

FMPカーネル性能評価プログラムマニュアル

名古屋大学 大学院情報科学研究科
附属組込み研究センター

性能評価プログラムの概要

- 目的
 - FMPカーネルのサービスコール発行に要する時間を測定する.
- 評価項目
 - FMPカーネルを2コアで動作させ以下のサービスコールを評価する
 - act_tsk
 - mig_tsk
 - mact_tsk

性能評価プログラムのファイル構成

- act_tsk性能評価
 - perf_act_tsk.c/h/cfg
- mig_tsk性能評価
 - perf_mig_tsk.c/h/cfg
- mact_tsk性能評価
 - perf_mact_tsk.c/h/cfg

ターゲット依存部の設定(1/2)

本性能評価プログラムを動作させるためには、
ターゲット依存部において、以下の設定が必要となる

- 時間計測用関数の定義
 - マクロ HIST_GET_TIM として定義する. 定義しない場合は, get_utm()を用いる.
 - 計測の前後で呼び出す.
 - 引数と戻り値は次の通りである
 - ER HIST_GET_TIM(HISTTIM *p_tim)
- 時間計測用関数で取得するデータの型
 - マクロ HISTTIM として定義する
- 計測結果単位変換関数の定義
 - マクロ HIST_CONV_TIM として定義する.
 - HIST_CONV_TIMで計測した2点間の差分を引数にとり, 単位変換した結果を返す.

ターゲット依存部の設定 (2/2)

- 実行時間計測開始時フック
 - マクロ HIST_BM_HOOK として定義する.
 - 実行時間計測の開始時に呼び出す. キャッシュのパーズ等を想定
- 測定用タイマの初期化関数
 - void perf_timer_initialize(intptr_t exinf);
 - 性能評価プログラムのcfgファイルにおいて, コア1(クラスTCL_1)で起動時に呼び出されるように, ATT_INI指定されている.
- 測定前後のフック
 - ターゲット依存で, 各コアで測定の前後に行いたい場合は, 次のマクロに定義する.
 - 計測中のタイマ割込みを禁止等を想定している.
 - CPU1_PERF_PRE_HOOK : CPU1測定開始時フック
 - CPU1_PERF_POST_HOOK : CPU1測定終了時フック
 - CPU2_PERF_PRE_HOOK : CPU2測定開始時フック
 - CPU2_PERF_POST_HOOK : CPU2測定終了時フック

性能評価プログラムの実行方法

- プロジェクトディレクトリの作成
 - Makefileはsample1付属のものをベースにする
 - コア数は2とする.
- コンパイル対象のファイルのコピー
 - テスト毎のファイル(h/c/cfg)
- Makeファイルの編集
 - APPNAMEをテスト毎のファイル名とする
 - APPL_COBJS に histogram.o を追加

