Arrow keys navigate the menu.  <Enter> selects submenus ---> (or empty submenus ----).  Highlighted letters are hotkeys.
Pressing <Y> includes, <N> excludes, <M> modularizes features.  Press <Esc><Esc> to exit, <?> for Help, </> for Search.
Legend: [*] built-in  [ ] excluded  <M> module  < > module capable

                        [ ] Enable the 'btrsubvol' command
                        -*- ext2 command support
                        -*- ext4 command support
                        [*]   ext4 write command support
                        -*- FAT command support
                        -*- filesystem commands
                        [ ] fsuuid command
                        [ ] jffs2 command
                        [*] MTD partition support
                        [ ]   Padd partition size to take account of bad blocks
                        [ ]   Show net size (w/o bad blocks) of partitions
                        (nand0=nand0) Default MTD IDs
                        (mtdparts=nand0:3m(uboot),768k(uboot-env),256k(device-tree),24m(kernel),100m(rootfs)) Default MT
                        [ ] reiser - Access to reiserfs filesystems
                        [ ] zfs - Access of ZFS filesystem
```

For 8 KB page size example.

```
                                   Filesystem commands
Arrow keys navigate the menu.  <Enter> selects submenus ---> (or empty submenus ----).  Highlighted letters are hotkeys.
Pressing <Y> includes, <N> excludes, <M> modularizes features.  Press <Esc><Esc> to exit, <?> for Help, </> for Search.
Legend: [*] built-in  [ ] excluded  <M> module  < > module capable

                        [ ] Enable the 'btrsubvol' command
                        -*- ext2 command support
                        -*- ext4 command support
                        [*]   ext4 write command support
                        -*- FAT command support
                        -*- filesystem commands
                        [ ] fsuuid command
                        [ ] jffs2 command
                        [*] MTD partition support
                        [ ]   Padd partition size to take account of bad blocks
                        [ ]   Show net size (w/o bad blocks) of partitions
                        (nand0=nand0) Default MTD IDs
                        (mtdparts=nand0:10m(uboot),1m(uboot-env),1m(device-tree),16m(kernel),100m(rootfs)) Default MTD p
                        [ ] reiser - Access to reiserfs filesystems
                        [ ] zfs - Access of ZFS filesystem
```

2.4 Linux

Device tree needs to be modified.

In buildroot: the file path is

buildroot/output/build/linux-custom/arch/arm64/boot/dts/nuvoton/ma35d1.dtsi

In yocto: the path is

build/tmp-glibc/work-shared/ma35d1-iot/kernel-source/arch/arm64/boot/dts/nuvoton/ma35d1.dtsi

Use 'bitbake linux-ma35d1 –c devshell' can get the path.

- NAND Flash is 128 MB. Page size 2 KB, ECC8, 64 pages per block, and 128 KB block size.

```
…
        nand-ecc-strength = <8>;
        nand-ecc-step-size = <512>;
…
…
                uboot@0 {
                        label = "uboot";
                        reg = <0x0000000 0x300000>;
                        read-only;
                };
                uboot-env@300000 {
                        lable = "uboot-env";
                        reg = <0x300000 0xC0000>;
                };
                linux-device-tree@3C0000 {
                        lable = "linux-device-tree";
                        reg = <0x3C0000 0x40000>;
                };
                linux@400000 {
                        label = "linux";
                        reg = <0x400000 0x1800000>;
                };
                rootfs@1C00000 {
                        label = "rootfs";
                        reg = <0x1C00000 0x6400000>;
                };
```

- NAND Flash is 32 GB. Page size is 8 KB, ECC8, 128 pages per block, and 1 MB block size.

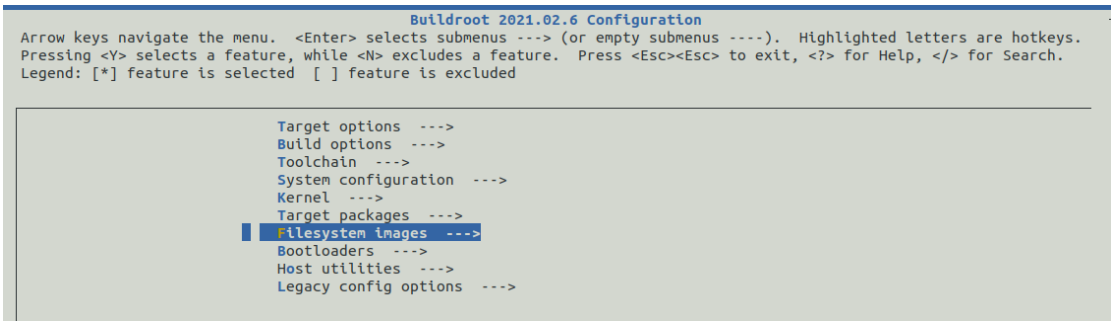
```
…
        nand-ecc-strength = <8>;
        nand-ecc-step-size = <512>;
```

