



μITRON4.0 仕様の徹底解説
パート4
拡張機能と
自動車制御用プロファイル」

NECマイコンテクノロジー
宮内 哲夫

内容

- 拡張機能
 - ミューテックス (新機能)
 - メッセージバッファ
 - ランデブ
 - 可変長メモリプール
 - アラームハンドラ
 - オーバーランハンドラ (新機能)
- 自動車制御用プロファイル
 - 制約タスクの導入

ミューテックス

- 共有資源を使用する際にタスク間で排他制御を行うための機能
- μITRON4.0仕様で拡張同期・通信機能に新たに追加された

ミューテックスの概要 (1)

- タスクは排他制御する資源を使用する前に、ミューテックスをロック
- タスクは資源を使い終わったら、ミューテックスのロックを解除
- すでにロックされているミューテックスをロックしようとしたタスクはロックが解除されるまでミューテックスのロック待ち状態になる

ミューテックスの概要 (2)

- 優先度継承プロトコル、優先度上限プロトコルの導入
- ロックしたタスクがロック解除を行わねばならない
- タスク終了時に自動的にロック解除される

ミューテックス関連サービスコール

- CRE_MTX ミューテックスの生成 (静的 API)
- cre_mtx ミューテックスの生成
- acre_mtx ミューテックスの生成 (ID 番号自動割り付け)

- del_mtx ミューテックスの削除

ミューテックス関連サービスコール

- loc_mtx ミューテックスのロック
- ploc_mtx ミューテックスのロック (ポーリング)
- tloc_mtx ミューテックスのロック (タイムアウト有り)
- unl_mtx ミューテックスのロック解除
- ref_mtx ミューテックスの状態参照

キューテックスの属性

- TA_TFIFO FIFO 順
- TA_TPRI 優先度順
- TA_INHERIT 優先度継承プロトコル
- TA_CEILING 優先度上限プロトコル

ミューテックスの特徴 (1)

- TA_TFIFO 属性 ,TA_TPRI 属性のミューテックスはバイナリセマフォと似た振る舞いをする

ミューテックスの特徴 (2)

