

スマートフォン等の携帯端末における OS カーネルの研究

伊藤 僚[†] 須栗 裕樹[‡]

宮城大学大学院事業構想学研究科[†] 宮城大学事業構想学部[‡]

1. 研究目的

本研究は、省電力デバイスで用いるためのオペレーティングシステムのカーネルに必要な機能を設計・実装し、且つアプリケーションおよびユーザランドによって評価実験を行うことを目的としている。対象となる省電力デバイスは、第一に Android を搭載した高機能なマルチメディア端末であるスマートフォンとするが、それ以外のタブレット、また従来型の組み込みシステムにも対応可能とする。

2. 研究動機

今日、スマートフォンやタブレットといった高機能な省電力デバイスが急速に普及しつつある。これらと従来の組み込みシステムとを比較して異なる点は、マルチメディア、PC やクラウドとの連携、GUI 等、高度な機能を提供していることである。これらの OS の中で代表的なものとして、Android と iOS の 2 つが挙げられる。特に Android 端末は、2011 年 3 月時点の日本国内におけるシェアで iOS を上回る [1] など、躍進が目立つ。

Android と iOS に共通しているのは、どちらも Unix をベースとした OS だという点である。カーネルとしては、Android が Linux、iOS が Mach/BSD であり、どちらもデスクトップやラップトップの Linux ディストリビューションや Mac OS との連続性があるため、高度な機能を実装するのが容易である。

他方、これらは、数十年にも渡る Unix の歴史を引きずっているため、最近の高機能省電力デバイスに最適化されているとは言い難い面もある。例えば、SoC として実装されるヘテロジニアスマルチコアへの対応、省電力とハイパフォーマンスとの両立、制限されたユーザインターフェース、ライトウェイトなプロセス、従来のファイルシステムを超える概念、異なる粒度のリアルタイム性を担保するスケジューラ等が問題点として挙げられる。

3. 問題点

3.1 SoC として実装されるヘテロジニアスマルチコアへの対応

対象とする端末は SoC として CPU と GPU を一緒に搭載しているが、GPU のプロセスを UNIX のプロセスでは扱えない。

3.2 制限されたユーザインターフェース

画面が小さく、タッチスクリーンのみであるということ従来 UNIX ではうまく扱えない。

3.3 ライトウェイトなプロセス

UNIX はプロセスを新たに作るのも、一度終了した後再度立ち上げるのも一苦勞である。ヘビーウェイトなプロセスは省電力性に悪影響を及ぼす為、ライトウェイトなプロセスにする必要がある。3.4 従来のファイルシステムを超える概念

従来のファイルシステムはパスで場所を指定している。これを例えば RDB にすることで検索等も簡単になる。

Android においては UNIX のようにパスがそのまま見える。しかし、iPhone の場合はそんなことはない。内部的にはパスを使っているが、それをユーザには意識させないという点で親切である。エンドユーザとしては iOS の方が親切でわかりやすい。人に分かりやすい UI を素直に作るうとした場合、従来のファイルシステムを越えるものを作る必要がある。

3.5 異なる粒度のリアルタイム性を担保するスケジューラ

処理によって 1 秒待てるものもあれば、0.01 秒も待てないものもある。そのように桁の違うリアルタイム性に対して UNIX のカーネルは対応できていない。

4. 開発環境

VirtualBox を用いて、Ubuntu10.10 上で作業を行った。Sourcery G++ Lite 2009q3-67 をクロスコンパイラとして使用した。

また、環境を構築する過程でシェルスクリプトを実行する必要があったのだが、うまく実行できなかったことがあった。結論から言うと、パーミッションがうまく設定できていなかったためであり、ターミナルから

```
chmod a+x シェルスクリプトのパス
```

で実行権限を与えてやるときちゃんと動くようになった。因みに a+x とは、すべてのユーザに実行権限を与えるという設定である。それ以来、パーミッションをきちんと意識するようになった。

5. 対象とする端末

Samsung 製 Galaxy S の国内向けモデル SC-02B(DOCOMO)

主な仕様 [2]

- 筐体サイズ : 64.2 x 122.4 x 9.9mm、重さ 118g
- CPU : Samsung Intrinsity S5PC110 1GHz
- GPU : PowerVR SGX540