```
…
…
                uboot@0 {
                        label = "uboot";
                        reg = <0x0000000 0xA00000>;
                        read-only;
                };
                uboot-env@A00000 {
                        lable = "uboot-env";
                        reg = <0xA00000 0x100000>;
                };
                device-tree@B00000 {
                        lable = "device-tree";
                        reg = <0xB00000 0x100000>;
                };
                kernel@C00000 {
                        label = "kernel";
                        reg = <0xC00000 0x1000000>;
                };
                rootfs@1c00000 {
                        label = "rootfs";
                        reg = <0x1C00000 0x6400000>;
                };
```

2.5 Rootfs

The MA35D1 uses UBIFS NAND file system.

In buildroot: Use menuconfig to select the correct setting. Enter ‘make menuconfig’. Select ‘Filesystem images’.



For 2 KB page size – page size 2 KB, 64 pages per block.

- Physical eraseblock size – 2048 * 64 = 0x20000
- Logical eraseblock size – this should be (physical eraseblock size – 2 * page size). Therefore, this value is 0x20000 – 2 * 2048 = 0x1f000.

- Minimum I/O unit size – this means page size.
- Maximum logical eraseblock count – block count for UBIFS.

```
                                    Filesystem images
Arrow keys navigate the menu.  <Enter> selects submenus ---> (or empty submenus ----).  Highlighted letters are hotkeys.  Pressing <Y> selects a feature, while <N> excludes a feature.  Press
<Esc><Esc> to exit, <?> for Help, </> for Search.  Legend: [*] feature is selected  [ ] feature is excluded

                                    [ ] exfs root filesystem
                                    [ ] btrfs root filesystem
                                    [ ] cloop root filesystem for the target device
                                    [ ] cpio the root filesystem (for use as an initial RAM filesystem)
                                    [ ] cramfs root filesystem
                                    [ ] erofs root filesystem
                                    [*] ext2/3/4 root filesystem
                                          ext2/3/4 variant (ext4)  --->
                                    (rootfs) filesystem label
                                    (240M) exact size
                                    (0)   exact number of inodes (leave at 0 for auto calculation)
                                    (5)   reserved blocks percentage
                                    (-O ^64bit) additional mke2fs options
                                          Compression method (no compression)  --->
                                    [ ] f2fs root filesystem
                                    [ ] initial RAM filesystem linked into linux kernel
                                    [ ] jffs2 root filesystem
                                    [ ] romfs root filesystem
                                    [ ] squashfs root filesystem
                                    [*] tar the root filesystem
                                          Compression method (no compression)  --->
                                    ()    other random options to pass to tar
                                    [*] ubi image containing an ubifs root filesystem
                                    (0x20000) physical eraseblock size
                                    (512) sub-page size
                                    [ ]   Use custom config file
                                    ()    Additional ubinize options
                                    -*- ubifs root filesystem
                                    (0x1f000) logical eraseblock size
                                    (0x800) minimum I/O unit size
                                    (800) maximum logical eraseblock count
                                          ubifs runtime compression (lzo)  --->
                                          Compression method (no compression)  --->
                                    ()    Additional mkfs.ubifs options
                                    [ ] yaffs2 root filesystem
```

For 8 KB page size – page size 8 KB, 128 pages per block.

- Physical eraseblock size – 8192 * 128 = 0x100000
- Logical eraseblock size – this should be (physical eraseblock size – 2 * page size). Therefore, this value is 0x100000 – 2 * 8192 = 0xfc000.
- Minimum I/O unit size – this means page size.
- Maximum logical eraseblock count – block count for UBIFS.

