SafeG
高信頼組込みシステム向けデュアルOSモニタ

Daniel Sangorrín, 本田晋也, 高田広章

名古屋大学

2010年12月3日

TOPPERS
Toyohashi OPen Platform for Embedded Real-time Systems

この研究の一部は文部科学省のサポート受けて実施しています

目次

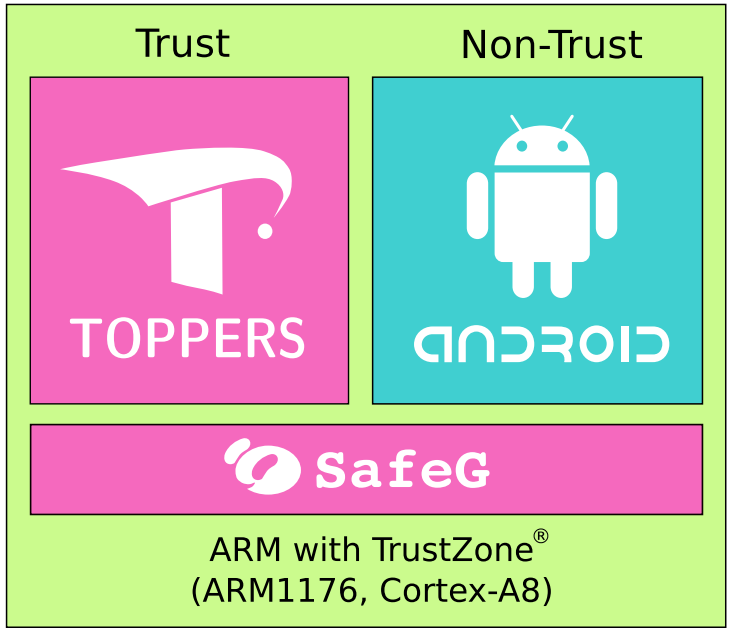
目次

自己紹介

- Daniel Sangorrín （サンゴリンダニエル）
- スペイン北部
- 経歴:
  - 2000 ~ 2006：電気通信の修士課程修了
  - 2006 ~ 2009：EUのFRESCORプロジェクトの研究者
    Cantabria大学の Computers and Real-Time group に所属
  - 2009 ~ 現在：名古屋大学大学院情報科学研究科情報システム学専攻 博士後期課程 2年 高田研究室
- 研究トピックス
  - 組み込みシステム
  - リアルタイムスケジューリング
  - 分散システム

概要

- RTOSと汎用OSが共存するシステムでは
  - RTOSのリアルタイム性とメモリ保護の確保が重要
  - OS毎に異なるハードウェアを使用する手法はコストを引き上げてしまう
- SafeGはシングルプロセッサで両アプリを安全に実行できる高信頼性デュアルOSモニタ



組み込みシステムの傾向

車載情報システムやスマートフォン等の高機能な組込みシステム

- 新たな機能を次々と取り込み，リソースは増大している
  - 例：ナビゲーション、ビデオ、ゲーム、インターネット
- 同時にリアルタイムアプリも実行しなければならない
  - 例：クルーズコントロール、携帯のベースバンドとセキュリティシステム