性能評価内容詳細

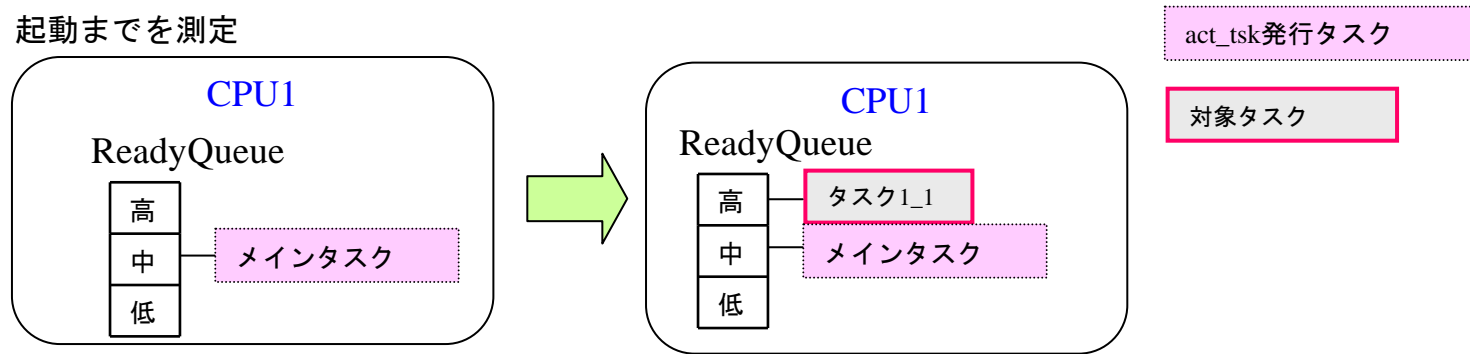
act_tsk 性能評価：概要

- 対象タスクの状態により分類
 - パターン1：対象タスクが休止状態 → ReadyQueueにつなぐ
 - パターン2：対象タスクが休止状態以外 → キューイング数の操作のみ
- 上記パターン1, 2に対して、対象タスクが起動するコアのレディキューの状態により、さらに分類し、【1】～【5】のパターンを抽出
 - 【1】対象タスクの優先度が、実行状態のタスクより高い。実行状態のタスクを切り替える。
 - 【2】空のReadyQueueにつないで、実行状態へ。
 - 【3】対象タスクの優先度が、実行状態のタスクより高いので、実行状態のタスクを切り替える。
 - 【4】対象タスクの優先度が、実行状態のタスクより低い。ReadyQueueにつなぐ。
 - 【5】起床待ち状態のタスクに対してキューイング数を＋する。

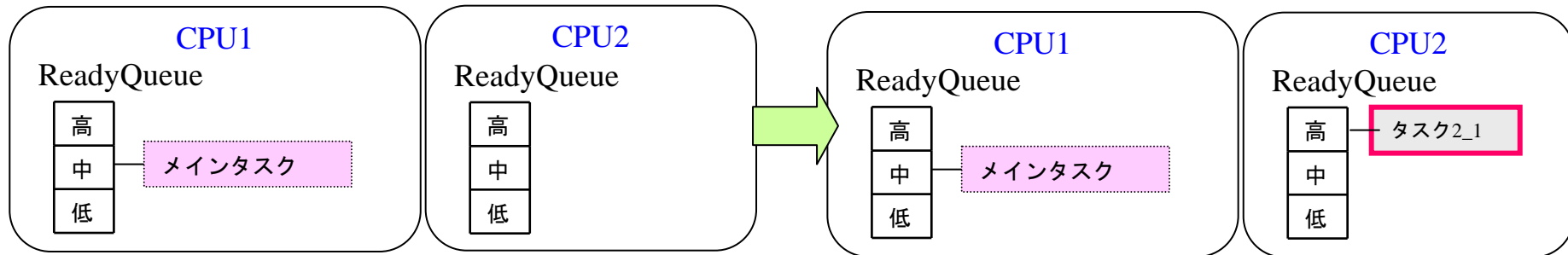
対象タスクの状態	対象タスク	最高優先度	実行状態のタスクと同優先度	実行状態のタスクより低い優先度	対象タスクのみ
パターン1 休止状態 (レディキューへつなぐ作業発生)	自CPU	【1】		【4】	
	他CPU	【3】			【2】
パターン2 休止状態以外 (レディキューへつなぐ作業なし)	自CPU	【5】			
	他CPU				

act_tsk 性能評価：詳細(1)

