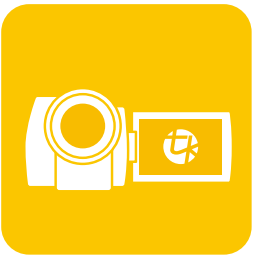
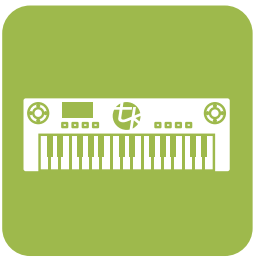


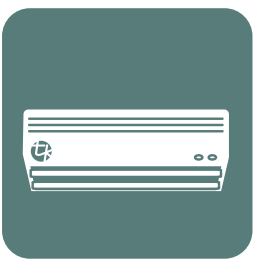
T-Kernel 組込みプログラミング 強化書®

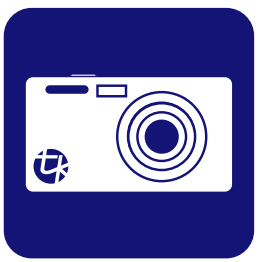
坂村健 監修
パーソナルメディア株式会社 編著



開発現場ですぐに役立つ実践テクニック








パーソナルメディア

本書の著作権に関するご案内

パーソナルメディア株式会社
http://www.personal-media.co.jp/

本書は、以下の購入者にのみ使用・閲覧の権利が与えられています。

本書の書名：『T-Kernel 組込みプログラミング強化書』
購入者：
メールアドレス：
購入日時：
購入時の IP アドレス：

連絡先

パーソナルメディア株式会社 出版部
pub@personal-media.co.jp

目　次

コラム

例題

監修のことば

サステナビリティ（持続可能性）の高い社会を構築するためには、快適さや便利さを維持しながらエネルギー効率を高めることが必要である。快適さや便利さを犠牲にした努力には限界があり、その意味で持続可能性につながらないからだ。

そのためには、21 世紀の我々の生活環境を構成する身の回りの多くの電子機器や住宅設備を、機能向上するとともに最適制御することが重要である。過去の時代と比べ電子機器や住宅設備などが、身の回りに飛躍的に増え生活にかかせないモノとなっているからである。このような機器・設備の最適制御のために重要な役割を担うのがそれらを制御する組込みコンピュータである。

組込みコンピュータのソフトウェアの開発には、制御対象機器の多様性、少ない計算リソース、高度に要求されるリアルタイム性や信頼性といった点で、パソコンやサーバ用のソフトウェアとは違った難しさがある。このような組込みシステム開発の困難を緩和し、開発効率を向上させることを目的として、私は組込みシステム開発のオープン·プラットフォーム構築のための T-Engine プロジェクトを開始した。

プロジェクト開始から 5 年たち、開発ハードウェアプラットフォームである T-Engine ボードは組込み用の大多数の CPU に対応した製品がリリースされ、中核となるオープンでフリーなリアルタイム OS「T-Kernel」のソースプログラムも数多くの開発者によりダウンロードされ利用されている。その結果、プロジェクト開始からわずかの時間で多くの T-Kernel 採用製品が生まれてきている。

大規模な組込みシステムを効率よく開発するためには、プログラムのモジュール化を行い、個々のモジュールの独立性や再利用性を高めることが重要である。移植作業を可能な限り減らすことは、単に工数削減というだけでなくエンバグの可能性をも減らす。T-Kernel ではこのために開発プラットホームの標準アーキテクチャを規定しており、この T-Engine アーキテクチャに準拠するターゲットハードウェアを用意することで多くのモジュールを再コンパイルの手間だけで移植できる。また、デバイスドライバやサブシステムなどモジュール化のための機能が用意してあり、大規模な組込みシステムを効率よく開発できる。このしくみを使った、T-Kernel 用のデバイスドライバ、ミドルウェアはすでに多数流通しており、これらを組み合わせることによりシステムの開発期間が大きく短縮される。

さらに、ファイルシステムを持つような高度な組込システムのためにサブシステムの機能を使って実装されたミドルウェア群――T-Kernel Extension も T-Engine フォーラムからダウンロード可能となっている。とくに、T-Kernel Extension によって実現されるプロセスの機能は、プログラムのモジュール化によるシステムの信頼性を向上に貢献し、大規模で複雑な組込みシステムの開発には必要不可欠になってきている機能である。