ARM Ltd. Copyright 2010

汎用OSとRTOSの連携の問題

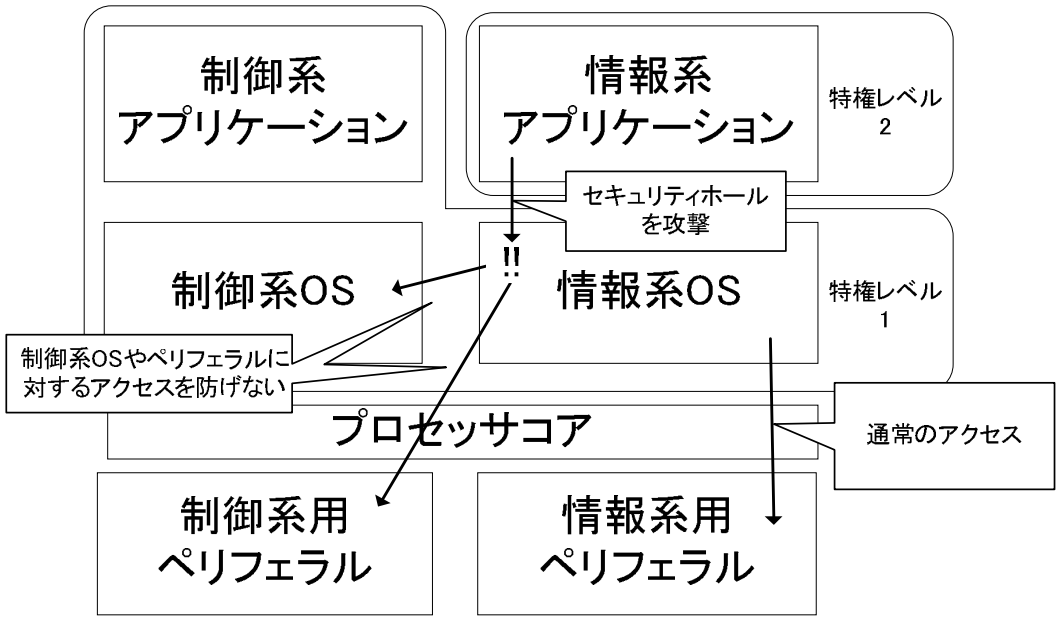
汎用OSとRTOSに対する要求は大きく異なる

- スループットvsリアルタイム性
- 高信頼性vs高機能
  - 新機能の取り込みのスピードを高めるためには，オープンなソフトウェアを実行できる必要がある
  - しかしながら，大規模になるため検証が非常に困難
  - 高い信頼性を確保するには，動作全てを検証する必要がある
  - しかし，新機能を取り込むスピードは低下する
- 単一のOSの上でこの相反する要求を同時に満たすのは困難です
- またOS毎に別々のハードウェアを使うのとコストが増加する

仮想化

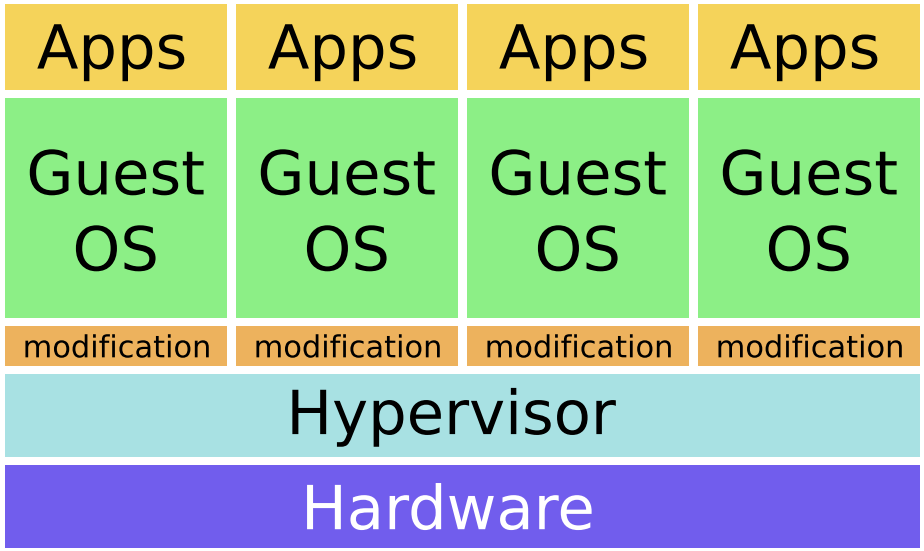
既存の組み込みシステム向けの仮想の方式

- 汎用OSへのパッチ（例：Linux RT）
  - RTアプリは保護されておらず信頼性が低い ×
- ハイブリッドカーネル（例：XenomaiやRTAIやRTLinux）
  - リアルタイム性と高パーフォマンス ○
  - しかし、メモリ保護がない ×



既存の組み込みシステム向けの仮想の方式

- VMM・ハイパーバイザー（例：OKL4、XtratuM、Integrity OS）
  - 汎用OSは非特権モードで動作するためメモリ保護は実現できる ○
  - 特権処理の実行オーバヘッドが高い ×
  - 汎用OSは大規模なため，変更やメンテナンスが困難 ×
  - プロセッサ以外のバスマスタから保護できない（例：DMA） ×