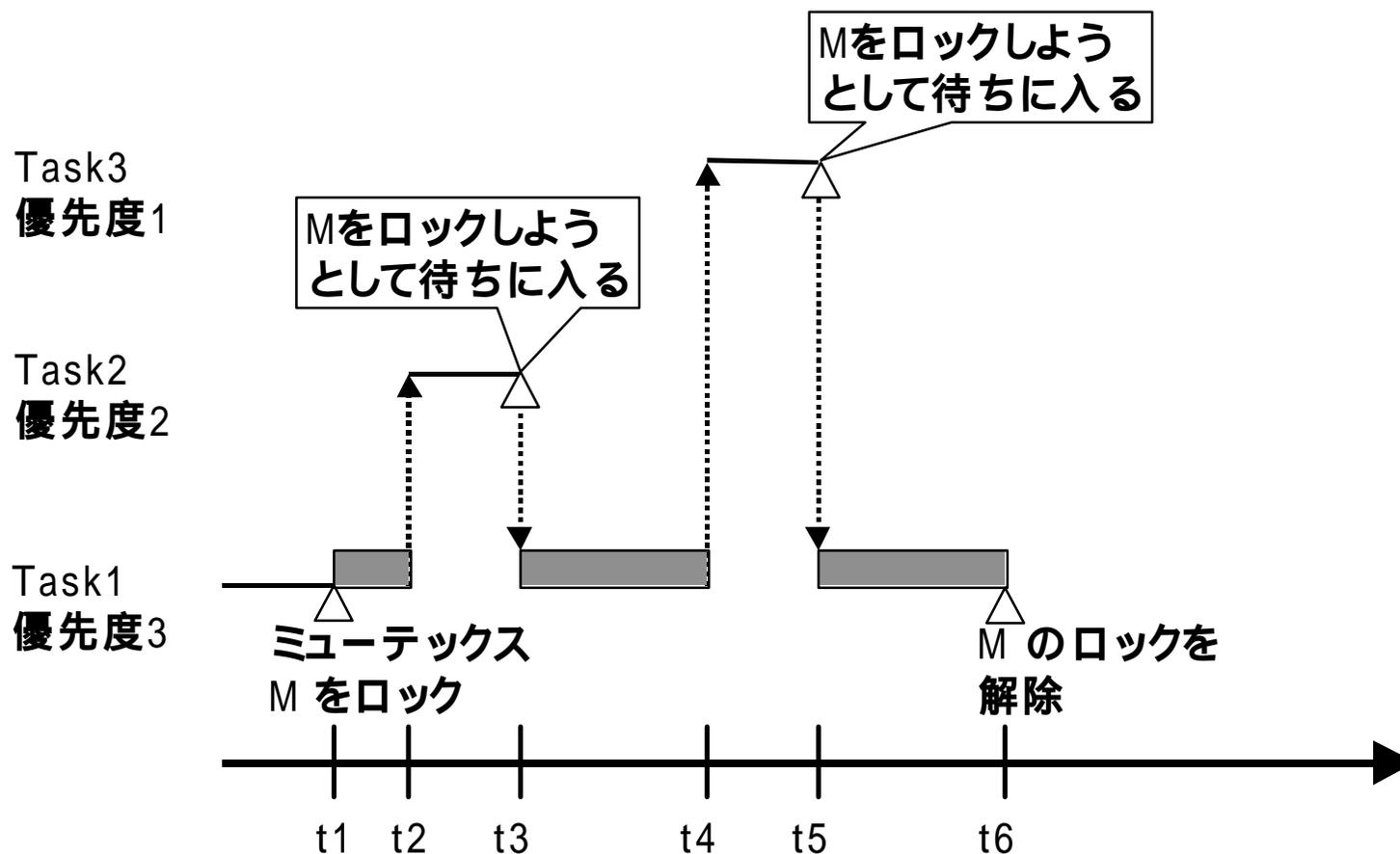
ミューテックスはバイナリセマフォと比べて以下のような特徴がある

- ロックしたタスクがロック解除しなければならぬ
- タスク終了時に自動的にロック解除される