T-Kernel を利用して組込み機器の開発効率を上げるには、デバイスドライバ、サブシステム、プロセス等のしくみをよく理解し、それらの機能を駆使することが重要であるが、本書は T-Kernel のノウハウを知り尽くした現場の技術者により執筆された実践的な解説書であり、どのように組込みシステムを開発するのかが具体的にわかる――まさに、組込みシステムの設計者・プログラマ必携の書である。

本書が T-Kernel の理解や普及に役立つことはもちろん、組込み技術者の不足という問題解消の一助となり、その結果として多くのすぐれた組込み機器が実現され、サステナブルな社会の構築に多少なりとも貢献できれば、これに勝る喜びはない。ぜひ 1 人でも多くの技術者に本書を活用していただければと思う。

2007 年 12 月

坂村 健

■本書のねらい

約 500 社が参加し、組込み業界最大の団体とも言える T-Engine フォーラムは、組込みシステムの開発の効率化を目的とした標準化活動を推進しているが、その一環として、T-Kernel および T-Kernel Extension の仕様を策定し、その実装（プログラム）とともに公開している。T-Kernel は組込みシステム用 OS である ITRON の流れを継承する次世代の組込み向けリアルタイム OS であり、T-Kernel Extension は T-Kernel にプロセスやファイルなどの拡張機能を付加する基本的かつ重要なミドルウェアである。

大規模な組込みシステムを効率よく開発するには、単にリアルタイム OS を導入して、アプリケーションをマルチタスク化するだけでは十分ではない。各ソフトウェアモジュールの独立性を高めて見通しをよくし、再利用性を高めるには、タスクによって実現されるより上位の機能についても、標準的な枠組みを設け、各モジュールの汎用性を高めていくことが重要である。こういった「上位の枠組み」に相当する機能は、一般にはミドルウェアと呼ばれるが、T-Kernel においては、デバイスドライバおよびサブシステムといった T-Kernel/SM（System Manager）の機能によりこの枠組みを実現しており、その代表的かつ最も重要な実現例が T-Kernel Extension である。T-Kernel、T-Kernel Extension が、ITRON をはじめとするリアルタイム OS の長年に渡る蓄積と実績を基礎としながらも、単なるリアルタイム OS ではなく、大規模な組込みシステムの開発にも効率よく対応できるのは、デバイスドライバやサブシステムといった「上位の枠組み」を備えているからにほかならない。

本書は、この T-Kernel や T-Kernel Extension を用いたシステムの開発について、豊富なプログラミング例やノウハウを交えて詳述した、開発者向けの実践的な解説書である。本書にはリアルタイム OS である T-Kernel 自体の解説も含んでいるが、それだけではなく、デバイスドライバやサブシステム、T-Kernel Extension やその上でユーザーの開発するプロセスなど、T-Kernel を応用したシステムの全領域のプログラミングを対象としている。デバイスドライバやサブシステムなど「上位の枠組み」を利用することが T-Kernel のメリットであることを考えると、むしろ、後者の方に本書のより大きな意義があると言えよう。

ぜひ本書を通して T-Kernel を使った組込みシステムのソフトウェア開発のノウハウを習得し、実際の製品開発にもご活用いただきたい。

■本書の特色

前項のねらいを具体化しつつ本書の執筆を行うにあたり、特に次の 3 点に留意した。

1.T-Kernel と T-Kernel Extension の両方にわたって総合的に解説すること

T-Kernel の応用システムでは、T-Kernel 上に開発するデバイスドライバやサブシステムと、T-Kernel Extension 上に開発する制御用アプリケーションを組み合わせて利用するのが一般的である。詳しくは本文に記すが、T-Kernel 上に開発するプログラムと T-Kernel Extension 上に開発するプログラムにはそれぞれのメリットとデメリットがあり、特に大規模な組込みシステムにおいては、両者をバランスよく利用して性能や開発効率を高めることが重要である。すなわち、組込みの世界ではこの 2 つがまさに車の両輪であって、実際の開発現場では両方に縦横にまたがるプログラミング技術が必要とされる。