SafeGの目的

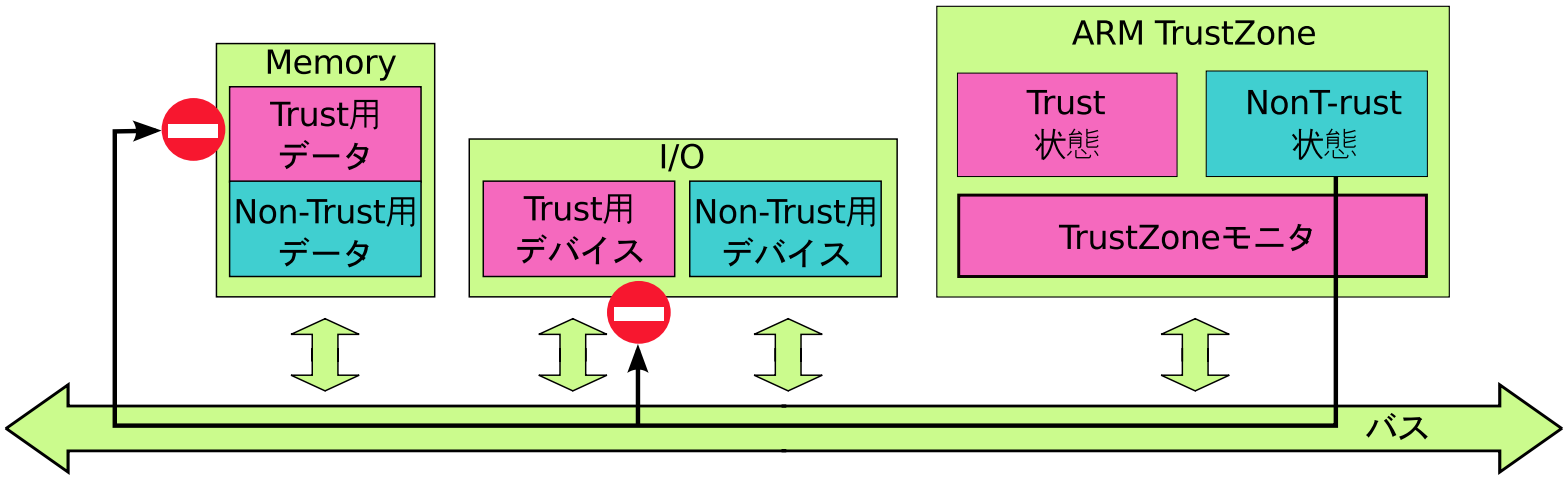
- 汎用OS(Linux/Android)とRTOS(TOPPERS/ASP)をシングルプロセッサで実行
- RTOSを安全に実行
- RTOSのリアルタイム性を保証
- 実行オーバヘッドは小さい
- 汎用OSの変更は少ない
- SafeGは小規模で検証が容易