- 上限のない優先度逆転現象の回避が可能

タスクの優先度

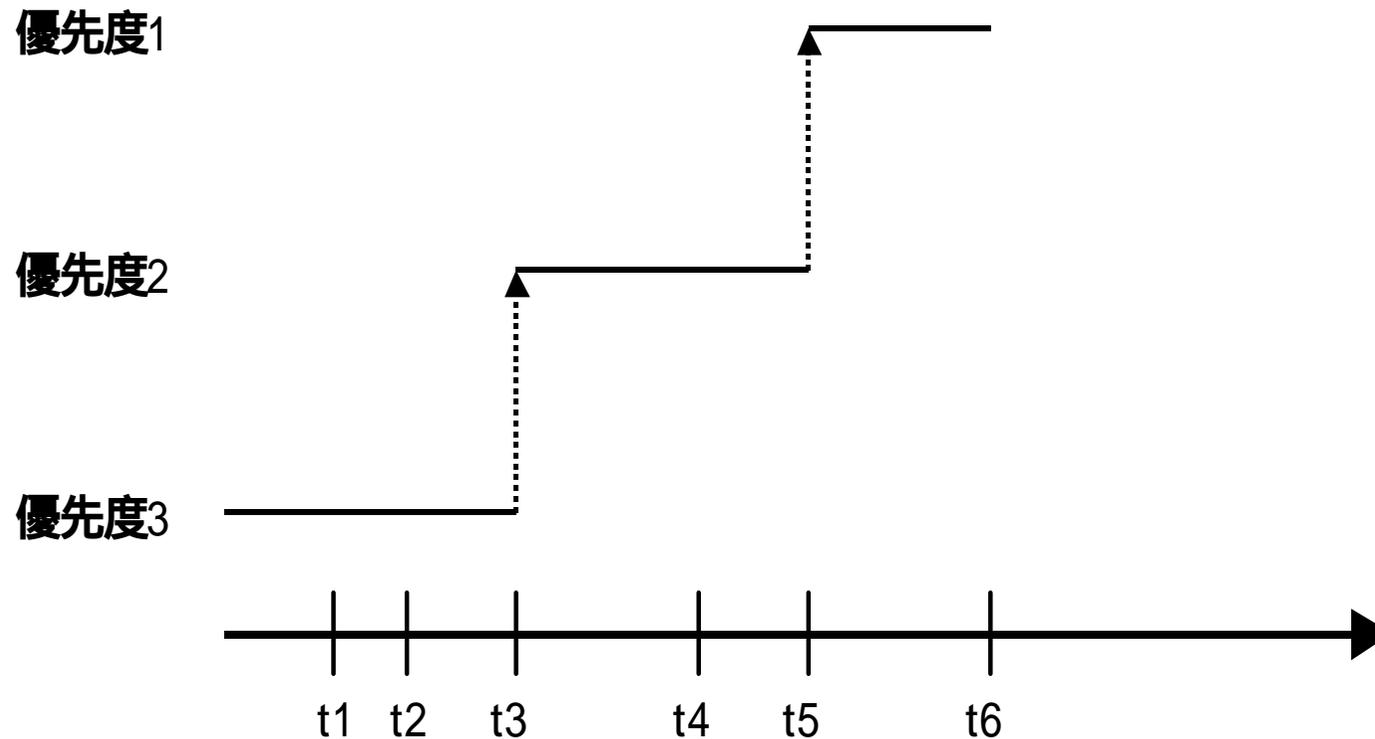
- 起動時優先度
- ベース優先度
- 現在優先度
 - ミューテックスを使用しない場合はタスクのベース優先度と現在優先度は常に一致
