

## 仮想計算機上でのリアルタイム OS の動作に対する評価と考察

永島 力<sup>†</sup>      金城 聖<sup>††</sup>      荒木 裕靖<sup>††</sup>      毛利 公一<sup>†</sup>  
<sup>†</sup>立命館大学情報理工学部    <sup>††</sup>立命館大学大学院理工学研究科

## 1 はじめに

近年、仮想化技術を搭載したマルチコアプロセッサの登場によって仮想計算機モニタ（以下、VMM）の研究が盛んになっている。VMMは、単一の計算機上で複数の OS を同時に動作させることを可能とするシステムソフトウェアである。VMMは、各仮想計算機（以下、VM）に対して実計算機資源を割当て、ゲスト OS の動作を可能としている。我々は、この仮想化機能を用いて、VM 上でリアルタイム OS を動作させ、そのリアルタイム性を保証する研究を行っている。

携帯電話や、車載システム、情報家電などの様々な組込みシステムでは、各種装置の制御、ユーザインタフェースなど機能ごとにプロセッサ、メモリ、OS が搭載されることがある。この場合、それらの数が増えてコストが増したり、製品のサイズが大きくなってしまったりといった問題がある。このような問題の解決策として、組込み分野においても仮想化技術が注目されている。プロセッサをまとめ、VM 上でリアルタイム OS を動作させることで、複数の OS を単一の計算機資源上で動作させ、コストの軽減や、物理的スペースを確保することが可能となる。

しかし、Xen[1]をはじめとした VMM は、リアルタイム OS を動作させることを目的に開発されていないため、リアルタイム OS の動作を保証していない。例えば、単一の計算機上で動作させるために計算機資源を仮想化しているため、CPU が割付けられていない状態や、ある一つのデバイスに対して複数の OS が I/O 要求をした際の問題などがある。

これを解決するために、VM 上でリアルタイム OS を動作させるにあたって、現在、VMM がどの程度のリアルタイム性をゲスト OS に提供しているのか、Xen 上でリアルタイム OS の動作について性能評価を行った。

以下、本稿では、2 章で代表的な VMM である Xen について、その特徴と Xen 上でリアルタイム OS を動かす際の問題点について述べ、3 章において Xen 上で動作するリアルタイム OS の性能評価について、その評価手法と計測結果を述べ、4 章で考察を行う。

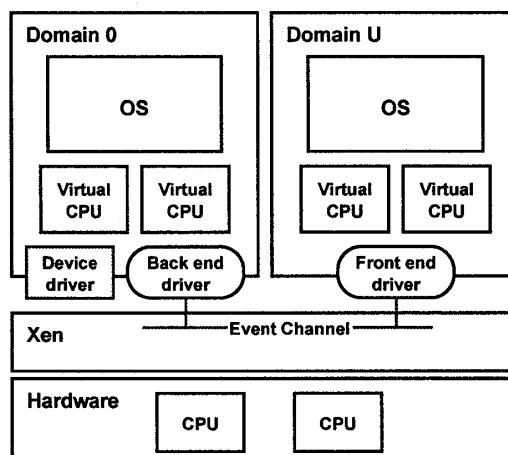


図 1 Xen の構成

## 2 Xen

## 2.1 概要

Xen は、CPU、タイマ、I/O デバイスなどの計算機資源を抽象化したドメインと呼ばれる環境を提供し、その上で OS を動作させる。図 1 に示すように、Xen では Xen の制御や VM の起動・終了を行うためのドメインをドメイン 0、ドメイン 0 上の OS に管理される OS のドメインをドメイン U と呼ぶ。ドメイン 0 には Xen に合わせて変更された Linux OS が使用される。

## 2.2 リアルタイム OS を動作させる上での問題点

Xen は、仮想 CPU を VM 上の OS に対して提供している。そして、VMM 内のスケジューラが仮想 CPU と実 CPU を対応付けることによって VM を実行させている。すなわち、スケジューラによって遷移する仮想 CPU の状態には、実行状態と実行状態でない状態があり、その状態によってタイマの処理が異なる。

仮想 CPU が実行状態である状態では、そのまま仮想タイマ割込みを発生させるのに対して、実行状態でない状態では仮想タイマ割込みを瞬時に発生させることができない。このため、仮想 CPU が実行状態へ移行際にタイマハンドラが動作していたか調べ、動作していた場合は仮想タイマ割込みを発生させ、ゲスト OS がタイマハンドラが動作した回数分のタイマ割込みを発生させる。このような仕組みから、VM 上で動作している OS には、タイマ割込みの発生間隔にばらつきが生じる可能性がある。

Evaluation and discussion about real-time OS and its behavior on virtual machine environment

Chikara Nagashima<sup>†</sup>, Akira Kanasiro<sup>††</sup>, Hiroyasu Araki<sup>††</sup>, and Koichi Mouri<sup>†</sup>

<sup>†</sup>College of Information Science and Engineering, Ritsumeikan University

