



コンピュータアーキテクチャ & オペレーティングシステムグループ



アーキテクチャとオペレーティングシステム

❖ コンピューターというハードウェアを音楽プレイヤーなどの応用プログラムに「使いやすい形」で提供する「基本的プログラム」がオペレーティングシステムです。「使いやすいかたち」に抽象化されたコンピューターの構成をアーキテクチャといいます。

現在の研究テーマ

- ❖ 現在卒論およびその準備として以下のテーマで研究を行なっています。各テーマのパネルに詳しい説明があります
 - サーバーアプリケーションの性能解析
 - マルチコアシステムによる電力効率改善
 - マイクロアーキテクチャの性能、電力効率への影響
 - IoT デバイスのセキュリティ

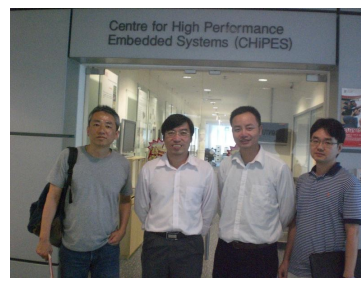
研究パートナー

- ❖ "Politehnica" University of Timisoara (ルーマニア)
 - 研究テーマ: リコンフィギュラブルコンピューティング



❖ Nanyang Technological University (シンガポール)

- 研究テーマ:
 - 組み込みシステムとリアルタイム OS
 - マルチコアシステムアーキテクチャ
- 交換留学 (二星翔 2010年9月から12月)
- 国際会議共催 (ISIC 2014)



❖ Universidade do Porto (ポルトガル)

- 研究テーマ:
 - 論理プログラミング言語の並列化
 - コンピューターシステムの最適化問題
- 交換留学
 - 高橋和晃 (2009年8月から9月)
 - Rui Rei (2010年1月から4月, ポルトから)
 - 古河智弥 (2015年8月から9月)
- ポルト教員, 大学院生による特別講義





コンピュータアーキテクチャ & オペレーティングシステムグループ

コンピュータと仮想化

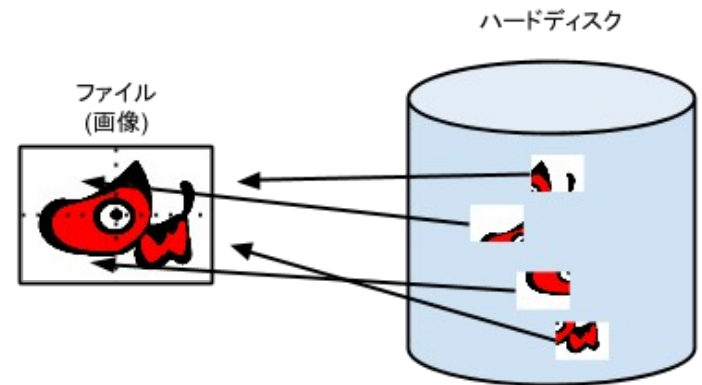
- ❖ コンピューターのハードウェアは複雑な構造で仕様も多種多様のため、そのままでは使いにくい。そこで色々な方法で「つかいやすく」見せかける工夫がされている。これを「仮想化」という。

マルチプログラミング

- ❖ 一台のコンピュータで常時複数のプログラムが走っている。例えば音楽プレイヤー、ネットワーク処理、画面処理等々。
- ❖ マルチプログラミング環境では、それぞれのプログラムが単独で動いているように仮定してプログラムすることができる。
- ❖ OS が実行中のプログラム (プロセス) を切り替え、各プロセスに独立した実行環境を提供する。また「入力待ち」のプロセスがシステム全体を止めたりせず、効率を上げるよう調整する。

