

組み込み機器に搭載した Android の通信性能に関する評価

大杉 絵理奈 (指導教員：小口 正人)

1 はじめに

近年、米グーグル社が提供する Android はスマートフォンだけでなく組み込み機器の開発現場でも注目されてきている [1]。また、BeagleBoard[2] は低価格で高性能な小型のマザーボードとして現在注目を集めている。本研究では、Android が組み込み一般機器向けに採用され、実装されることを想定し、BeagleBoard に搭載された Android の様々な環境における通信性能をカーネル内部の振舞まで含めて解析し、質の高い通信の実現を目指す。

2 研究背景

2.1 Android

Android はスマートフォンやタブレット PC などの携帯情報端末を主なターゲットとして開発されたプラットフォームである。カーネルからミドルウェア、ユーザインターフェース、Web ブラウザ、電話帳などの標準的なアプリケーションソフトウェア群までを 1 つのパッケージにして提供しており、カーネルには Linux の関連技術を使用している。

2.2 組み込み機器における Android

組み込み機器は特定の機能を実現するために家電製品や機械、装置に組み込まれるコンピュータシステムである。今や、家庭用機器、産業用機器、医療用機器等、電子制御を必要とするほとんどの製品に用いられている。

しかし、現在の組み込み機器開発を取り巻く環境には、様々な課題と問題点が存在する。たとえばシステム環境は機種ごとに異なり、開発やデバッグに手間とコストが掛かる。しかし、Android の特徴（オープンソースであること、統一された開発環境であること、高い接続性と表現力・操作性があること）によって多くの問題が解決されると考えられ、現在注目されている。

2.3 BeagleBoard

BeagleBoard は BeagleBoard.org が開発および販売している組み込み機器向けの小型のマザーボードである。約 8cm 四方の基板であるにも関わらず豊富な外部インタフェースを備えており、他のマザーボードよりも低価格かつ省電力である事が特徴である。

2.4 カーネルモニタ

カーネルモニタ [6] は通信時に、どの時間にカーネルコードのどの部分が実行されて、その結果カーネル内部のパラメータの値がどのように変化したかを記録する事ができるツールである。図 1 に示すように、カーネル内部の TCP ソースにモニタ関数を挿入しカーネルを再コンパイルすることで TCP パラメータをモニタ可能にしている。これによりモニタできるようになった値には、輻輳ウィンドウ、ソケットバッファのキュー長その他、各種エラーイベント (Local device congestion, 重複 ACK, SACK 受信, タイムアウト検出) の発生タイミングなどがある。カーネルモニタによって、正常動作時のカーネルの振舞を知る事ができ、さらには通信

において問題が生じている場合に、その問題を特定し何が起きているのか調べる事も可能となる。

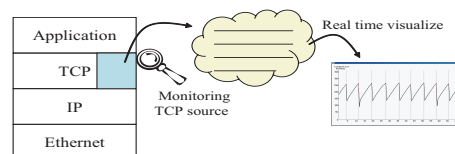


図 1: カーネルモニタ

3 研究概要

本研究では、まず Android のディストリビューションを取得し、これをビルドして、SD カードにコピーした Android を BeagleBoard から起動出来るように設定した。次に、Ethernet 接続での通信を評価した。また、Android 本体では利用できるコマンドが少ないため busybox[3] を、スループット測定のために iperf-2.0.5[4] をそれぞれクロスコンパイルし、SD カードに送り込んで利用した。また、輻輳ウィンドウの振る舞いを解析するために、カーネルモニタを Android カーネルに挿入した。

3.1 実験環境

表 1,2 および図 2,3 に本研究の実験環境の一部を示す。

表 1: 環境設定

Android 開発用	OS	Ubuntu10.04
	CPU	Intel Core i3
ホスト PC (Ubuntu)	Main Memory	4.00GB
	Clock Frequency	2.93GHz
BeagleBoard 制御用 PC (Windows)	OS	Windows XP
	CPU	Intel Pentium 4
	Main Memory	504MB
	Clock Frequency	3.00GHz

表 2: 環境設定 (Android on BeagleBoard 側)

BeagleBoard	Rev C4	
Android	version	2.1(eclair)
	distribution	0xdroid
	Java	5.0
	U-boot	2009.01-dirty

3.2 BeagleBoard の制御

Android を載せた BeagleBoard の動作をターミナルエミュレータである Tera Term で制御する。接続方法は図 2 の様に RS-232C によるシリアル接続とする。