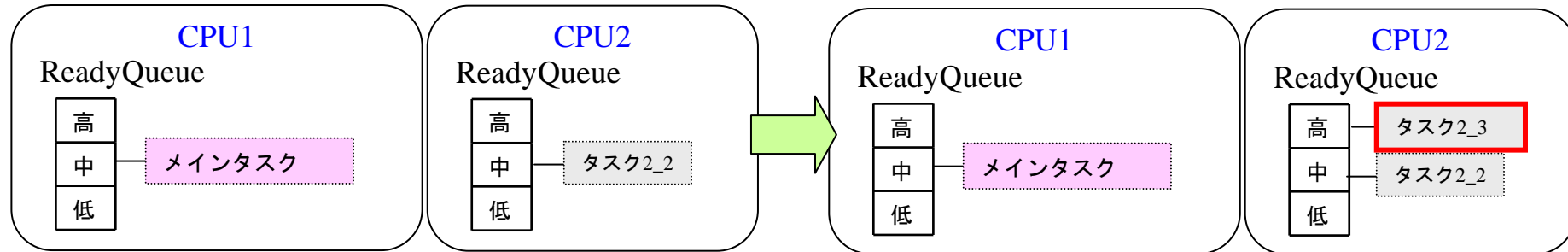
【1】 act_tsk発行前から、起動までを測定



【2】 act_tsk発行前から、起動までを測定

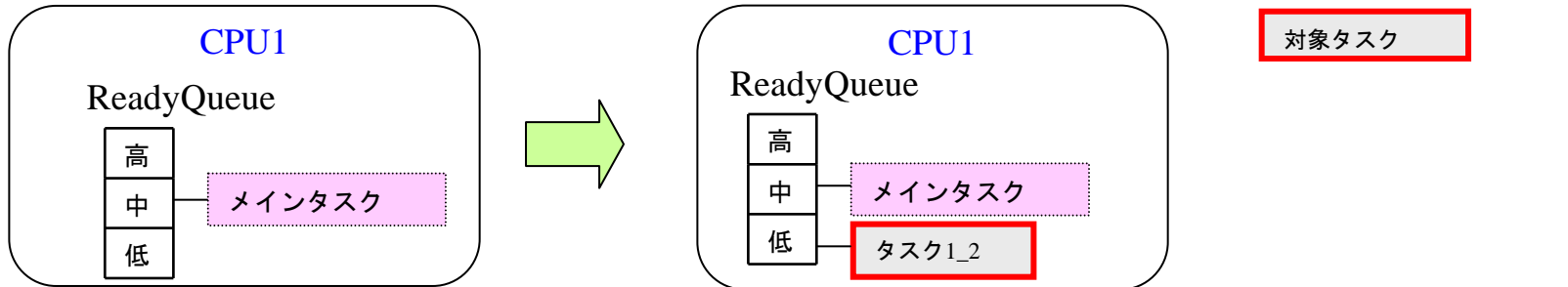


【3】 act_tsk発行前から、起動までを測定

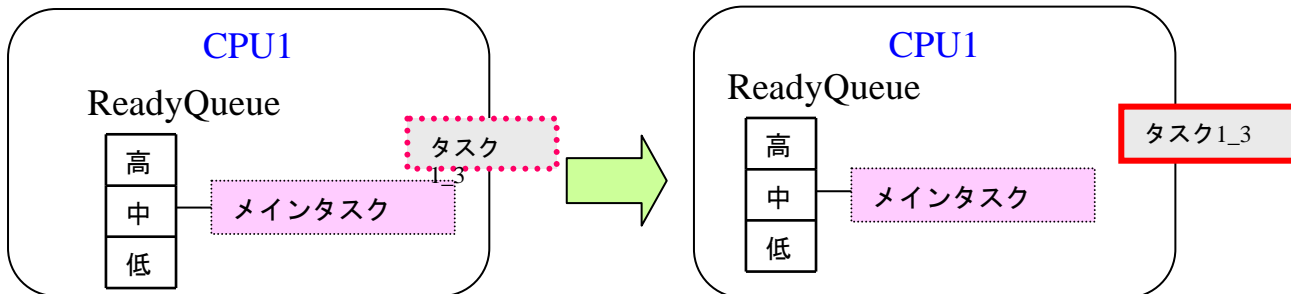


act_tsk 性能評価 : 詳細(2)

