

Linuxブート可能 NANDフラッシュ書込み手法

株式会社東陽テクニカ 汎用計測営業部

May 21, 2011 – June 29, 2011

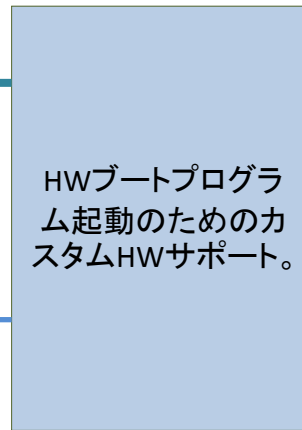
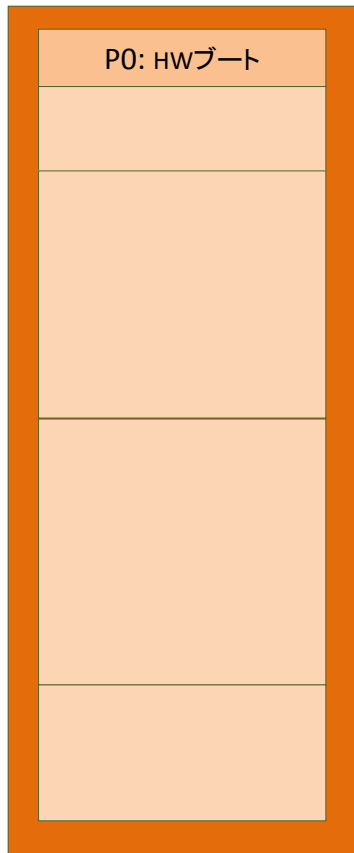
MK ver1.1

NANDからのLinuxブート手順

- ① カスタムHWサポートによる、ブートローダーを起動するためのNAND/DRAMドライバを含むHWブートプログラム
- ② HWブートプログラムにてU-BootのDRAMへのロードと起動
- ③ U-bootによるシステム初期化、Linuxカーネルのロード、カーネルの起動
- ④ カーネルは、ファイルシステム(YAFFS等)の起動を行い、アプリケーションや必要なファイルをロード後、タスクを起動