<sup>††</sup>Graduate School of Science and Engineering, Ritsumeikan University

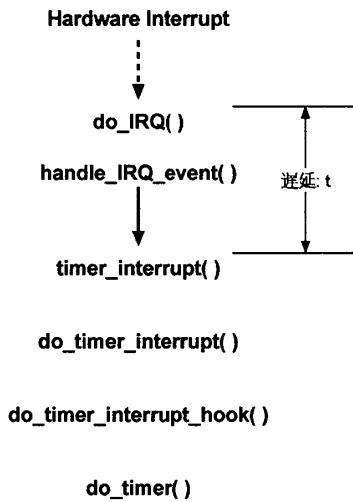


図2 タイマ割込みの処理内容

また、そのドメインが持つ仮想 CPU の状態によって、ハードウェア割込みを受けてからタイマ割込み処理の開始するまでの時間にも、ゆらぎが発生することが考えられる。これに対して、リアルタイム OS は、処理時間を予測してそのスケジューリングを決定しているため、デッドラインミスなどの問題が起こる可能性がある。

### 3 Xen 上で動作するリアルタイム OS の性能評価

#### 3.1 性能評価環境

今回、性能評価に利用した評価環境には、OS に Linux(2.6.23.9) を、CPU は Core 2 Duo T5600 を利用した。また、ゲスト OS は Xen の Virtual Machine Manager を利用して VM 上で構築し、OS にはドメイン 0 と同様に Linux(2.6.23.9) を利用した。

#### 3.2 性能評価

今回、リアルタイム性能として、割込みハンドラにおける遅延時間を計測した。具体的には、ハードウェア割込み発生からタイマ割込み処理が開始されるまでの時間を計測した。タイマ割込みは、図2のようにハードウェア割込みに対して、do\_IRQ() が起動し、handle\_IRQ\_event() が登録されたタイマ割込みハンドラを実行する。呼び出された timer\_interrupt() 以下がタイマ割込み処理を実行している。具体的な計測手法としては、図2のように、do\_IRQ() から timer\_interrupt() の開始までに要するクロック数について計測を行った。その結果、得られた計測結果を図3、図4に示す。

### 4 考察

図3、図4のように、ネイティブ Linux の Clock 数が 12000~13000 の間に集中したのに対し、VM 上の Linux の Clock 数は 15000~20000 の間に集中している。ま

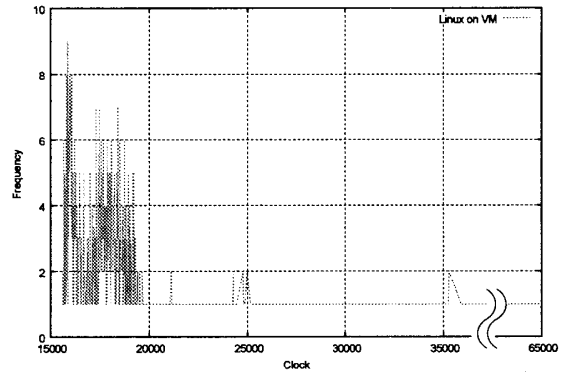


図3 VM 上の Linux での割込みハンドラにおける遅延

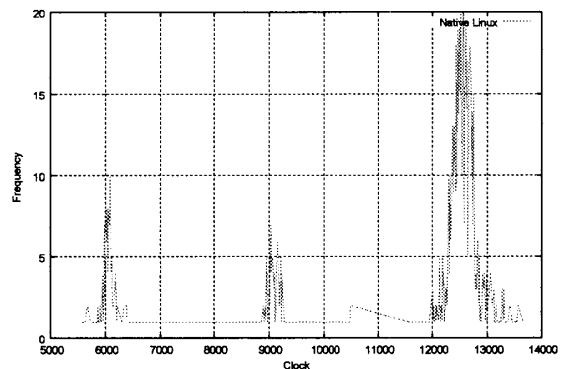


図4 Native Linux での割込みハンドラにおける遅延

た、その全体の範囲も、ネイティブ Linux の Clock 数が 5000~14000 の間に収まったのに対し、VM 上の Linux の Clock 数は 15000~65000 と広いのが分かる。

頻度については、一部にばらつきがあるものの、実環境上での Linux が 0~20 であるのに対し、VM 上の Linux の頻度は 1~10 となっていて広く分散していることが分かる。このため、リアルタイム OS が VM 上で適切にスケジューリングするには、このばらつきを抑える必要がある。

### 5 おわりに

本稿では、VMM の代表的な Xen について、その特徴と Xen 上でリアルタイム OS を動かす際の問題点について述べ、Xen 上でリアルタイム OS の動作について性能評価について述べた。

#### 参考文献

- [1] P. Barham, B. Dragovic, K. Fraser, S. Hand, T. Harris, A. Ho, R. Neugebauer, I. Pratt, A. Warfield: Xen and the Art of Virtualization, ACM Symposium on Operating Systems Principles, pp. 164-177, 2003.