目次

ARM TrustZone

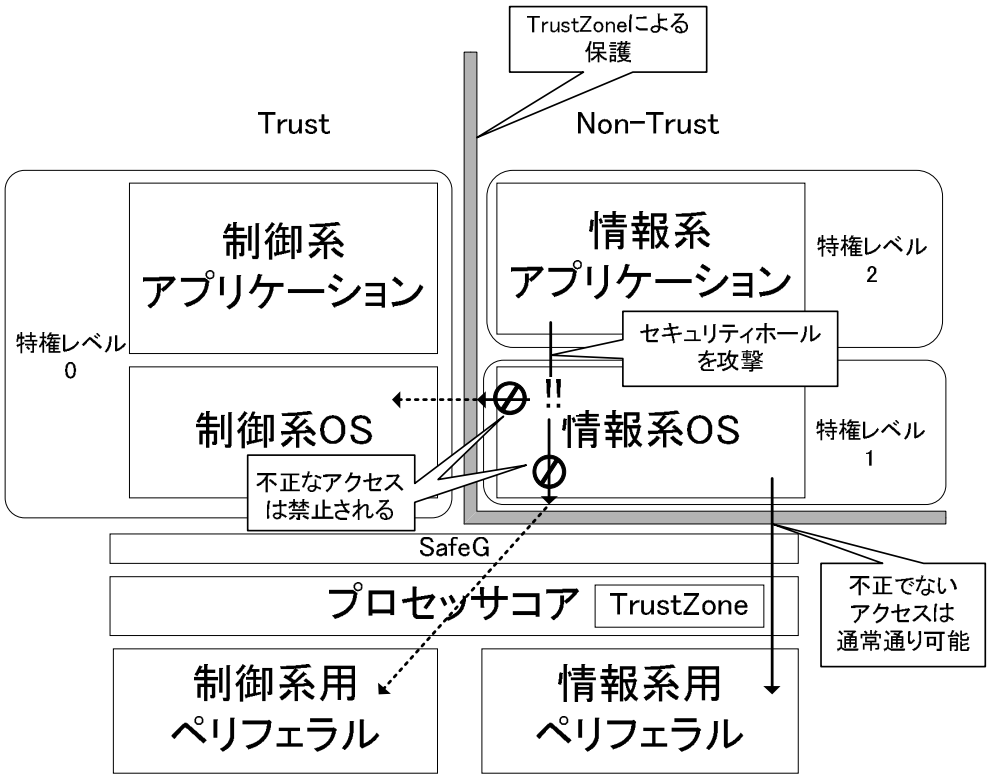
SafeGはアームのTrustZoneのセキュリティ拡張機能を利用する

- TrustZoneの概要
  - CPUにTrustとNon-Trustワールドを導入
  - メモリやデバイスをTrustとNon-Trustワールドに分割
  - FIQとIRQの割り込みはそれぞれのワールドで利用する
  - Trustのメモリやデバイスは Non-Trustワールドから保護されてる
- 三つの特権レベル：非特権・特権・モニタモード（低→高）
- モニタモードでCPUのTrustとNon-Trust状態を切り替える



SafeGのアーキテクチャー

- 汎用OSは大規模で，次々に新機能を導入する必要があるため，セキュリティホールが無くならない
- セキュリティホールに起因する汎用OSからの不正なアクセスからRTOSを保護

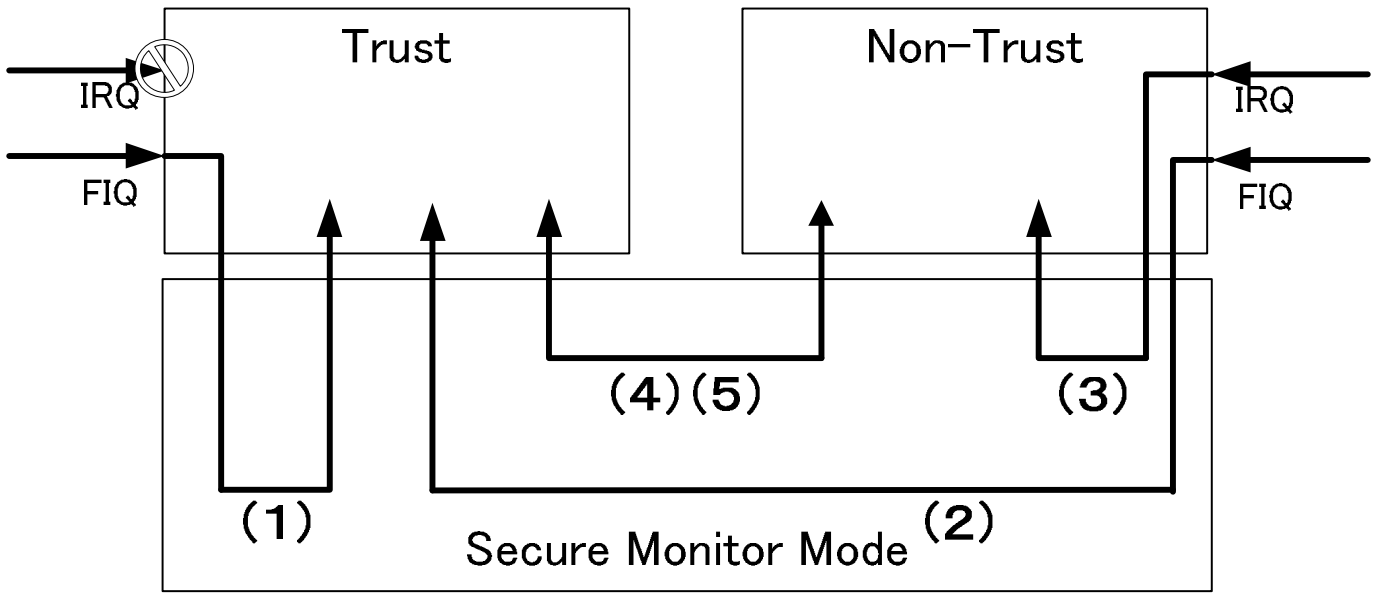


SafeGの詳細

- SafeGはTrustZoneのモニタとして実装
- 割り込み禁止状態で実行する
- RTOS実行中はIRQ(Non-Trust）割り込みは禁止
- 汎用OS実行中はFIQ（Trust）割り込みは許可
- 現状の対応プロセッサ：ARM1176JZF-S（TZPC搭載）
- 現状の対応ボード：PB1176JZF-SとIDEA6410
  - Cortex-A9へのポーティング中
- 対応するOS：
  - Non-Trust側：Linux・Android・TOPPERS/ASP
  - Trust側：TOPPERS/ASP