① HWブート

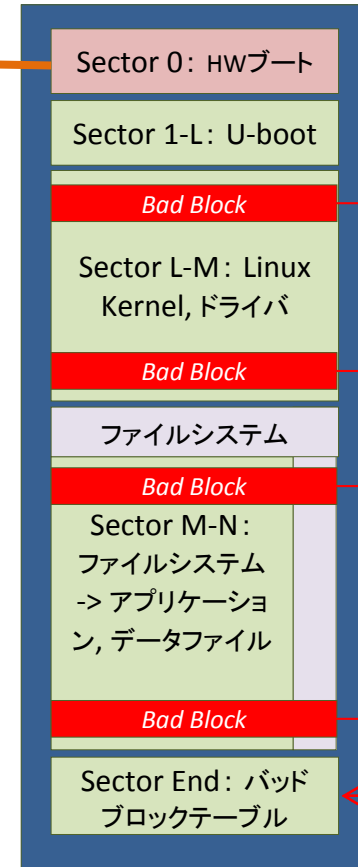
システムメモリエイジー



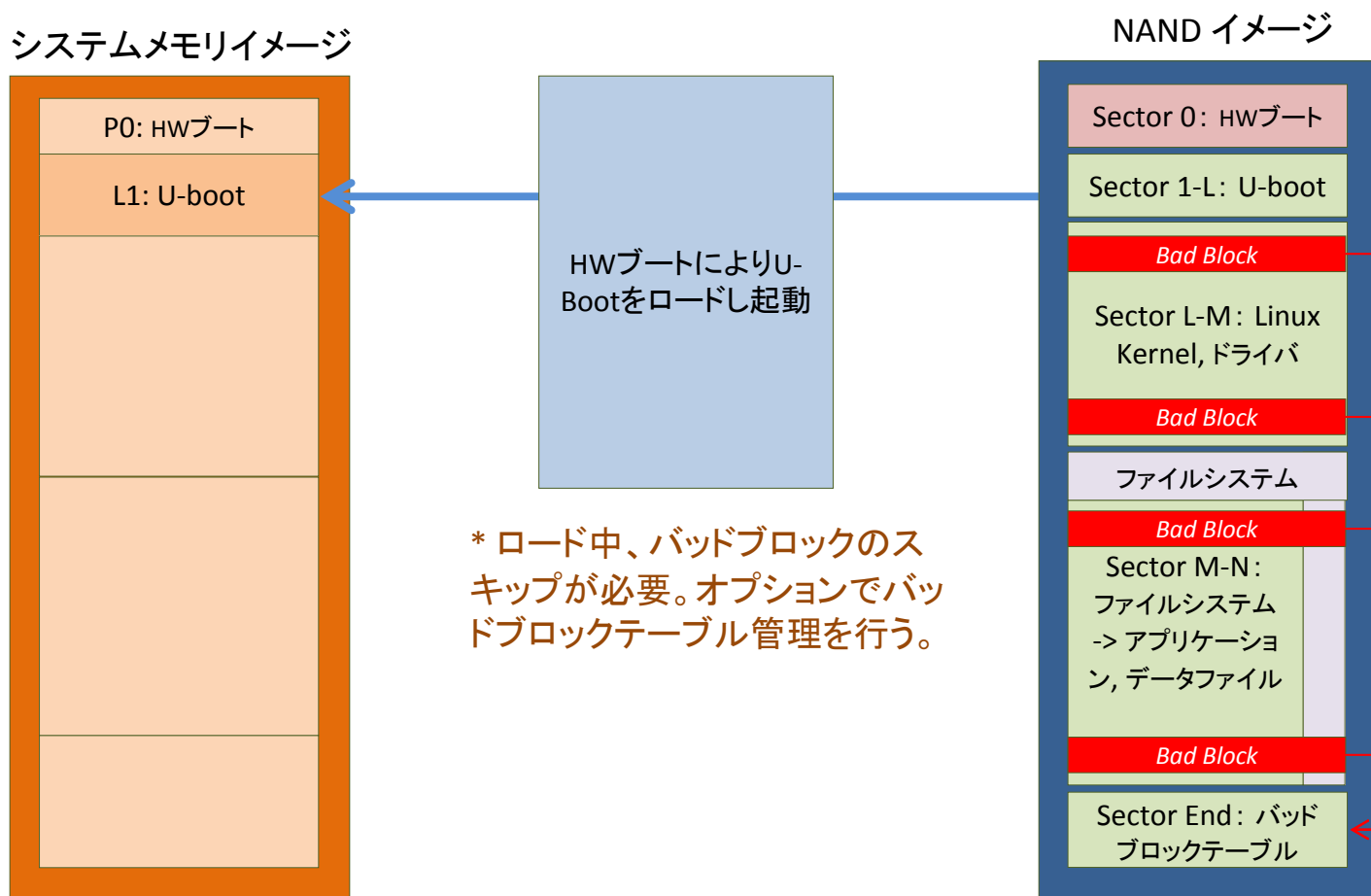
Processor reset

HW reset

NAND イメージ

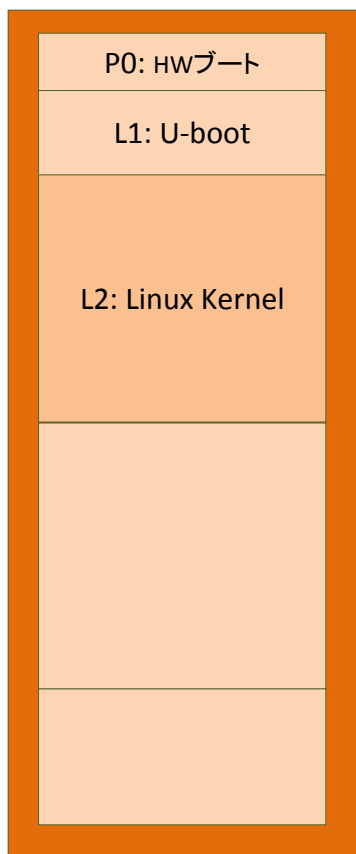


② U-Bootのロード、実行

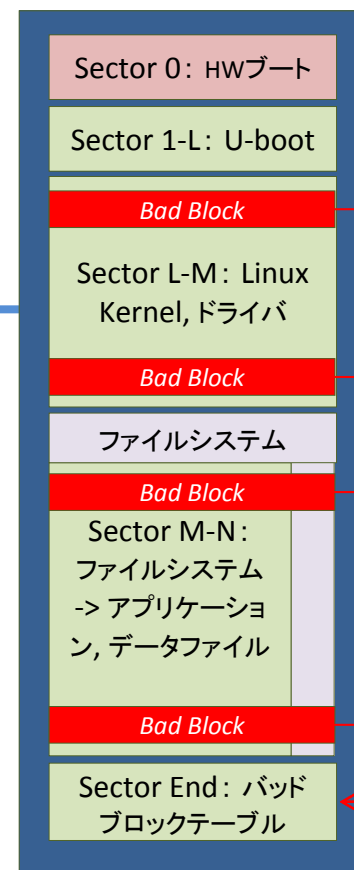