ファイルシステム

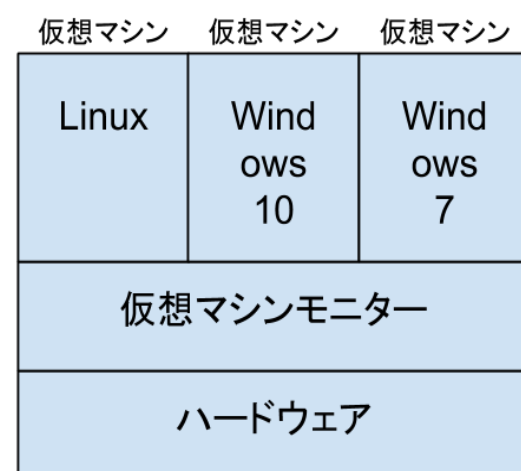
- ❖ コンピューターにはプログラムや多種のデータ (写真, 文章, 動画など) が大量に保存されている。
- ❖ これらのデータはハードディスクなどに記録される。ハードディスクはたくさんの「番号がふられた一定の容量の箱」(ブロック) からできている。
- ❖ 人間や応用プログラムにとってはブロック番号でアクセスや管理するのは使い勝手が悪い。OS は各情報を「ファイル」という単位で管理し、「名前」「フォルダー」でのアクセスや、セキュリティ (管理者のみアクセスできる、等) を提供する。



ファイルシステムはディスクブロックとファイルの対応や、アクセス権限、空きブロックの管理などを行う。

仮想マシン

- ❖ 一台のコンピュータ (物理マシン) を複数の「独立したコンピュータ」(仮想マシン) に見せる技術。
- ❖ 利点:
 - 各マシンで別々の OS (例: Windows と Linux) を同時走らせる事ができる。
 - コンピューター自体の価格に加え、ランニングコスト (電気代など) を抑える事ができる。
- ❖ 仮想マシンの OS による物理マシンのハードウェアへのアクセスにより、仮想マシン間での干渉が起きる。ハードウェアと仮想マシンの間に置かれた「仮想マシンモニター」(VMware など) がハードウェアへのアクセスを調整し、干渉を抑える。





ソフトウェアの特徴と マイクロアーキテクチャの関係

s1210241 学部 4 年 Yasushi Nagao

SPEC CPU2006

- ❖ SPEC CPU2006[1]はSPEC(Standard Performance Evaluation Corporation)によって開発された、プロセッサの性能評価用ベンチマークである。
- ❖ ベンチマークは 12 個の整数演算測定用(CINT)と 17 個の浮動小数点演算測定用(CFP)のプログラムにより構成される。
- ❖ プログラムごとに入力データが用意されており、いくつかのプログラムは複数の入力データを用いて複数回実行される。

ODROID C1+,C2

- ❖ ODROID-C1+,C2 は韓国の Hardkernel 社が販売しているボードコンピューターである。
- ❖ C1+と C2 はそれぞれ ARM 社の Cortex-A5,A53[2]を用いており、C2 はスーパー scaler による 2 つの命令の同時実行が可能である。
- ❖ C1+,C2 は命令セットが異なるため、同じプログラムを実行した際の命令数が異なる場合がある。今回の実験では命令数がほぼ等しいものを選択した。

実験

- ❖ C1+,C2 上で同じプログラムを実行した場合の命令数と実行時間、IPC を測定する。また、実行時間から相対性能を求める。
- ❖ 実行したプログラムは表 1 にまとめた。h264ref は入力データの中で最も大きな SSS を用いて実行した。

結果

- ❖ 相対性能を見てみると、同時実行の点で優れる C2 は C1+ に比べ、1.1 ~ 1.4 倍ほど性能が良かった。
- ❖ IPC の比率は最大でも 1.4 倍(h264ref)という結果となった。

