

# Mintオペレーティングシステムを用いた 割り込み処理のデバッグ支援環境の提案

山本 凌平

岡山大学 工学部 情報工学科

平成26年2月14日

# 研究背景

OSの複雑化, 多機能化に伴いバグが増加  
割り込み処理は非同期性からデバッグが困難



割り込み処理のデバッグ手法やデバッグ環境が必要

<既存研究>

仮想計算機方式を利用したデバッグ

欠点: 任意のタイミング, 実時間での割り込み発生が困難

- ∴ (1) デバッグ対象OSとハイパーバイザの間の  
処理遷移のオーバヘッド
- (2) デバッグ支援機構の処理負荷

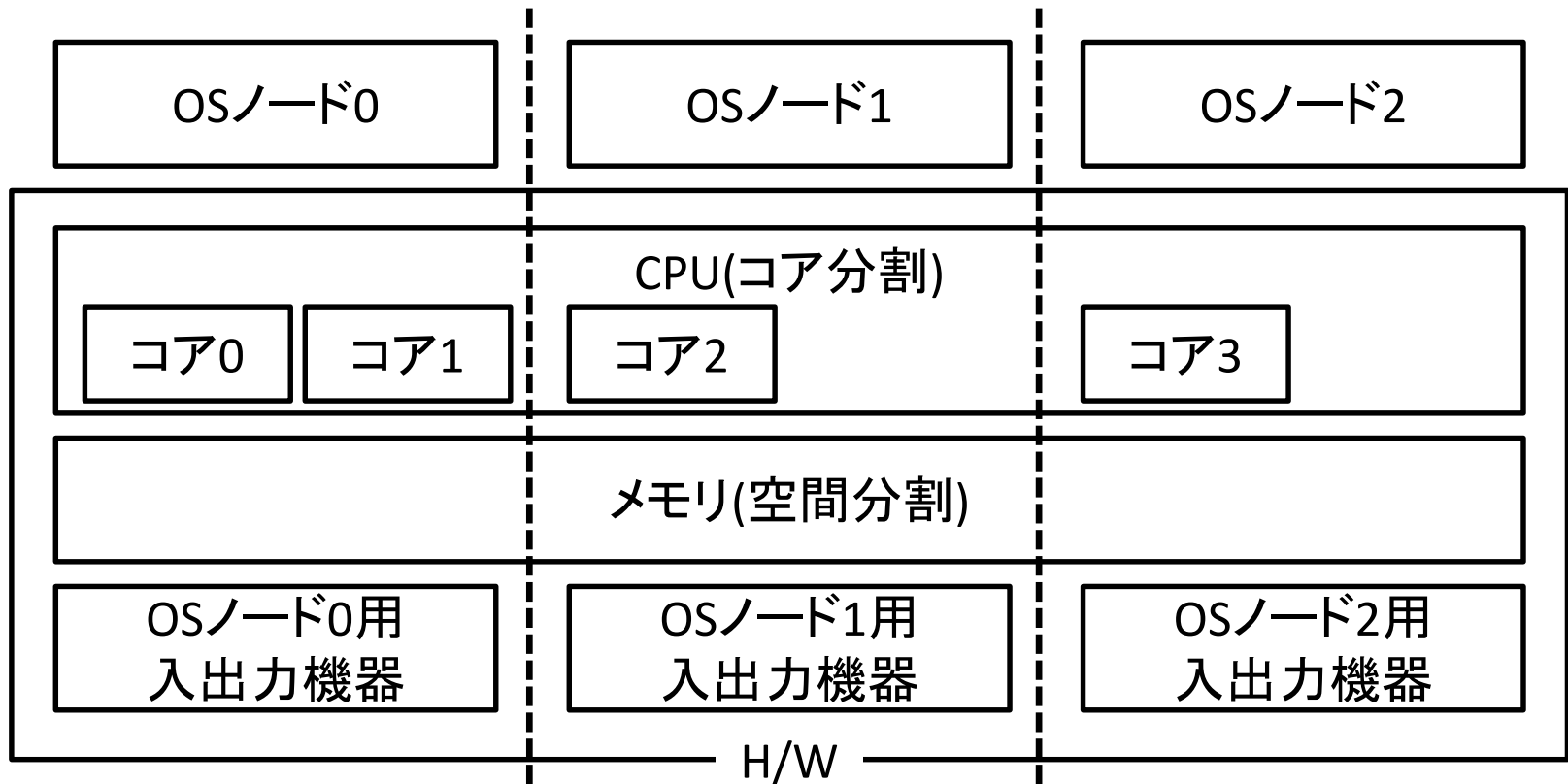