しかしながら、T-Kernel と T-Kernel Extension では仕様書も別々であるし、仮想記憶か実記憶かといったメモリモデルも異なっている。そもそも、T-Kernel は純粋なリアルタイム OS であり、μITRON との比較など制御系 OS の文脈で語られるのに対し、T-Kernel Extension は機能的には UNIX や Windows のカーネル（GUI を除く部分）に相当しており、情報系 OS の文脈で語られることが多い。両者は別の世界の OS のように扱われる場合もあり、今まで両者の関係が具体的に説明されることは少なかった。

こういった背景から、本書では T-Kernel と T-Kernel Extension を常に対比し、両者の類似点や相違点に注目しながら、総合的に解説するように努めた。

本書が橋渡しとなって、μITRON などの制御系リアルタイム OS をやってこられた方が仮想記憶やプロセスといった T-Kernel Extension 上に開発するアプリケーションを体験されたり、UNIX など情報系 OS のアプリケーション開発をやってこられた方がデバイスドライバなどリアルタイム処理の世界を体験されることを期待している。

2. 実際の開発にすぐに役立つように、具体的な使い方を解説すること

T-Kernel の応用システムを開発する場合、まず参照するのは T-Kernel や T-Kernel Extension の仕様書であろう。T-Kernel の仕様書は『T-Kernel 標準ハンドブック 改訂新版』として出版されているし、T-Kernel Extension の仕様書は T-Engine 開発キットなどの付属ドキュメントに含まれているほか、T-Engine フォーラムからも入手できる。

しかし、これらの仕様書は、あくまでも仕様書として正確な情報を提供するという位置付けから、具体的なプログラミング例が掲載されているわけではない。本書では、この点を補うべく、実践的なプログラミング方法を数多く示すことに努めた。特に、内容を基本的な話題に限定せず、開発を担当する第一線のエンジニアにすぐ役立つような高度な内容、たとえば USB のドライバ開発や非同期のドライバの実現方法なども含めて解説した。一方、重要な機能をわかりやすく解説するという観点から、仕様書に書かれているすべての機能を網羅的に説明しているわけではないため、T-Kernel や T-Kernel Extension のすべての機能を知るには、仕様書を合わせてご覧いただく必要があるのでご注意いただきたい。

T-Kernel の応用システムのソフトウェアを開発される方は、本書を常に手元に置き、「こんなことがやりたいがどうすればよいか？」という疑問が生じたときに随時参考にしていただければ幸いである。

3. 初学者の教科書としても使えること

近年の組込み分野の需要の高まりに応じて、新たに組込みの世界に入られる方や、組込み技術者を目指して勉強される方もますます増えている。

本書では、そういった組込みシステムの初心者の方々でも理解できるように、リアルタイム OS の基本的な機能であるタスクやセマフォなどの概念についてもひととおりの説明を行うようにした。また、これらの基本的な事項については適宜例題を用意して、読者自身が手を動かして習得を確認できるように努めたほか、特につまずきやすい点については注意を喚起する説明を随所に挿入した。さらに、巻末には用語集をつけて、読者の理解を助けるように配慮した。本書を読むために必要な知識は、C 言語によるプログラミングの基本的な知識のみとなっているはずである。

T-Kernel や T-Kernel Extension は、業務用端末やカーナビといった実際の組込み製品の中にもすでに多数使われている OS であり、実践的という意味からも、リアルタイム OS の教材として最適なものである。本書を参考にして、実際に T-Kernel や T-Kernel Extension 上のプログラミングを行い実機上で動作確認しながら、組込みのプログラミング技術を習得していただきたい。

■本書の読み方

本書は、組込みシステムの初心者の方から、開発現場の第一線で活躍されているエンジニアの方まで、幅広い読者を対象としている。主として第 I 部で初心者向けの話題を扱い、第 II 部、第 III 部で実践的な応用に役立つ高度な話題を扱っている。具体的な読者層と各部の構成を次に示すので、本書購読のヒントとしていただきたい。

●組込みシステム開発の第一線で活躍中のエンジニアの方に対するガイドブックとして

すでに T-Kernel や T-Kernel Extension に関してひととおりの知識がある方や、使用経験のある方は、必ずしも本書を最初から読む必要はない。主に本書の第 II 部、第 III 部から、必要に応じて個別の話題を参照していただくのが効率的であろう。

とはいえ、一般的なリアルタイム OS や ITRON には含まれず、T-Engine や T-Kernel において新たに加わった重要な概念が存在する。そこで、第 I 部の基本事項においても、各節の終わりにある「まとめ」の部分には、ぜひともひととおり目を通して、T-Kernel や T-Kernel Extension の基礎知識や全体像をご確認いただくことをお勧めしたい。