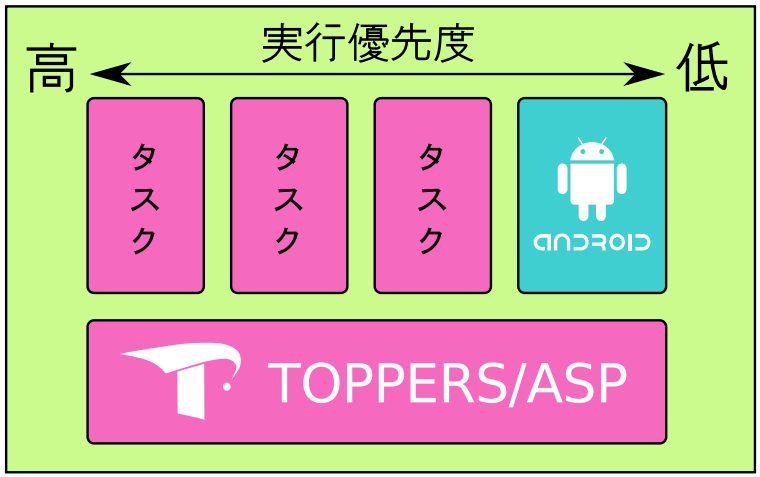
SafeGの実行パス

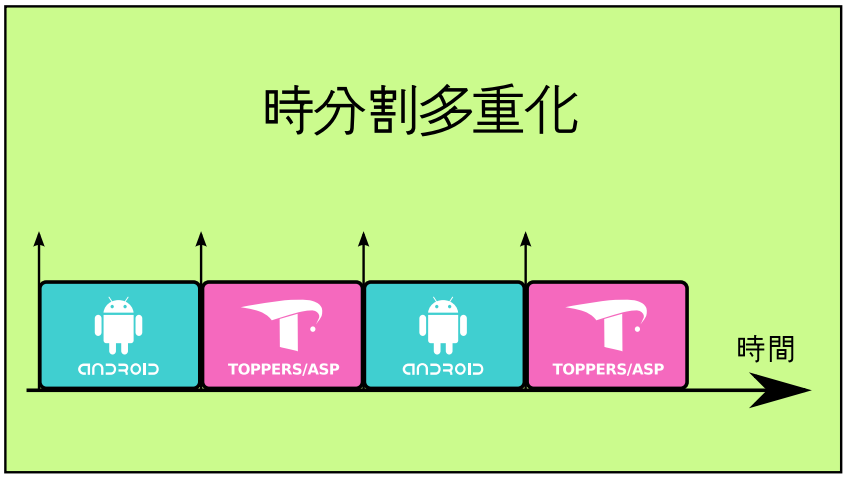
1. Trust状態でFIQ割り込みが発生
2. Non-Trust状態でFIQ割り込みが発生（SafeGはTrust状態に移行）
3. Non-Trust状態でIRQ割り込みが発生
4. SMCシステムコールの後でSafeGはCPU状態を切り替えす