A Study on OS Kernel for Smartphone Devices

[†]Ryo Ito, Graduate School of Project Design, Miyagi University

[‡]Hiroki Suguri, School of Project Design, Miyagi University

- メモリ：RAM 512MB、ROM 16GB
- OS：Android2.3.3(Gingerbread)
- バッテリー：1,500mAh、連続待受 510 時間、連続通話 380 時間

6. 評価と結果

5.1 Ver.0.1

標準のソースコードを、前述した開発環境でコンパイルしたバージョンである。SIM カードを挿した状態で数時間程度運用してみたが、電話の発着信・メールの送受信・ネットワークへの接続等、通常の運用で想定される機能はすべて問題なかった。これにより、開発環境は問題ないと判断した。

5.2 Ver.0.2

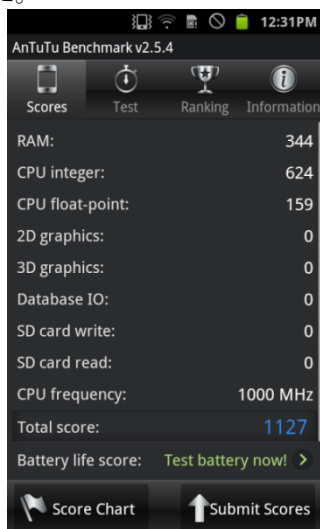
初回の起動と同時に root 権を取得できるよう、`initramfs` に手を加えたバージョンである。root 権を安全に取得する方法はいくつかあるが、今回は Android の起動スクリプトである `init.rc` に記述を追加し必要ないくつかのファイル (`BusyBox`、`SU` バイナリ等) を、シェルスクリプトを用いて `/system` 以下のディレクトリにコピーする方法を採用している。

root 権が取得できているかどうかは、root 権限を必要とするクロック制御用アプリケーション (`SetCPU v2.2.4`) で確認した。結果、問題なくアプリに root 権限を付与できた。

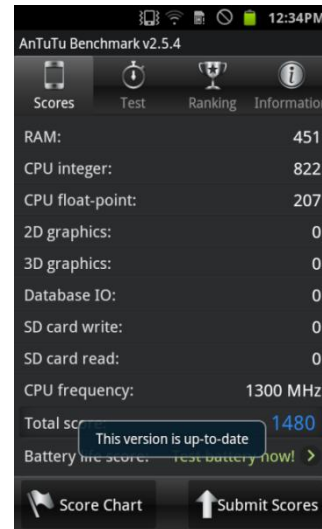
5.3 Ver.1.0

Ver.0.2 に加えて、パフォーマンス向上のため、1300MHz までオーバークロックできるようにソースコードの一部書き換えたバージョンである。具体的には、CPU の周波数を設定しているファイルの、クロックの定義について記述されている部分を変更・追加し、その定義を使用している部分を新しい定義に対応するように修正した。

このバージョンでは、ベンチマークアプリケーション (`AnTuTu v2.5.4`) を用いて本当に高速化できているか試した。



1.標準クロック(1000MHz)時



2.オーバークロック(1300MHz)時

スコアは、多少とはいえ向上しており、目的であったパフォーマンスの向上は達成出来た。

7. 今後の課題

これまでの研究では3で挙げた問題点を解決できていないので、今後はこれらに取り組んでいきたい。特に、②「ファイルシステムの改良」に重点を置いていく。

具体的には、ファイルシステムについての理解を深めることも兼ね、ファイル検索を速くすることを当面の目標とする。どのような形になるか分からないが、Windows でいうインデックスを貼って検索を速くするというようなことを試したい。また、Linux ディストリビューションはファイルシステムを追加するのが容易なので、新しくファイルシステムを追加するというようなこともやってみたい。

最終的には、WinFS[3]みたいなものを実装したい。

参考文献

- [1] スマートフォンの利用が加速！ グーグル・アンドロイド OS がアップル iOS のシェアを抜く！ - comScore, Inc
http://www.comscore.com/jpn/Press_Events/Press_Releases/2011/6/Google_Android_Leads_Acceleration_in_Smartphone_Adoption_in_Japan
 2012年6月20日
- [2] GALAXY S SC-02B サポート情報：サービス・機能とスペック | お客様サポート | NTT ドコモ
<http://www.nttdocomo.co.jp/support/utilization/product/sc02b/spec.html>
 2012年6月20日
- [3] WinFS で Windows のデータ/ファイル管理はどう変わる？
http://www.atmarkit.co.jp/fwin2k/insiderseye/20060126winfs/winfns_01.html
 2013年1月6日