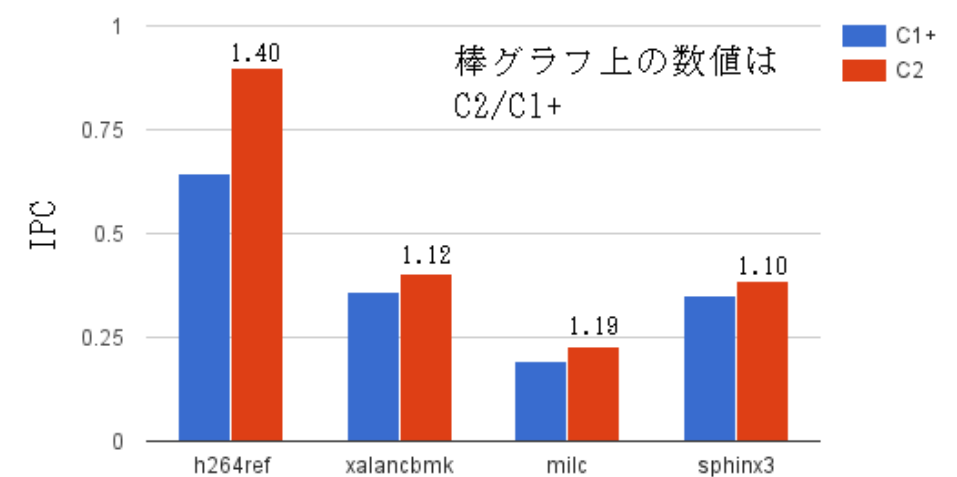
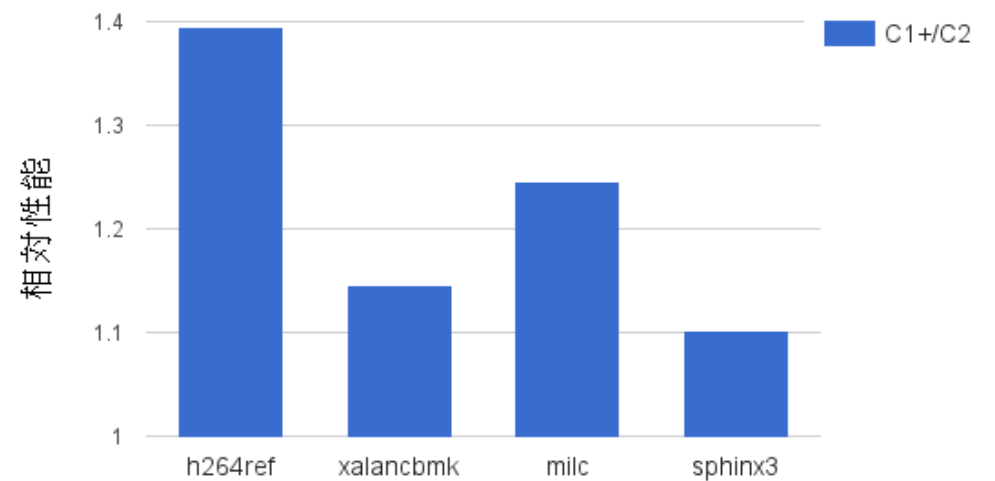


図 1: C1+,C2 における SPEC CPU2006 の相対性能(上)と IPC(下)

今後の課題

- ❖ perf [3]を用いてプログラムの処理の内部まで見ることで、ソフトウェアの特徴を調べ、マイクロアーキテクチャとの関係を見出す。
- ❖ 電力効率に与える影響を調べるために、電力測定環境を作る必要がある。

参考文献

- [1] SPEC, SPEC CPU2006, <http://www.spec.org>
- [2] ARM, ARM PROCESSORS, <http://www.arm.com>
- [3] perf, Linux Tools, <https://github.com/torvalds/linux/tree/master/tools/perf>

プログラム名	説明	C1+での命令数	C2での命令数	命令数の違い
h264ref (input:SSS)	CINT,非圧縮形式の動画ファイルをエンコードする	3.02×10^{12}	3.03×10^{12}	0.42%
xalancbmk	CINT,XMLドキュメントを他のドキュメントに変換する	1.01×10^{12}	9.93×10^{11}	1.92%
milc	CFP,格子ゲージ理論のシミュレーションを行う	1.07×10^{12}	1.02×10^{12}	4.70%
sphinx3	CFP,オーディオ形式のファイルが入力され、音声認識を行う	2.99×10^{12}	3.00×10^{12}	0.43%

表 1: 計測したプログラムの説明と命令数



IoT デバイスセキュリティ

学部 4 年 s1210052 Tsubasa Sugiyama

IoTとは

- ❖ IoTとは”Internet of Things”の略称で、従来は主にパソコンやサーバー、プリンタ等のIT機器が接続されていたインターネットに接続する”モノ”のことを指しています。具体的な動作としては、インターネットに接続されるようになった物が自身の状態や周辺の状態を収集し、通知や可視化、制御などを行います。