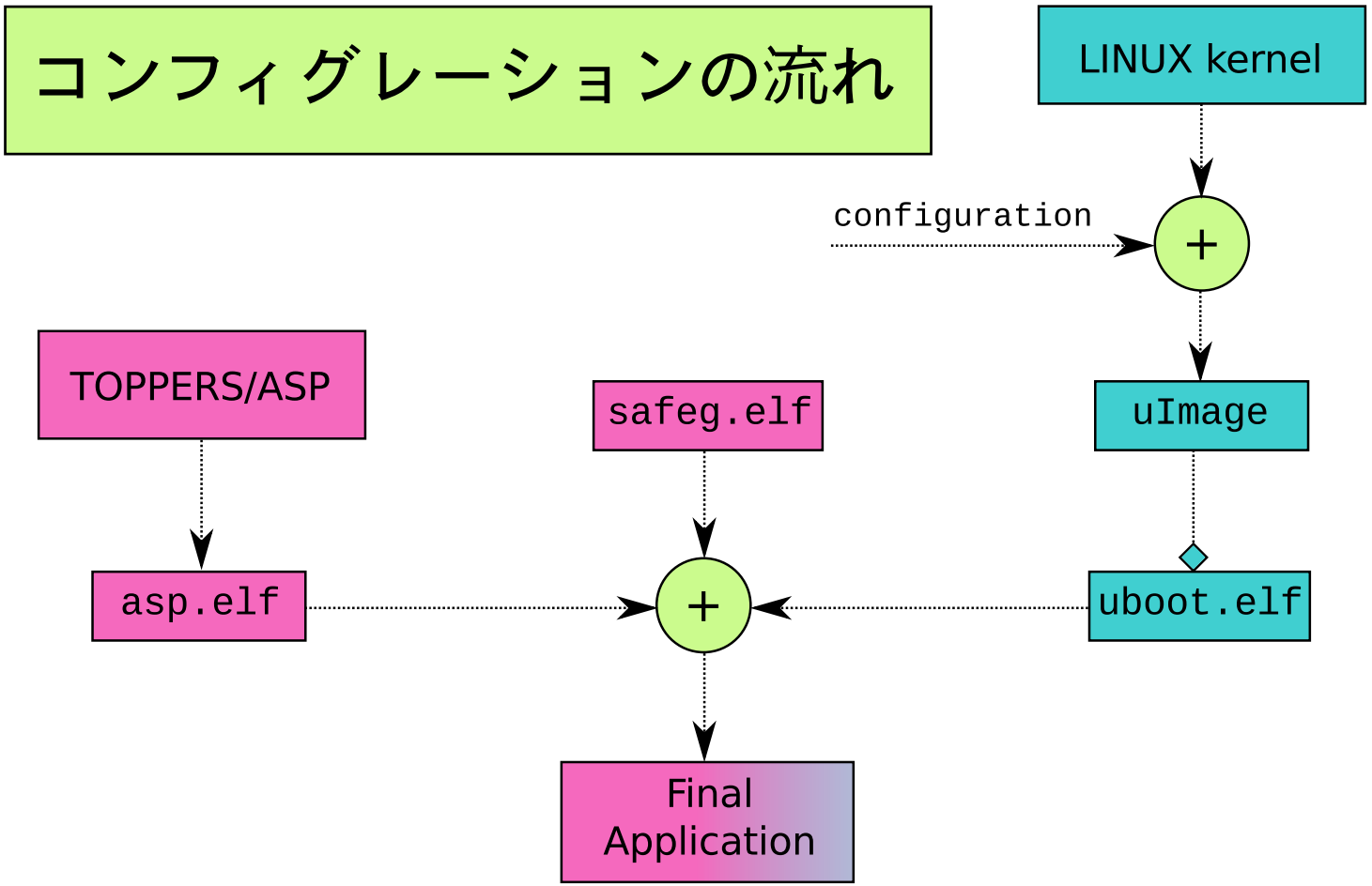
RTOSと汎用OSのスケジューリング

- 汎用OSはRTOSの1タスクとしてスケジューリング
  - RTOSのAPIに（例：μITRON）より動作を制御
- 対応するスケジューリング方式
  - 優先度スケジューリング
  - 時分割多重化スケジューリング



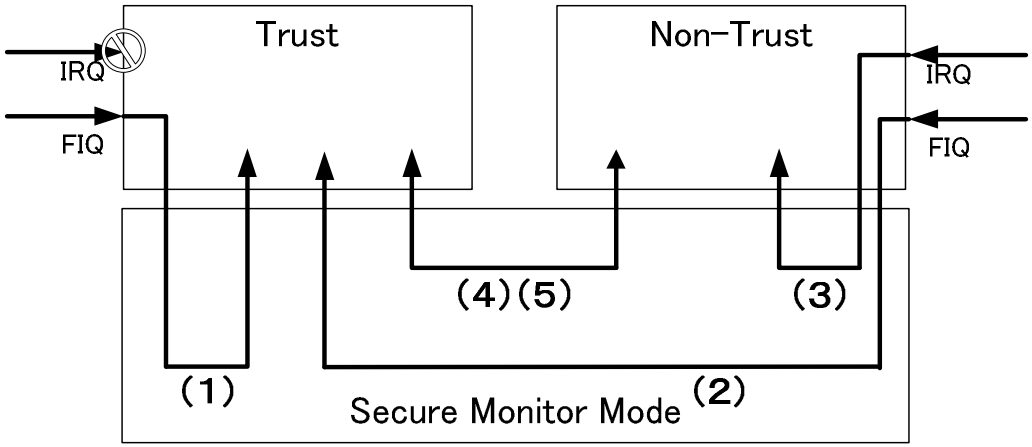


SafeGのコンフィグレーション



目次

SafeG内の実行オーバヘッド(210MHz)



パス	実行時間
(1) RTOS実行時にFIQ入力	0.7μs
(2) 汎用OS実行時にFIQ入力	1.6μs
(3) 汎用OS実行時にIRQ入力	1.2μs
(4) RTOSから汎用OSに移行	1.5μs
(5) 汎用OSからRTOSに移行	1.7μs
ASPの割込みベクタから割込み禁止解除まで	5.1μs