Mintを用いた割り込み処理のデバッグ支援環境を提案



仮想化なしにデバッグ支援OSとデバッグ対象OSを走行

# Mintオペレーティングシステム



- (1) CPU, メモリ, および入出力機器を分割し, 各OSノードが占有
- (2) ハイパーバイザを用いずに各OSノードが同時走行

# デバッグ支援環境の課題と対処

割り込み処理のデバッグを支援するため、  
任意のタイミング、実時間での割り込みを発生させる

## <課題>

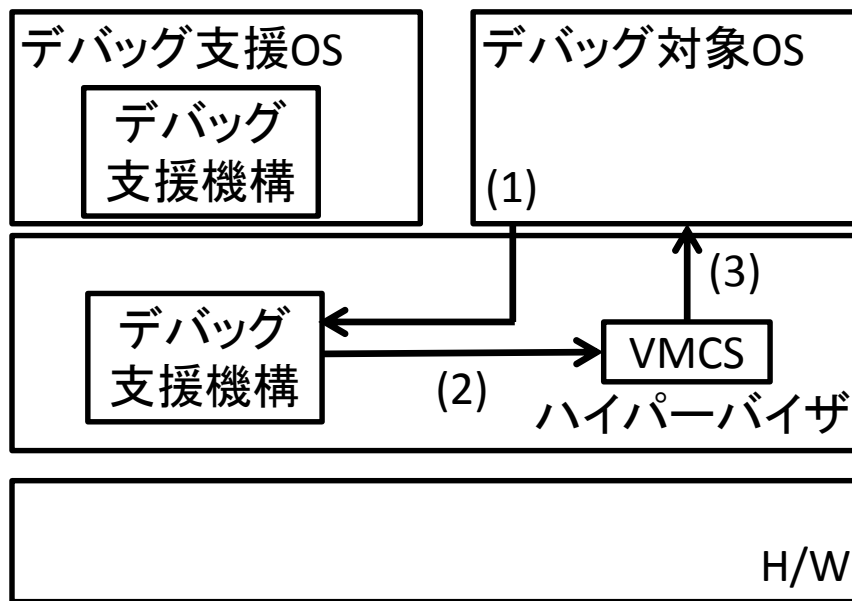
- (1) ハイパーコールを用いない
- (2) ハイパーバイザを用いない
- (3) デバッグ対象OSがデバッグ支援機構の  
処理負荷の影響を受けない