【4】 act_tsk発行前から, act_tsk終了までを測定



【5】 act_tsk発行前から, act_tsk終了までを測定



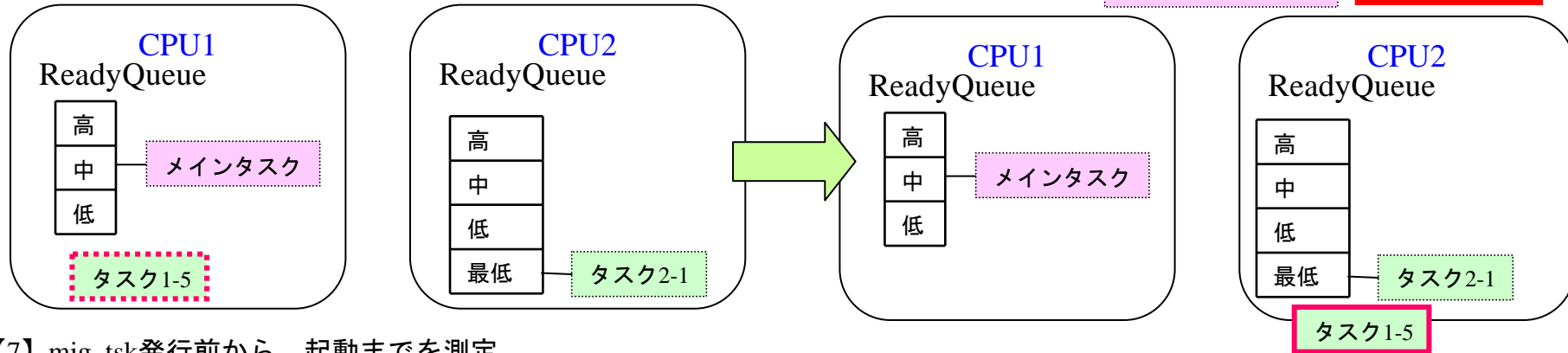
mig_tsk 性能評価：概要

- ・ mig_tskの実行に関して
 - ・ 【6】対象タスクが、休止状態
 - ・ 【7】対象タスクが実行可能状態かつ、起動の結果実行状態となる
 - ・ 【8】対象タスクが実行状態かつ、起動の結果実行状態となる

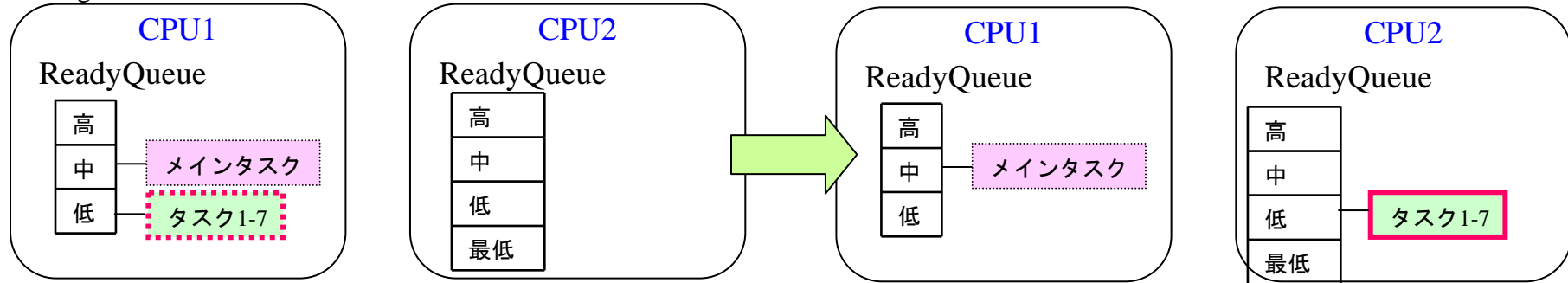
起動の結果 対象タスクの状態	最高優先度	実行状態の タスクと 同優先度	実行状態の タスクより 低優先度	レディキューが 空の所へ 移動する
パターン1 休止状態 (レディキューに つながっていない)	【6】			
パターン2 ready状態 (レディキューに つながっている)	【7】			
パターン3 自分自身 (running状態)	【8】			

