ITRON Newsletter No.11

(社) トロン協会 ITRON 専門委員会

〒 108 東京都港区三田 1 丁目 3 番 3 9 号　勝田ビル 5 階

TEL: (03) 3454-3191　　FAX: (03) 3454-3224

ITRON 関連書籍の一覧

1994 年 8 月 1 日時点で、ITRON 専門委員会が編集し、発行されている ITRON 関連の書籍は別表の通りです。ITRON・μITRON 標準ハンドブックは、μITRON (Ver 2.0) と ITRON2 の仕様書を 1 冊にまとめたものです。ご希望の方は、各発売元にお問い合わせ下さい。

μITRON3.0 仕様の最新バージョンは、Ver 3.02.00 です。μITRON3.0 標準ハンドブック (Ver 3.00.00) からの改訂点は、ITRON ニュースレター No.5 および本号に掲載されています。

ITRON 標準ガイドブック'92-'93 は、タイトルの 1992 〜 93 年という期間を過ぎていますが、μITRON 仕様 (Ver 2.0) や ITRON2 仕様を使われている場合には、現在でも有効に活用できます。現在、ITRON 標準ガイドブックの新版を、μITRON3.0 仕様をメインのターゲットとして編集作業中です。

μITRON3.0 仕様のバージョンアップ

μITRON3.0 仕様を別表の通り改訂します。いずれも、μITRON3.0 標準ハンドブックを発行して以降に発見された問題点の修正です。この改訂を行った仕様のバージョン番号は、Ver 3.02.00 とします。

フリーソフトウェア紹介

ここでは、市販されている製品ではないために、ITRON 関連製品登録制度に登録されない ITRON 関連のフリーソフトウェアについて、簡単な紹介をします。

ASURA-I

木元峰之

ASURA-I は、MS-DOS に常駐して機能する、μITRON3.0 仕様 OS です。デバイスドライバとして組み込みますので、通常のアプリケーションはもちろん、TSR や他のデバイスドライバからでも μITRON のシステムコールを利用することができます。現在は DOS 汎用ということでタイマー関係を実装していませんが、将来的には対応したい考えです。

ASURA-I のもう一つの特徴は、ソースコードの記述に、BTRON の実身仮身ネットワークを採用している点です。DOS 用のアーカイブには入っていませんが、暫定公開ドキュメントである「inside of ASURA-I」には実身仮身形式のソースリストが収められています。

付属のシステムアプリケーションとして、各オブジェクトの状態表示、簡易 CLI ワークエリアテスタがあり、アセンブラ用のインクルードファイルもはいっています。

なお ASURA-I は、NIFTY-Serve のパーソナルメディアフォーラム (FPMC) と、インターネットの ftp サーバ utsun.s.u-tokyo.ac.jp [133.11.11.11] と ftp.waseda.ac.jp [133.9.1.4] にて公開しています。

文献紹介

情報処理 (情報処理学会 学会誌) の 1994 年 10 月号 (Vol.35 No.10) で、「TRON プロジェクトの現状と展望」というタイトルの特集が組まれています。ITRON サブプロジェクトに関する記事としては、東京大学の高田広章氏、(株) 東芝の田丸喜一郎氏らの ITRON 専門委員会メンバの共著による「ITRON サブプロジェクトの現状と展望 −カーネル仕様とその実装技術−」というタイトルの解説を掲載しています。

この解説では、まず、小規模組み込みシステムの現状、ITRON 仕様の設計ポリシーとその概要について紹介した後、ITRON 仕様 OS の実装・応用状況を報告し、標準化によってどういうメリットが生じたかについてが議論されています。次に、ITRON サブプロジェクトを通じて研究開発されたリアルタイムカーネルの実装技術が技術的な視点から紹介されています。また、分散システムをサポートするための新しい仕様である μITRON3.0 についても解説されており、最後に今後の展望が述べられています。全体に、ITRON サブプロジェクトについての知識を前提としないように書かれており、ITRON について詳しく知らない方にもお勧めできる解説記事となっています。

† このニュースレターは、TRONWARE vol.30 および TRON PROJECT BIMONTHLY No.35 に掲載されたものです。

ITRON 関連書籍一覧

書籍名	分類	価格	発売元	ISBN 番号
ITRON・μITRON 標準ハンドブック	和文仕様書	4,800 円	パーソナルメディア	ISBN4-89362-079-7
μITRON3.0 標準ハンドブック	和文仕様書	4,000 円	パーソナルメディア	ISBN4-89362-106-8
ITRON/FILE 標準ハンドブック	和文仕様書	3,000 円	パーソナルメディア	ISBN4-89362-092-4
ITRON 標準ガイドブック'92-'93	和文参考書	3,500 円	パーソナルメディア	ISBN4-89362-197-6
μITRON Specification Ver 2.01.00.00	英文仕様書	12,000 円	トロン協会	–
ITRON2 Specification Ver 2.02.00.10	英文仕様書	15,000 円	トロン協会	–
μITRON3.0 Specification Ver 3.00.00	英文仕様書	–	トロン協会	–