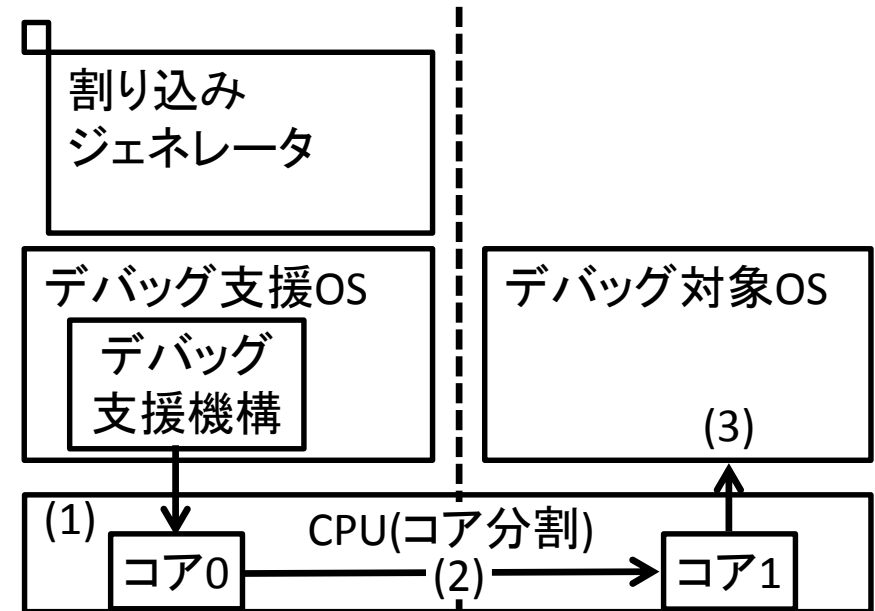
## <対処>

Mintを用いてデバッグ対象OSとデバッグ支援OSを同時走行させ、  
プロセッサ間割り込み(IPI)を送信

# 設計



割り込み挿入手法



提案環境

## <割り込み挿入手法>

VMCSを書き換え, 割り込みを発生

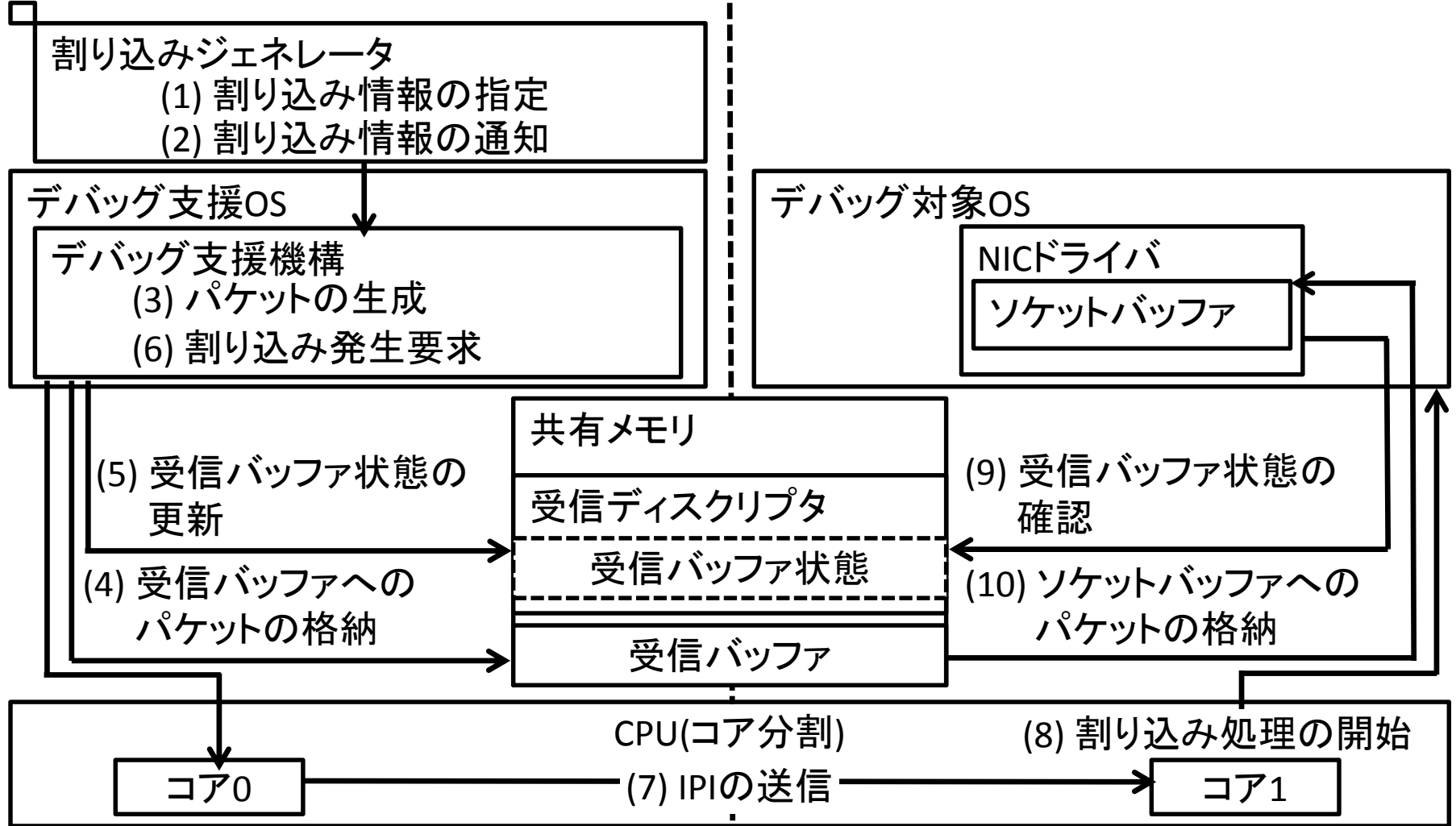
➡ ハイパーバイザによる処理負荷

## <提案環境>

ICRへの書き込みにより, IPIを送信

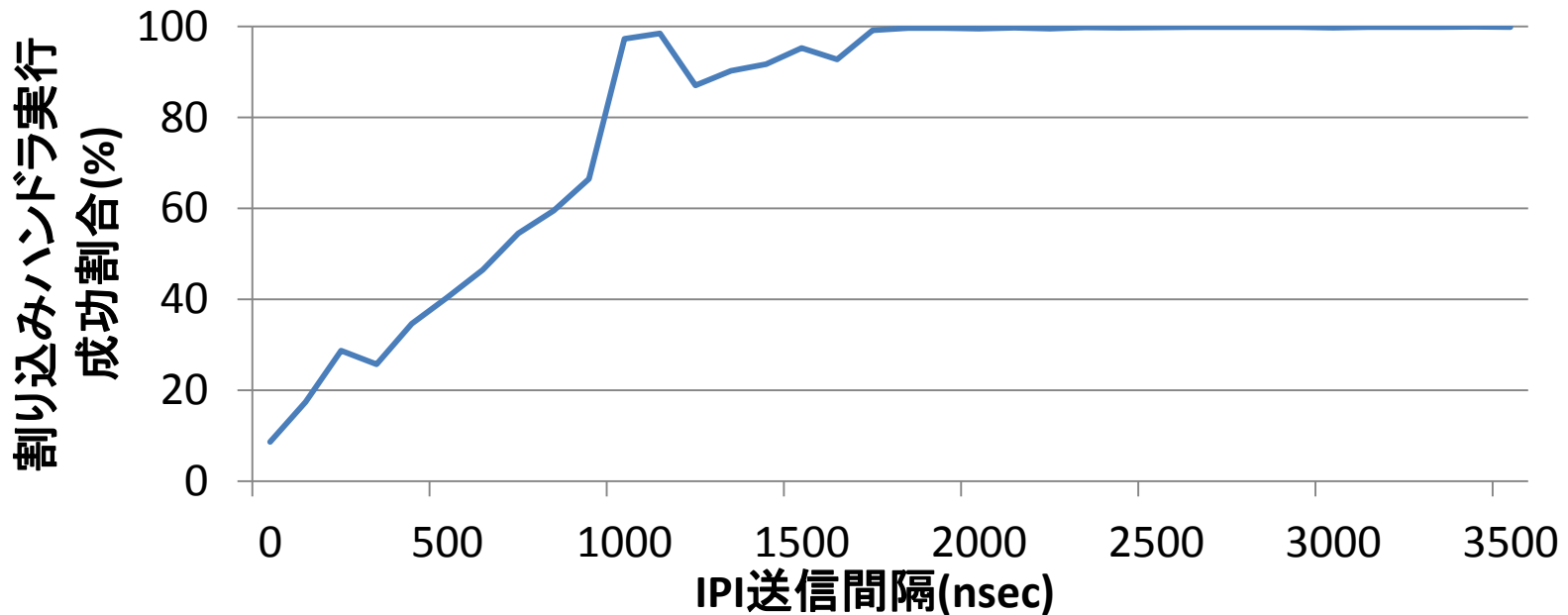
➡ ハイパーバイザの処理負荷無しにIPIを送信

# NICドライバを対象としたデバッグ支援環境



デバイス支援OSはNICの処理を実行

# IPIの送信間隔について



## <調査方法>

送信間隔を調整してIPIを送信し、割り込みハンドラ実行回数を計測  
IPIの送信間隔と割り込みハンドラ実行成功割合の関係を調査

(1) IPI送信システムコール: IPIを3000回送信

(2) 割り込みハンドラ: 実行時にメッセージを表示

## <調査結果>

IPIの送信間隔が2300nsec以上のとき、成功割合は約99.7%



十分小さい間隔で割り込みを発生可能

# まとめ

## <実績>

- (1) 仮想計算機を用いたOSのデバッグ手法の調査
  - (A) 割り込み挿入手法の調査
  - (B) ロギング/リプレイ手法の調査
  - (C) 仮想計算機の問題点の明確化
- (2) Mintを用いた割り込み処理のデバッグ支援環境の提案
  - (A) 実割り込みを発生させる環境の提供
  - (B) 任意のタイミングで割り込みを発生させる環境の提供
- (3) NICドライバを対象としたデバッグ支援環境の実装

## <残された課題>

- (1) デバッグ支援環境によるパケットの生成
- (2) 評価