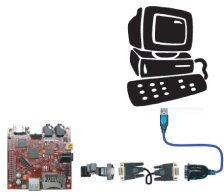


図 2: BeagleBoard 制御実験環境

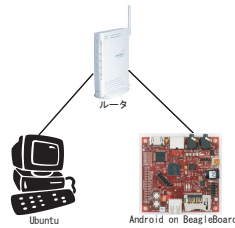


図 3: Ethernet 接続における実験環境

4 実験概要

4.1 ディストリビューションの取得

ディストリビューションとは Linux カーネルと Android のカーネル以外の部分を構成する Android Userland をひとまとめにしたものである。Android をビルドするホスト PC 上に環境設定とその他のパッケージをインストールし,repo を取得する。これを用い,Android の組み込み機器向けディストリビューションである 0xdroid[5] とカーネルのソースコードを取得しビルドする。

4.2 Android on BeagleBoard の起動

SD カードのパーティションを 2 つに分け,それぞれにファイルシステムを生成する。次に,先頭パーティションには xloader(MLO),u-boot,kernel イメージを,次のパーティションには Android の Userland を配置し,格納する。作成した SD カードを挿入した BeagleBoard を起動し,TeraTerm で SD カードからブートするように環境設定を行った。次に HDMI インタフェースにモニタを接続し,モニタへの出力の確認と,接続したマウスとキーボードが正常に動作することを確認した。

4.3 Ethernet 接続

Ethernet アダプタとして Logitech LAN-TX/U2H3B を用いた。対応するドライバ(asix.ko)を選択してカーネルを再コンパイルし,SD カードにコピーした。BeagleBoard で起動し,ドライバを有効にした後に DHCP で IP アドレスを取得し,インターネットに接続した。

有線接続における TCP 通信と UDP 通信のスループットを取得した結果をまとめたものが表 3 である。BeagleBoard と Ubuntu では,TCP 通信,UDP 通信に

表 3: スループットの測定結果

クライアント	サーバ	通信	平均 (Mbps)	標準偏差
BeagleBoard	Ubuntu	TCP	94.0	0.80
Ubuntu	BeagleBoard	TCP	94.2	0.66
BeagleBoard	Ubuntu	UDP	95.5	0.17
Ubuntu	BeagleBoard	UDP	95.7	0.21

関わらず,若干 Ubuntu の通信性能の方が安定しているものの,双方の通信性能にはほぼ差がないことが分かった。また,BeagleBoard から Ubuntu への通信の平均スループットも Ubuntu から BeagleBoard への通信の平均スループットも TCP 通信よりも UDP 通信の

方が若干高く,安定している事が分かった。

4.4 カーネルモニタの導入

Android カーネル内部の TCP ソースにモニタ関数を挿入しカーネルを再コンパイルすることでカーネルモニタを Android に組み込んだ。図 4 はカーネルモニタを Ethernet 接続したときの輻輳ウィンドウサイズの振舞である。サーバは Ubuntu,クライアントは BeagleBoard である。輻輳が起こっていないことから,非常に安定した通信を行っていることが分かった。

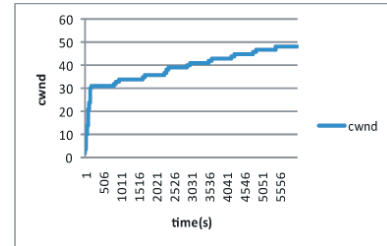


図 4: 輻輳ウィンドウサイズ

5 まとめと今後の課題

Android を BeagleBoard に搭載し,正常に動作することと Ethernet 接続が出来ることを確認した。また,TCP 通信,UDP 通信ともに Ubuntu と BeagleBoard で通信性能に大差がなく,TCP 通信よりも UDP 通信の方が若干性能が高く安定している事が分かった。さらに,カーネルモニタを動作させて Android の通信時の内部動作を解析することが可能である事を示し,Ethernet 接続では非常に安定した通信が行われている事が分かった。従来,組み込み機器においては,リソースやインタフェースの制限からプリミティブなデバッグしか行えなかったが,本研究のアプローチにより組み込み向け Android では,PC と同じ高度なデバッグが可能であることを示せた。今後はこの仕組みを用い,無線通信も含めた組み込み機器向け Android の通信性能の解析と向上を目指す。

参考文献

- [1] Android:<http://monoist.atmarkit.co.jp/fembedded/android/>
- [2] BeagleBoard:<http://elinux.org/BeagleBoard-JP>
- [3] busybox:<http://busybox.net/>
- [4] Iperf:<http://downloads.sourceforge.net/project/iperf/>
- [5] 0xdroid:<http://code.google.com/p/0xdroid/wiki/Source>
- [6] Reika Higa, Kosuke Matsubara, Takao Okamawari, Saneyasu Yamaguchi, and Masato Oguchi, "Analytical System Tools for iSCSI Remote Storage Access and Performance Improvement by Optimization with the Tools," In the 3rd IEEE International Symposium on Advanced Networks and Telecommunication Systems (ANTS2009), December 2009.
- [7] 大杉絵理奈,小口正人:組み込み機器に搭載した Android の通信性能に関する評価,情報処理学会第 73 回全国大会,2011 年 3 月発表予定