mig_tsk 性能評価：詳細

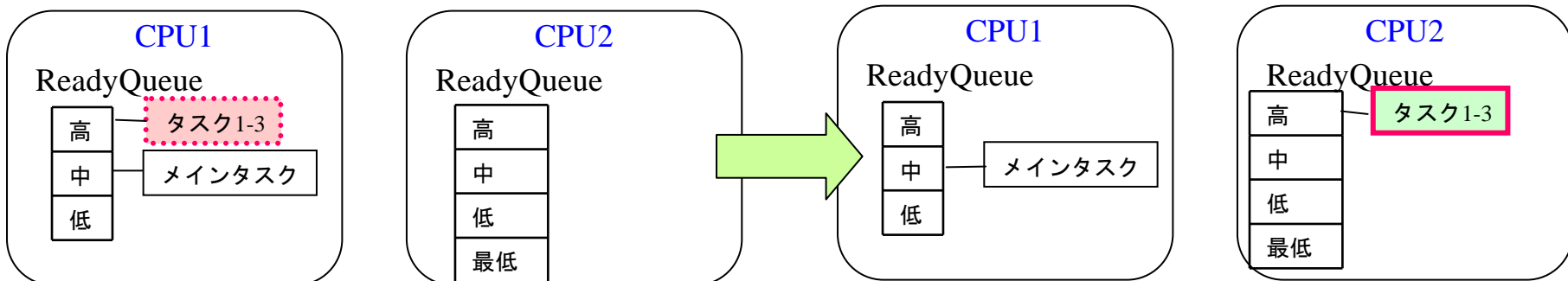
【6】 mig_tsk発行前から, mig_tsk終了までを測定



【7】 mig_tsk発行前から, 起動までを測定



【8】 mig_tsk発行前から, mig_tsk終了までを測定



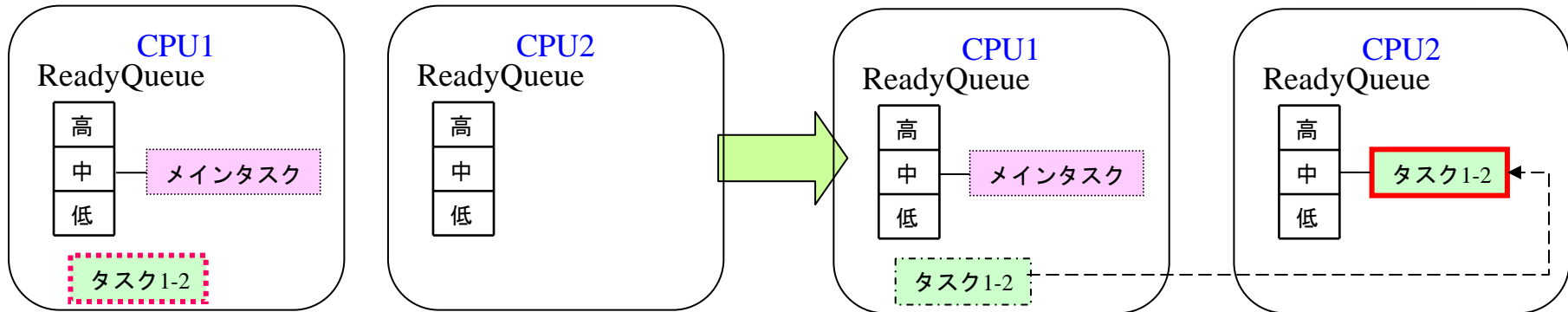
mact_tsk 性能評価：概要

- mact_tskの実行に関して次のパターンの測定を行う
 - ・【9】対象タスクが休止状態かつ、自プロセッサ所属のタスクを他プロセッサで起動
 - act_tsk 【2】 と類似処理
 - ・【10】対象タスクが休止状態かつ、他プロセッサ所属のタスクを自プロセッサで起動
 - act_tsk 【1】 と類似処理

対象タスクの状態	起動の結果	最高優先度	実行状態のタスクと 同優先度	実行状態のタスクより 低い優先度	レディキューが空
	対象タスクの所属				
パターン1 休止状態 (レディキューへ つなぐ作業発生)	パターンa 自→他				【9】
	パターンb 他→自	【10】			
	パターンc 他→他	実施せず	実施せず	実施せず	
パターン2 休止状態以外 (レディキューへ つなぐ作業なし)	パターンa 自→他	実施せず			
	パターンb 他→自	実施せず			
	パターンc 他→他	実施せず	実施せず	実施せず	実施せず

mact_tsk 性能評価：詳細

【9】 mact_tsk発行前から，起動まで



【10】 mact_tsk発行前から，起動まで