※ 価格には消費税を含みません。

※ トロン協会発売の書籍には、トロン協会会員向け価格が設定されています。

※ 英文仕様書については、インターネット上での無料配布を行っています。

インターネット経由で仕様書をダウンロードする方法については、ITRON ニュースレター No.8 をご覧下さい。

μITRON3.0 仕様の改訂点 (Ver 3.01.00 → Ver 3.02.00)

1. 「システムコール索引」中の cre_sem のページ数を以下の通りに修正 (p.VIII)

   誤: cre_sem [EN] セマフォ生成 11

   正: cre_sem [EN] セマフォ生成 111

2. 「図 3. ITRON のタスク状態遷移図」に、以下の但し書きを追加 (p.23)

   ※ この状態遷移図は、あくまでも代表的な状態遷移のみを図示したもので、インプリメントによってはこの図にない遷移が起こる場合もある。

3. p.27 の下から 2 行目の記述を以下の通りに修正 (p.27)

   誤: 「…割込み A の②〜③の部分の実行が…」

   正: 「…割込み X の②〜③の部分の実行が…」

4. 「1.6 ITRON の基本概念」中の「タスク部と準タスク部」の節の次に以下の節を追加 (p.29)

   ディスパッチ遅延中のタスク状態

   ITRON では、遅延ディスパッチの原則や、ディスパッチの禁止を行うシステムコールの発行により、一時的にディスパッチが禁止されることがある。その間のタスク状態については、次のように考える。

   ディスパッチが禁止されている間には、実行中のタスクがプリエンプトされるべき状況となっても、新たに実行すべき状態となったタスクにはディスパッチされない。実行すべきタスクへのディスパッチは、ディスパッチ禁止状態が解除されるまで遅延される。ディスパッチが遅延されている間は、実行中のタスクが RUN 状態であり、ディスパッチ禁止状態が解除された後に実行すべきタスクは READY 状態であると扱う。なお、ディスパッチが遅延されている間に、RUN 状態のタスクがレディキューの該当優先度のどの位置にあると扱うかは、インプリメント依存である。

   また、ディスパッチが禁止されている間に、実行中のタスクに対して sus_tsk を発行して SUSPEND 状態へ移行させようとした場合および ter_tsk を発行して DORMANT 状態へ移行させようとした場合、タスク状態の遷移はディスパッチ禁止状態が解除されるまで遅延される。状態遷移が遅延されている間は、実行中のタスクは過渡的な状態にあると考え、その具体的な扱いはインプリメント依存とする。一方、ディスパッチ禁止状態が解除された後に実行すべきタスクは、READY 状態であると扱う。

5. dis_dsp の解説中のディスパッチ禁止状態の間の動作の 1 番目を以下の記述に差し替え (p.74)

   – 割込みハンドラあるいは dis_dsp を実行したタスクから発行されたシステムコールによって、dis_dsp を実行したタスクがプリエンプトされるべき状況となっても、新たに実行すべき状態となったタスクにはディスパッチされない。新たに実行すべき状態となったタスクへのディスパッチは、ena_dsp によってディスパッチ禁止状態が解除されるまで遅延される。

μITRON3.0 仕様の改訂点 (Ver 3.01.00 → Ver 3.02.00) (続き)

6. dis_dsp の解説中のディスパッチ禁止状態の間の動作の 2 番目に以下の記述を追加 (p.74)

   − ディスパッチ禁止状態で起動された割込みハンドラ中から、実行中のタスク (dis_dsp を実行したタスク) に対して sus_tsk を発行して SUSPEND 状態へ移行させようとした場合および ter_tsk を発行して DORMANT 状態へ移行させようとした場合、タスク状態の遷移はディスパッチ禁止状態が解除されるまで遅延される。

7. rot_rdq の解説中の第 2 段落を以下の記述に差し替え (p.81)

   タスク部から発行される rot_rdq では、tskpri=TPRI_RUN=0 により、自タスクの持つ優先度のレディキューを回転させる。

8. rot_rdq の解説中の最後の段落を補足事項とする。また、rot_rdq の補足事項として以下の記述を追加 (p.82)

   インプリメントによっては、周期起動ハンドラなどのタスク独立部から rot_rdq(tskpri=TPRI_RUN) を発行することも可能で、その場合には、その時実行中のタスクを含むレディキューないしはレディキュー中の最高優先度のタスクを含むレディキューを回転させる。通常、この 2 つは一致しているが、タスクのディスパッチが遅延されている状態においては、一致しない場合がある。この場合に、実行中のタスクを含むレディキューを回転させるか、最高優先度のタスクを含むレディキューを回転させるかは、インプリメント依存である。なお、この機能は互換性や接続性の保証されない拡張機能 [レベル X] であり、常に利用できるとは限らない。

9. rsm_tsk の仕様決定の理由の第 3 段落の第 1 文を以下の記述に差し替え (p.100)

   仕様面では、あるタスクが SUSPEND 状態になっても、プリエンプトされた場合と同様に、タスクスケジューリングの順序には影響しないのが望ましい。

   rsm_tsk の仕様決定の理由の第 4 段落の 2 文目に以下の記述を追加 (p.100)