③ カーネルのロードと起動

システムメモリイメージ



NAND イメージ

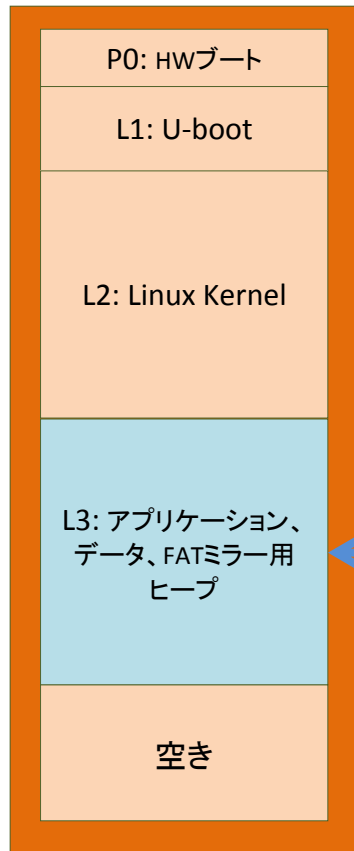


U-bootがLinux Kernelのロードを行った後カーネルに実行を移す

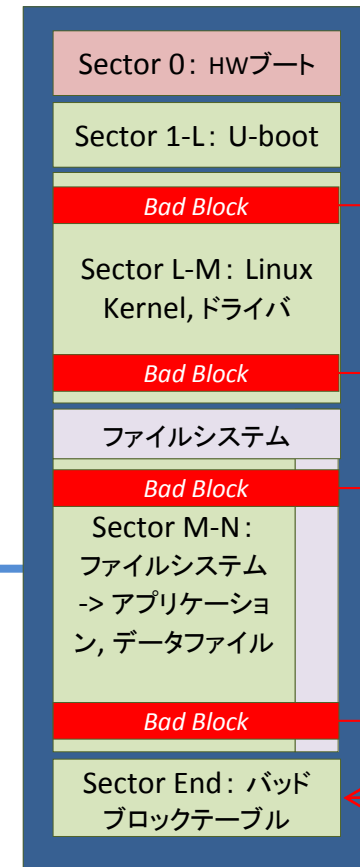
* ロード中、バッドブロックのスキップが必要。オプションでバッドブロックテーブル管理を行う。

