



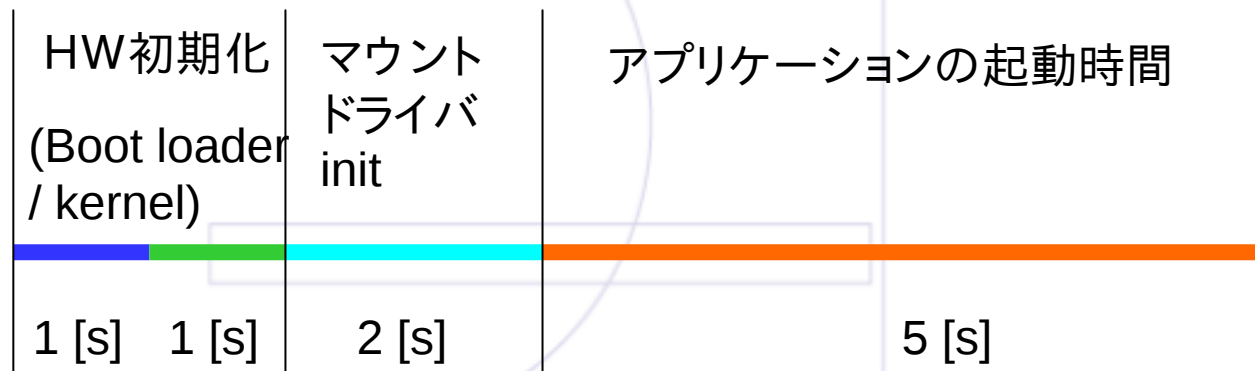
趣旨:

- ハイバネーション技術を用いたシステムの起動高速化の検討
- 開発の各フェーズのスナップショットの保存(検証などの目的に)



目的:

- アプリケーションの起動時間を短縮したい



- アプリケーションが `main()` に到達するまでに行われる処理
 - イメージの転送
 - ダイナミックリンク
 - グローバルコンストラクタ
- システム全体を稼動する為のアプリケーション間の IPC 処理



従来の手法:

- Prelink
 - ダイナミックリンクの処理時間の短縮
- Execute in place (XIP)
 - ストレージから直接コードを実行する。
実行コードのメモリへの展開を短縮



今回の手法:

- ハイバネーションからの復帰を利用する
 - アプリケーションなどシステムの初期化をメモリのロードに置き換えることにより高速起動を実現する



Linux 上のハイバネーション:

- Swsusp
 - 2.6 バニラカーネルに含まれている
(Documentation/power/swsusp.txt)
 - 制作者:Pavel Machek
- Software suspend 2
 - <http://www.suspend2.net/>
 - 制作者:Nigel Cunningham



ARMアーキテクチャへの移植:

- swsusp および suspend2 は主に i386 用
 - ARM へのポーティングが必要
 - CE Linux Forum の public wiki に ARM 用移植パッチがある

<http://tree.celinuxforum.org/CelfPubWiki/SuspendToDiskForARM>



Swsusp (1):

- Linux には3つの状態がサポートされている
 - スタンバイ状態
 - 最小限の省電力、動作状態までの低レイテンシ
⇔ ACPI S1
 - RAMへのサスペンド
 - 十分な省電力、システムは低電力状態、メモリはセルフリフレッシュモード
⇔ ACPI S3
 - ディスクへのサスペンド
 - 最大限の省電力、RAMへのサスペンドに酷似、メモリ内容がディスクに書き込まれる
⇔ ACPI S4



Swsusp (2):

- Swsusp はサスペンド2ディスク
- ディスクではなく、FLASHに書き込むようにした
- レジューム時にFLASHからイメージを読み込むようにした



Swsusp (3):

- Swsusp処理の流れ (suspend時)
 - Freeze processes
 - Free unnecessary memory
 - Suspend devices
 - Make snapshot image (in memory)
 - Resume devices
 - Write snapshot image to swap
 - Powerdown machine

Note: swsusp needs half of total memory



Swsusp (4):

- Swsusp処理の流れ (resume時)
 - Resume begin at `late_initcall`
 - Read image from swap to mem
 - Freeze process
 - Suspend devices
 - Restore snapshot image
 - Resume devices
 - Thaw processes



Swsusp (5):

- Swsusp の resume に時間がかかる原因
 - Swsusp 処理の開始が late_initcall
 - ほぼ init process の開始直前
 - 2重の I/O 処理
 - Swap to mem (allocated mem)
 - Mem to mem (allocated mem to orig mem addr)
 - 冗長なデバイス状態の遷移
 - Device gets setup, suspend, and resume again.



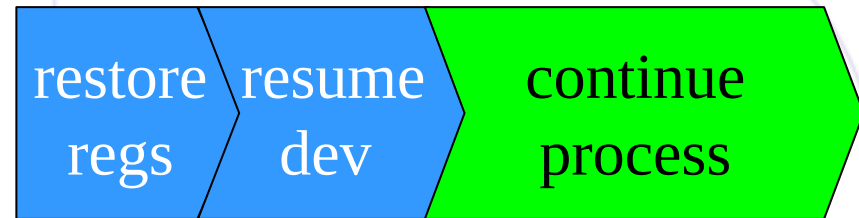
Snapshot boot (1):

- カーネルとブートローダの協調
- カーネルが作成したイメージを使用
 - 同一イメージを繰り返し使用する
- ブートローダ側の処理
 - ボードの起動(通常の役割)
 - スナップショットイメージのメモリへの展開
 - 復帰のための初期化・設定
 - カーネルの復帰ポイントへジャンプ

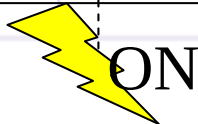


Snapshot boot (2):

KERNEL
LAND



BOOT
LOADER





Snapshot boot (3):

- u-boot ブートローダに実装
- 新しいコマンドとして実装:
 - bootss <image addr> <resume entry point>
- 処理手順:
 - FLASH上のスナップショットイメージをメモリに展開
 - 初期化とセットアップ
 - クロックスピードの設定
 - タイマーの設定
 - 各種レジスタの設定
 - MMUを有効にする
 - カーネルのレジューム処理の箇所にジャンプ



環境:

- Target board
 - OMAP Starter Kit (OSK 5912)
- Boot loader
 - U-boot
- OS
 - Linux 2.6
- Application
 - mplayer



Hardware Features:

ARM9 core operating at 192 Mhz.
DSP core operating at 192 Mhz.
TLV320AIC23 Stereo Codec
32 Mbyte DDR SDRAM
32 Mbyte Flash ROM
RS-232 Serial Port
10 MBPS Ethernet port
USB Host Port
Compact flash connector
On board IEEE 1149.1 JTAG



デモ:

- [[[demo]]]