- chg_pri: タスクのベース優先度変更
- get_pri: タスクの現在優先度参照

優先度継承プロトコル (1)



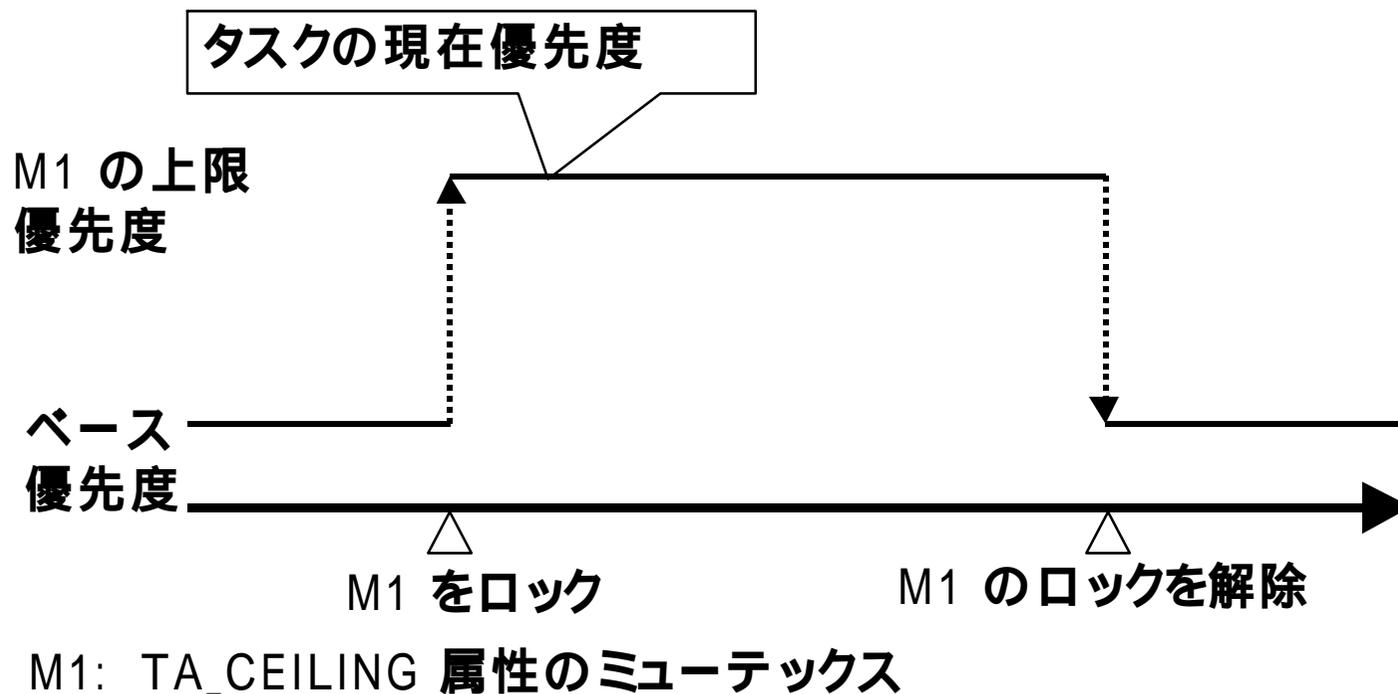
優先度継承プロトコル (2)