SafeGの検証

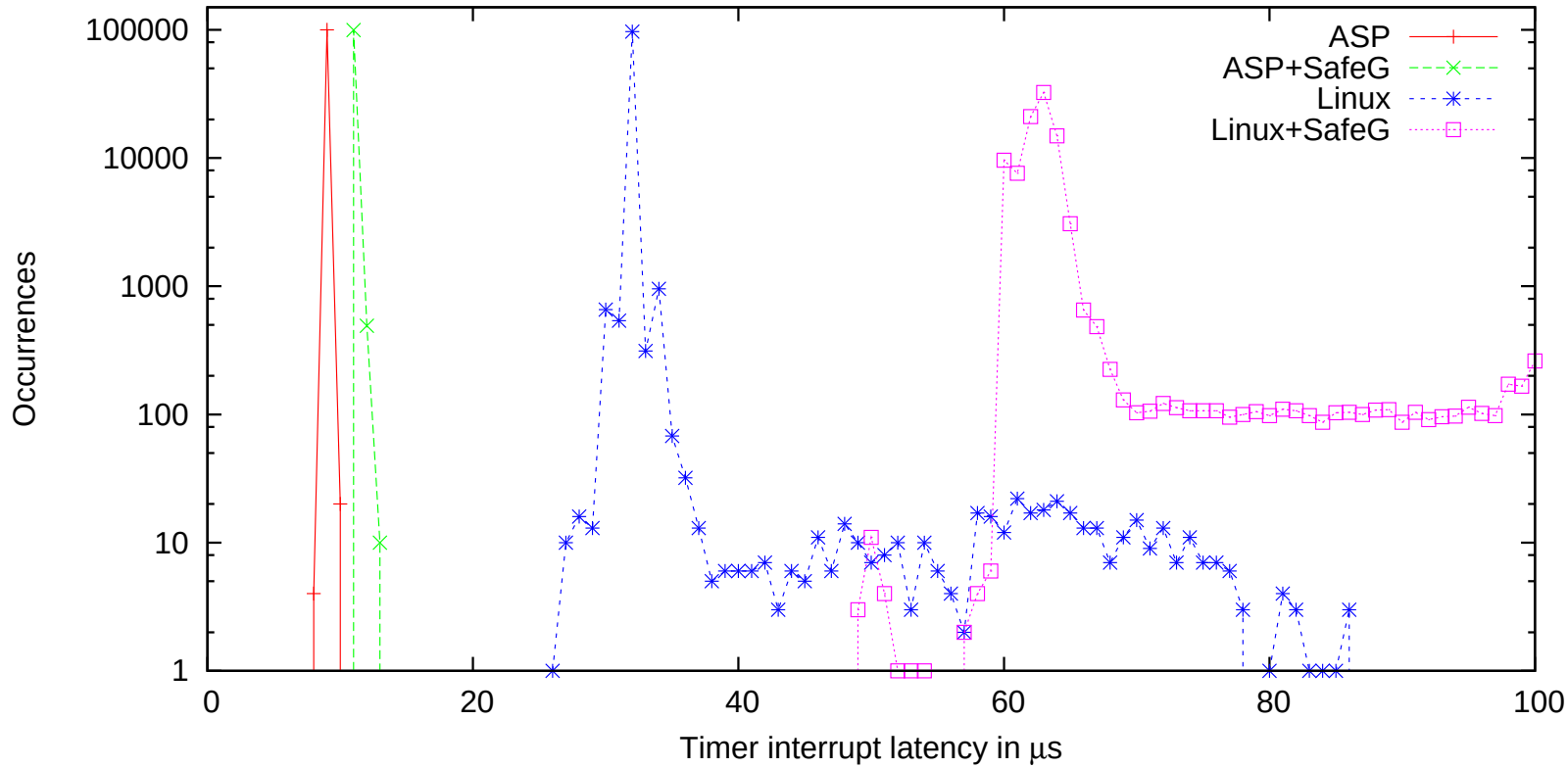
- コードとデータサイズ（バイト単位）

	text	data	bss	total
SafeG	1,520	0	448	1,968
ASP	34,796	0	83,140	117,936
Linux	1,092,652	14,8336	89,308	1,330,296