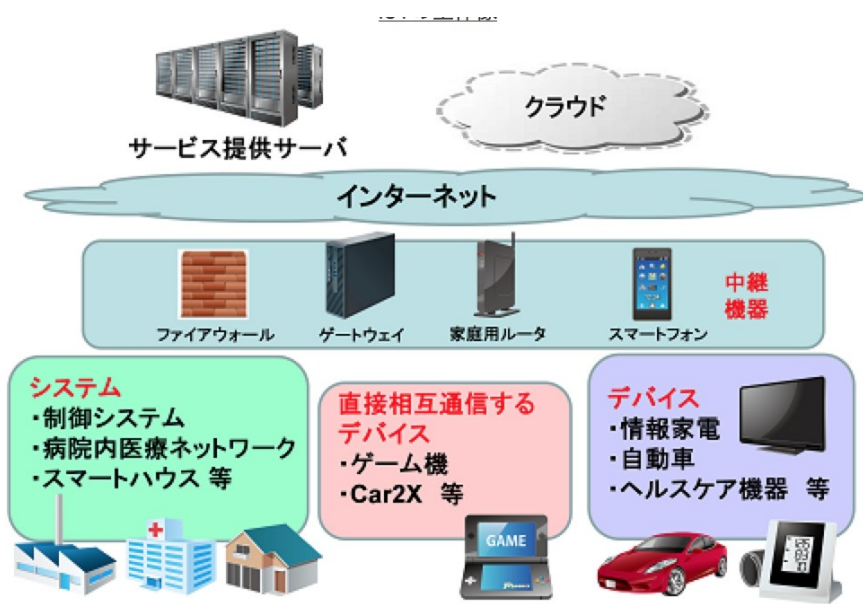


図 1: IoT の全体像 [1]

IoTのセキュリティ脅威

- ❖ 今後 IoT は世の中の課題を解決するために、利用され、多くの IoT が普及していきます。しかしこれに比例して IoT がサイバー攻撃を受けるリスクも増加していきます。
- ❖ IoT セキュリティの弱点[2][3]

-IoT デバイスの多くが省電力で軽量のデバイスという特性によって、従来のセキュリティ対策である通信の暗号化処理を行おうとしても、使えるエネルギーや電力が限られているため、エネルギーや電力不足が起こってしまい、暗号化が行えない場合があります。

-今までネットに繋がると想定されていなかったものがネットに繋がり、想定されていない被害が発生します。たとえば医療機器がインターネットに繋がることによって、患者の情報漏洩や医療機器の遠隔操作や停止などの想定外の被害が発生します。

IoTのセキュリティが侵害された例

- ❖ 車載機の例 [3]
モバイル経由で車載機に侵入し、車載ネットワークに繋がるチップのファームウェアを書き換えて、車載ネットワークに不正な命令を送り、遠隔からハンドルやエンジンを不正に操作されました。
- ❖ Web カメラの事例 [3]
遠隔から Web カメラの画像が見られる脆弱性が発見されました。これによってカメラの映像が見られ、場所が特定されてしまい、プライバシーを侵害されました。

研究について

- ❖ OWASP の IoT Security ToP 10 [4]を参考に、想定されるセキュリティリスクおよび、その対策(通信の暗号化など)を施したプロトタイプ的设计、実装を行います。
- ❖ 作成したプロトタイプに対し、想定されるセキュリティ攻撃(通信の盗聴や改ざんなど)のシミュレーションを行い、評価・分析を行う予定です。

参考文献

- [1] IoT 開発における設計の手引き (著) 独立行政法人 情報処理推進機構
- [2] Low-Energy Security: Limits and Opportunities in the Internet of Things (著)Wade Trappe,Richard Howard,Rovert S.Moore
- [3] IoT 時代のセキュリティ確保に向けて (著) 重要生活機器連携セキュリティ協議会
- [4] Internet of Things Top Ten (著) Open Web Application Security Project