Task1 の現在優先度の変化



優先度上限プロトコル

TA_CEILING 属性のミューテックスをロックするタスクの
現在優先度の変化



優先度制御規則

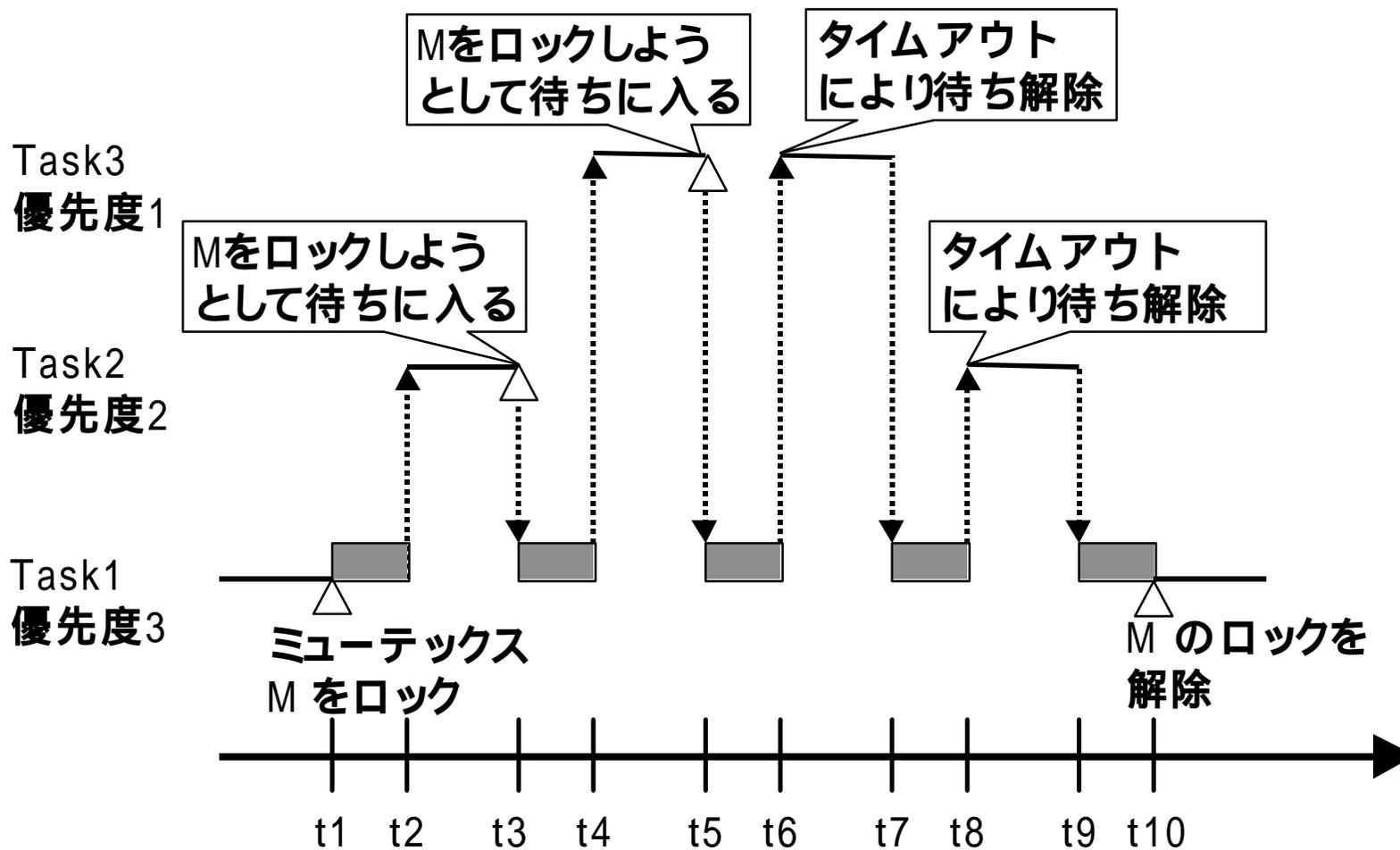
- 厳密な優先度制御規則
- 簡略化した優先度制御規則
 - どちらの優先度制御規則によるかは実装定義