- RTOSと比較しても十分小さく実装できたので検証可能

RTOSのリアルタイム性の確保の評価

- ASPとLinuxのタイマー割り込み応答時間
  - 一定範囲内に収束しているので，最悪応答時間は一意に定まると言える

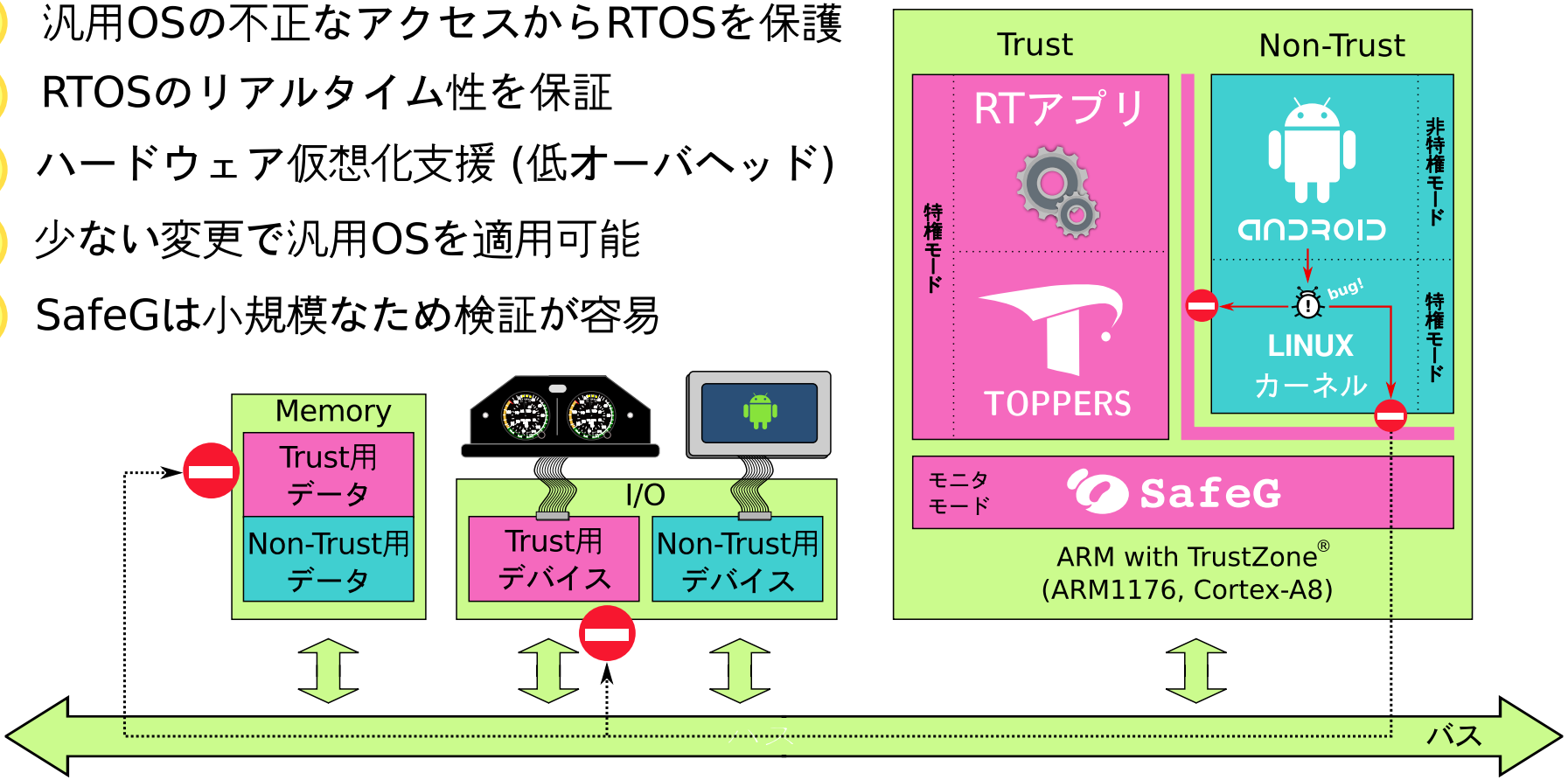


目次

まとめ

SafeG 高信頼組込みシステム向けデュアルOSモニタ

1. 汎用OSとRTOSをシングルプロセッサで実行
2. 汎用OSの不正なアクセスからRTOSを保護
3. RTOSのリアルタイム性を保証
4. ハードウェア仮想化支援 (低オーバヘッド)
5. 少ない変更で汎用OSを適用可能
6. SafeGは小規模なため検証が容易



未来のロードマップ


ロードマップ
OS間通信機能、
ミックスド
スケジューリング
を順次開発
一般公開
TOPPERS会員
向け公開
2010年12月
2011年5月
2011年度以降

質問

ご清聴ありがとうございました
Gracias por su atención
Thank you for your attention

Contact: dsl@ertl.jp