④ アプリケーションとデータのロード

システムメモリエイジー



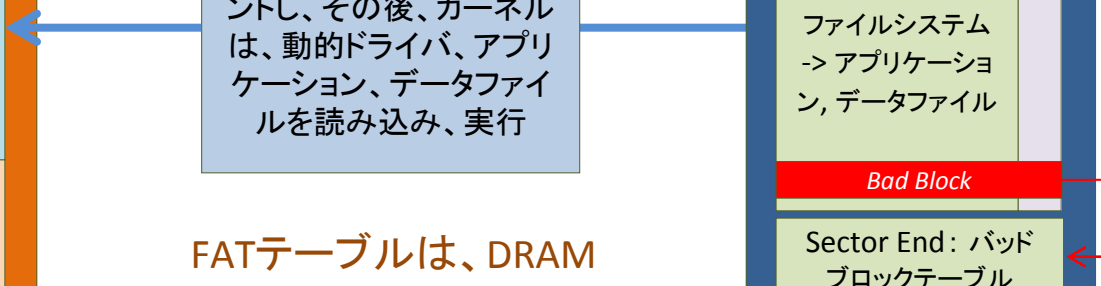
NAND イメージ



* ロード中、バッドブロックのスキップが必要。オプションでバッドブロックテーブル管理を行う。

カーネルは、YAFFS等のファイルシステムをマウントし、その後、カーネルは、動的ドライバ、アプリケーション、データファイルを読み込み、実行

FATテーブルは、DRAMにミラー化される



NANDイメージのビルド - 1

- HWブート:
 - リセットアドレスで実行するHWブートイメージをビルド
 - 次にU-BootがロードできるためのNANDドライバ、DRAMドライバなどをサポート

NANDイメージのビルド - 2

- U-boot:
 - LinuxのブートローダであるU-bootイメージをビルド
 - HWブートは、U-BootをDRAMにロードし、制御をU-Bootに移す

NANDイメージのビルド - 3

- Linux Kernel:
 - Linux kernelイメージのビルド
 - U-Bootは、Linux Kernelをロードし、制御をカーネルに移す

NANDイメージのビルド - 4

- アプリケーション, 動的ドライバ, データファイル:
 - 各ファイルをファイルシステムの管理バウンダリに合わせて重ならないアドレスに配置、管理テーブルデータと結合し一つのバイナリファイルを生成する

How to build NAND image - 5

個別ファイルイメージ

p0: HWブート

Adr p1-pL: Kernel

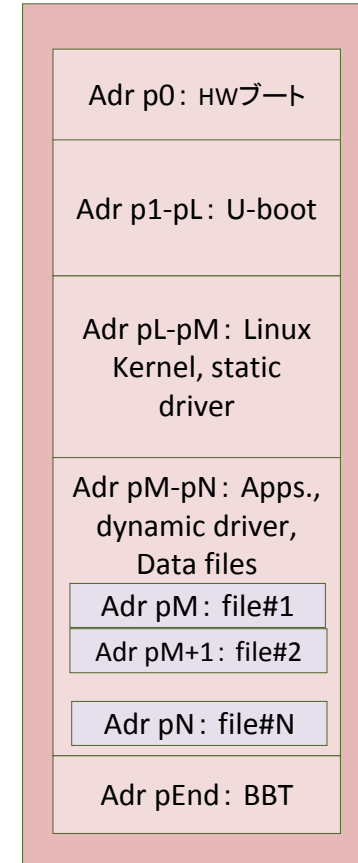
Adr pL-pM: Linux
Kernel, static
driver

Adr pM-pN: Apps.,
dynamic driver,
Data files

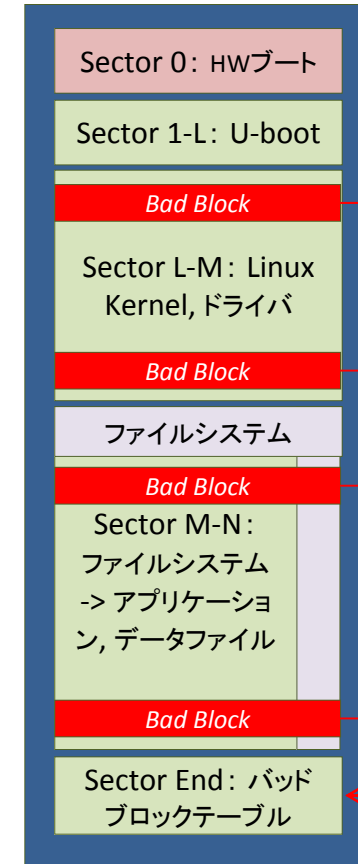
Adr pEnd: バッドブ
ロックテーブル初期値

Data
Mapper
Software

書き込みファイルイメージ



NAND イメージ



書き込み用に一つのバイナリイメージファイルをビルドします。個々の物理アドレスは、重ならない領域にとる必要があります。