厳密な優先度制御規則

- タスクの現在優先度は次に挙げる優先度の最高値と常に一致するようにする
 - タスクのベース優先度
 - タスクが TA_INHERIT 属性のミューテックスをロックしている場合、それらのミューテックスのロックを待っているタスクの中で、最も高い現在優先度を持つタスクの現在優先度
 - タスクが TA_CEILING 属性のミューテックスをロックしている場合、それらのミューテックスの中で、最も高い上限優先度を持つミューテックスの上限優先度

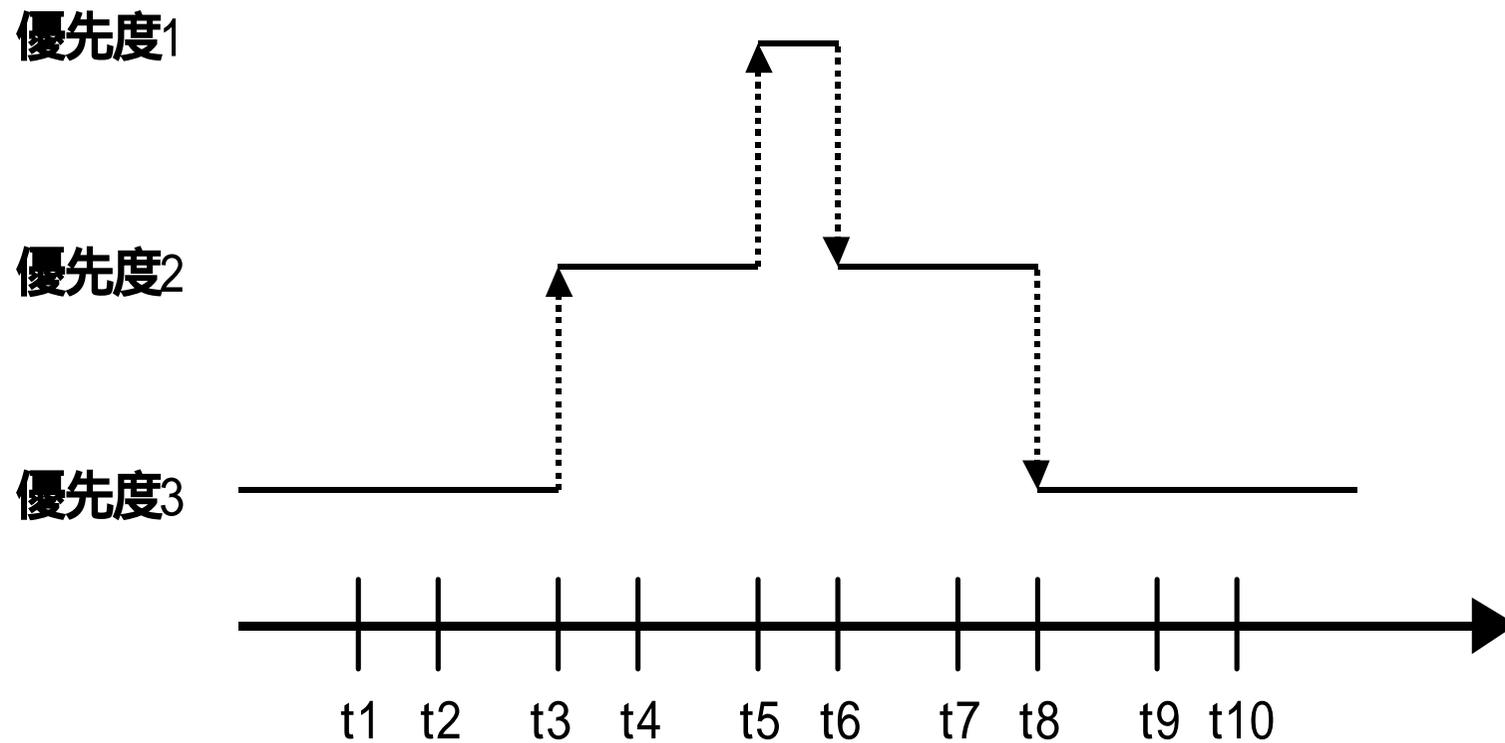
簡略化した優先度制御規則

- タスクの現在優先度を高くする方向の変更はすべて行う
- 現在優先度を低くする方向の変更は、タスクがロックしているミューテックスがなくなったときのみ行う



厳密な優先度制御規則に従う例

Task1 の現在優先度の変化



メッセージバッファ

新設されたサービスコール

- CRE_MBF

メッセージバッファの生成 (静的 API)

- acre_mbf

メッセージバッファの生成 (ID 番号自動割り付け)

μITRON3.0との違い

- 待ち行列につながれている順に送信
- TA_TPRI : μITRON3.0仕様では受信待ち行列が優先度順 μITRON4.0仕様では送信待ち行列が優先度順
- 生成情報に、メッセージバッファ領域の先頭番地を追加
- メッセージサイズの型の変更
- ref_mbfの内容

ランデブ

新設されたサービスコール

- CRE_POR

ランデブポートの生成 (静的 API)

- acre_por

ランデブポートの生成 (ID 番号自動割り付け)

- ref_rdv ランデブの状態参照

廃止

- tcal_por

μITRON3.0との違い

- tcal_por のタイムアウト値
- few_por で新たな呼び出しメッセージを保存する領域を限定
- 長さ0のメッセージ可能
- ポート ランデブポート
- RDVPTN, RDVNO 型の新設など型の変更,追加
- パラメータ,リターンパラメータの順序変更

可変長メモリプール

新設されたサービスコール

- CRE_MPL

可変長メモリプールの生成 (静的 API)

- acre_mpl

可変長メモリプールの生成 (ID 番号自動
割り付け)

μITRON3.0との違い

- サービスコールの名称を `get_blk`, `pget_blk`, `tget_blk`, `rel_blk` から `mpl` に統一するため `get_mpl`, `pget_mpl`, `tget_mpl`, `rel_mpl` にそれぞれ変更した
- 待ち行列につながれている順序でメモリブロックを獲得
- 生成情報、ブロックサイズの型等の変更
- `ref_mpl` の内容