   以下でレディキューといった場合には、インプリメント上のレディキューのことをいい、仕様を定義する上での (RUN 状態のタスクと READY 状態のタスクのみを含んだ) 概念上のレディキューとは区別して考える。

10. can_wup の解説中の第 2 段落を以下の記述に差し替え (p.105)

    tskid=TSK_SELF=0 によって自タスクの指定を行うことができる。ただし、タスク独立部から発行したシステムコールで tskid=TSK_SELF=0 を指定した場合には、E_ID エラーとなる。

11. del_sem の補足事項として以下の記述を追加 (p.115)

    [補足事項]
    複数のタスクが待っているセマフォを削除した場合に、待ち解除後のレディキュー中のタスクの順序は、同じ優先度を持つタスクの間ではインプリメント依存である。

    同様の補足事項を、del_flg, del_mbx, del_mbf, del_por, del_mpl, del_mpf にも追加する。

12. set_flg の解説の最後の段落の 4 文目に以下の記述を追加 (p.132)

    この場合に、待ち解除後のレディキュー中のタスクの順序は、同じ優先度を持つタスクの間では元のイベントフラグの待ち行列の順序を保存する。

13. snd_msg のエラーコードの説明中の E_PAR エラーの説明を以下の通りに修正 (p.149)

    E_PAR　　パラメータエラー (pk_msg が使用できない値, メッセージヘッダ中のパラメータ (msgpri など) が不正)

14. def_int の解説中の第 6 段落の第 1 文を以下の記述に差し替え (p.212)

    割込みハンドラの中でシステムコールを発行することにより、それまで RUN 状態 (実行状態) であったタスクがその他の状態に移行し、代わりに別のタスクが RUN 状態となるべき状況となっても、割込みハンドラ実行中はディスパッチ (実行タスクの切り替え) が起こらない。

    同様の修正を、def_cyc, def_alm の解説に対しても行う。

μITRON3.0 仕様の改訂点 (Ver 3.01.00 → Ver 3.02.00) (続き)

15. loc_cpu の解説中の割込みおよびディスパッチ禁止状態の間の動作の 2 番めを以下の記述に差し替え (p.220)

    − loc_cpu を実行したタスクから発行されたシステムコールによって、loc_cpu を実行したタスクがプリエンプトされるべき状況となっても、新たに実行すべき状態となったタスクにはディスパッチされない。新たに実行すべき状態となったタスクへのディスパッチは、unl_cpu によってディスパッチ禁止状態が解除されるまで遅延される。

16. def_cyc システムコールの補足事項中に以下の記述を追加 (p.275)

    タイマハンドラから def_cyc を呼び出せるインプリメントでは、ハンドラ内で同じ番号の周期起動ハンドラを再定義することができる。

17. Ver 3.01.00 への改訂で、def_alm システムコールの補足事項に追加した記述を、以下の記述に差し替え (p.285)

    アラームハンドラが起動された時点では、そのハンドラの定義はすでに解除されているものとする。つまり、ハンドラ内で ref_alm により起動されたハンドラの情報を参照すると、E_NOEXS エラーとなる。また、タイマハンドラから def_alm を呼び出せるインプリメントでは、ハンドラ内で同じ番号のアラームハンドラを再定義することができる。

18. get_ver の解説中の p.295 の上から 3 行目の記述を以下の通りに修正 (p.295)
    誤: 「… ITRON と BTRON で共通である。…」
    正: 「… ITRON、μITRON と BTRON で共通である。…」

19. ref_sys の解説中の 1 行目の記述を以下の通りに修正 (p.298)
    誤: 「… CPU や μITRON3.0 の実行状態…」
    正: 「… CPU や OS の実行状態…」

20. def_svc の解説中の p.303 の上から 8 行目の記述を以下の通りに修正 (p.303)
    誤: 「− システムコールの不可分性」
    正: 「− 拡張 SVC ハンドラの不可分性」

21. 5.2 中の高級言語対応ルーチンの節の最後に、以下の記述を追加 (p.356)

    なお、言語 C の関数で記述したルーチンが、アセンブラで記述したルーチンと同じインタフェースで呼び出せる場合には、TA_HLNG 指定を行った場合にも、必ずしも高級言語対応ルーチンを使用する必要はない。

22. 「6.4 データタイプ」中の「汎用的なデータタイプ」の節の型定義の部分を以下の記述に差し替え (p.376)

    B　符号付き 8 ビット整数
    H　符号付き 16 ビット整数
    W　符号付き 32 ビット整数
    UB　符号無し 8 ビット整数
    UH　符号無し 16 ビット整数
    UW　符号無し 32 ビット整数

    VB　データタイプが一定しないもの (8 ビットサイズ)
    VH　データタイプが一定しないもの (16 ビットサイズ)
    VW　データタイプが一定しないもの (32 ビットサイズ)
    VP　データタイプが一定しないものへのポインタ

    FP　プログラムのスタートアドレス一般

23. p.384 の節タイトルを以下の通りに修正
    誤: 「6.5 言語 C インタフェース一覧」
    正: 「6.6 共通定数と構造体のパケット形式」