```
                                    Filesystem images
Arrow keys navigate the menu.  <Enter> selects submenus ---> (or empty submenus ----).  Highlighted letters are hotkeys.
Pressing <Y> selects a feature, while <N> excludes a feature.  Press <Esc><Esc> to exit, <?> for Help, </> for Search.
Legend: [*] feature is selected  [ ] feature is excluded

                                    ↑(-)
                                    [ ] cpio the root filesystem (for use as an initial RAM filesystem)
                                    [ ] cramfs root filesystem
                                    [ ] erofs root filesystem
                                    [*] ext2/3/4 root filesystem
                                          ext2/3/4 variant (ext4)  --->
                                    (rootfs) filesystem label
                                    (240M) exact size
                                    (0)   exact number of inodes (leave at 0 for auto calculation)
                                    (5)   reserved blocks percentage
                                    (-O ^64bit) additional mke2fs options
                                          Compression method (no compression)  --->
                                    [ ] f2fs root filesystem
                                    [ ] initial RAM filesystem linked into linux kernel
                                    [ ] jffs2 root filesystem
                                    [ ] romfs root filesystem
                                    [ ] squashfs root filesystem
                                    [*] tar the root filesystem
                                          Compression method (no compression)  --->
                                    ()    other random options to pass to tar
                                    [*] ubi image containing an ubifs root filesystem
                                    (0x100000) physical eraseblock size
                                    (8192) sub-page size
                                    [ ]   Use custom config file
                                    ()    Additional ubinize options
                                    -*- ubifs root filesystem
                                    (0xfc000) logical eraseblock size
                                    (0x2000) minimum I/O unit size
                                    (100) maximum logical eraseblock count
                                          ubifs runtime compression (lzo)  --->
                                          Compression method (no compression)  --->
                                    (-F)  Additional mkfs.ubifs options
                                    [ ] yaffs2 root filesystem
```

In yocto: Modify ma35d1-base.inc file. The path is

sources/meta-ma35d1/conf/machine/include/ma35d1-base.inc

For 2 KB page size – page size 2 KB, 64 pages per block.

- -p – 2048 * 64 = 0x20000
- -e – this should be (physical eraseblock size – 2 * page size). Therefore, this value is 0x20000 – 2 * 2048 = 0x1f000 (126976).
- -s, -m, -O – this means page size.
- -c – block count for UBIFS.

```
# Use the expected value of the ubifs filesystem's volume name in the kernel
UBI_VOLNAME ?= "rootfs"
# The biggest NANDs on current modules are 100MB.
# This sums up to 2048 LEBs available for the ubifs (-c)
MKUBIFS_ARGS ?= "-F -m 2048 -e 126976 -c 800"
UBINIZE_ARGS ?= "-m 2048 -p 128KiB -s 2048 -O 2048"
```

For 8 KB page size – page size 8 KB, 128 pages per block.

- -p – 8192 * 128 = 0x100000
- -e – this should be (physical eraseblock size – 2 * page size). Therefore, this value is 0x100000 – 2 * 8192 = 0xfc000 (1032192).
- -m, -s – this means page size.
- -c – block count for UBIFS.

```
# Use the expected value of the ubifs filesystem's volume name in the kernel
UBI_VOLNAME ?= "rootfs"
# The biggest NANDs on current modules are 100MB.
# This sums up to 2048 LEBs available for the ubifs (-c)
MKUBIFS_ARGS ?= "-F -m 8192 -e 1032192 -c 100"
UBINIZE_ARGS ?= "-m 8192 -p 1024KiB -s 8192 -O 8192"
```

2.6 Build Flow

After modify TF-A, U-Boot, and Linux. Please rebuild it.

For buildroot:

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

For yocto:

```
$ bitbake tf-a-ma35d1 -C compile
$ bitbake u-boot-ma35d1 -C compile
$ bitbake linux-ma35d1 -C compile
$ bitbake core-image-minimal -c cleanall
$ bitbake core-image-minimal
```

3 REVISION HISTORY

Date	Revision	Description
2023.05.09	1.01	Modify u-boot description. The environment size setting must be fixed at 64KB.
2022.03.12	1.00	Initial version.

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.