アラームハンドラ

- 指定した時刻に起動されるタイムイベント
ハンドラ

新設されたサービスコール

- CRE_ALM, cre_alm (def_alm の名称を変更)
- del_alm
- sta_alm
- stp_alm

μITRON3.0との違い

- sta_alm (アラームハンドラ動作開始)の新設
静的にアラームハンドラを生成する場合を
考慮
- 起動時刻を絶対時刻で指定する機能の廃止
- ref_alm でアラームハンドラの動作状態を参照
できるようにした
- 生成情報、参照情報の型、順序等の変更

オーバーランハンドラ

- 時間管理機能に μITRON4.0 仕様で新しく追加された機能
- タスクが設定された時間を越えてプロセッサを使用した場合に起動されるタイムイベントハンドラ

オーバーランハンドラ

- システムにひとつだけ定義できる
- タスク毎にタスクの上限プロセッサ時間をオーバーランハンドラの起動条件として設定
- 設定されたタスクに対して、そのタスクの使用プロセッサ時間が上限プロセッサ時間を越えると、オーバーランハンドラが起動される
- オーバーランハンドラには、起動の原因となったタスクの ID 番号と拡張情報が渡される

オーバーランハンドラの記述形式

```
void ovrhdr(ID tskid, VP_INT exinf)
{
    /* オーバーランハンドラ本体 */
}
```

オーバーランハンドラ関連 サービスコール

- DEF_OVR オーバーランハンドラの定義
(静的 API)
- def_ovr オーバーランハンドラの定義
- sta_ovr オーバーランハンドラの動作開始
- stp_ovr オーバーランハンドラの動作停止
- ref_ovr オーバーランハンドラの状態参照

自動車制御用プロファイル

- 自動車制御応用を主なターゲットとした
μITRON4.0仕様のプロファイル規定のひとつ
- スタダードプロファイルと下位互換性をもつ
- 機能のサブセット化 ,追加
- 制約タスクの導入

サポートされない機能

- タイムアウト付きのサービスコール
- タスクの優先度順の待ち行列
- 強制待ち状態
- タスク例外処理機能
- メールボックス
- 固定長メモリプール
- その他、いくつかのサービスコール

制約タスク

- 待ち状態に入ることができない
- 優先度を変更することができない
- サービスコールによりタスクを終了することができない (メインルーチンからのリターンで終了)

タスク管理機能

- CRE_TSK
- act_tsk/iact_tsk
- can_act
- ext_tsk
- ter_tsk
- chg_pri
- get_pri

タスク付属同期機能

- slp_tsk
- wup_tsk/iwup_tsk
- can_wup
- rel_wai/irel_wai

同期・通信機能 (セマフォ)

- CRE_SEM
- sig_sem/isig_sem
- wai_sem
- pol_sem

同期 通信機能 (イベントフラグ)

- CRE_FLG
- set_flg/iset_flg
- clr_flg
- wai_flg
- pol_flg

同期 通信機能 (データキュー)

- CRE_DTQ
- psnd_dtq/ipsnd_dtq
- fsnd_dtq/ifsnd_dtq
- rcv_dtq
- prcv_dtq

時間管理機能

システム時刻管理

- isig_tim

周期ハンドラ

- CRE_CYC
- sta_cyc
- stp_cyc

システム状態管理機能

- get_tid/iget_tid
- loc_cpu/iloc_cpu
- unl_cpu/iunl_cpu
- dis_dsp
- ena_dsp
- sns_ctx
- sns_loc
- sns_dsp
- sns_dpn

割り込み管理機能

- DEF_INH (ATT_ISR)

システム構成管理機能

- DEF_EXC
- ATT_INI

機能が追加/限定 (1)

- CRE_TSK
 - TA_RSRT (制約タスク)を指定可
- CRE_SEM, CRE_FLG, CRE_DTQ
 - TA_TPRI (優先度待ち)はサポート不要
- ext_tsk
 - 制約タスクから呼び出された場合 未定義

機能が追加/限定 (2)

- `ter_tsk`, `chg_pri`
 - 制約タスクから呼び出された場合 未定義
- `slp_tsk`, `wai_sem`, `wai_flg`, `rcv_dtq`
 - 制約タスクから呼び出された場合 未定義