●専門学校や大学、企業の新人研修等における組込みシステム入門テキストとして

T-Kernel、T-Kernel Extension を使った組込みシステムの入門テキストとして使用する場合は、まず第 I 部を順に読み進めて、リアルタイム OS や組込みシステムに関する基礎

的な概念をしっかりと習得していただきたい。

さらに、随時掲載されている例題やプログラミングの実習に自らチャレンジし、実際に手を動かしながら、重要な概念を身につけていくことをお勧めしたい。

その上で、第 II 部、第 III 部の応用的な話題から、適宜興味のある話題を選択していただきたい。

本書はパーソナルメディアにて T-Engine の開発やサポート業務、セミナー業務などを担当している技術者が、その数多くの経験を活かして執筆した、T-Kernel を使いこなすためのノウハウの集大成である。

本書を開発の第一線でご活用いただくとともに、今後社会的にますます需要の高まる組込み技術者の裾野を広げるために本書が役立てば、これに勝る喜びはない。

パーソナルメディア株式会社

第 I 部　基礎編

第 I 部では、アプリケーションを開発する上でもデバイスドライバやサブシステムを開発する上でも基礎となる共通概念を説明する。

T-Kernel を用いたシステムのプログラムには、大きく分けて T-Kernel ベースのプログラムとプロセスベースのプログラムがある。プロセスベースは主にアプリケーション開発に向いたプログラミングモデルであり、T-Kernel ベースは主にハードウェアを直接操作するデバイスドライバやミドルウェアを実現するサブシステムなどに向いたプログラミングモデルである。そして実用規模の組込みシステムになると、このどちらか一方で済むことはなく、両方をうまく組み合わせることで、はじめて高い性能と開発効率を両立できる。

まず 1 章では、ソフトウェアの分類や階層構造について述べる。T-Kernel や T-Kernel Extension を使ったシステムは、スケーラビリティに富むさまざまな組込みシステムに対応するため、各種のソフトウェアモジュールから構成されており、これらのモジュールは階層構造を持っている。モジュール化されたソフトウェア階層をうまく組み合わせることによって、全体として 1 つの応用システムを構築する形になる。

次に 2 章では、マルチタスクのプログラミングについて扱う。マルチタスクは、プロセスベースでも T-Kernel ベースでも、リアルタイム処理を行い、かつプログラムのモジュール性を高めるための最も重要な技術である。本章では、まずタスクの本質とその必要性について論じた後、タスク関連のプログラミングについて詳述する。最後にタスク間の排他制御や同期、通信機能についても説明する。

第 1 章
ソフトウェアの構成と分類

この章では T-Kernel や T-Kernel Extension の上でプログラミングする際に重要なソフトウェアの分類や階層構造について述べる。プログラムは大きくプロセスベースと T-Kernel ベースに分かれる。プロセスベースのプログラムはメモリ保護が効き、独立したメモリ空間とリソースを持っているため、独立性にすぐれたモジュールとなる。一方、T-Kernel ベースのプログラムも、デバイスドライバやサブシステムという形で汎用的なモジュールにすることができる。この「モジュール化の促進」という点が、T-Kernel や T-Kernel Extension を採用する大きなメリットの 1 つである。

1.1　T-Engine とは

組込み業界を中心とした約 500 社のメンバーで構成される T-Engine フォーラムは、組込み向け標準リアルタイム OS として、T-Kernel および T-Kernel Extension の仕様を策定し、ソースコードを含めて公開している。この T-Kernel は T-License というライセンスにのっとって、誰でも自由に利用したり、製品に組み込んだりすることができる。T-Kernel Extension についてもほぼ同様である。

T-Engine フォーラムでは、さらに、T-Kernel やその上で動く応用システムの開発評価用ボードとして、標準 T-Engine ボードおよびμT-Engine ボードのハードウェア仕様を定め、ボード上に載せるべきデバイスの種類や基板の寸法などを含めて標準化している。一方、標準 T-Engine ボードおよびμT-Engine ボードに搭載する CPU やバスの構成などは自由である。このため各 CPU の特徴を活かしたさまざまな標準 T-Engine ボードやμT-Engine ボードが存在する（図 1.1）。

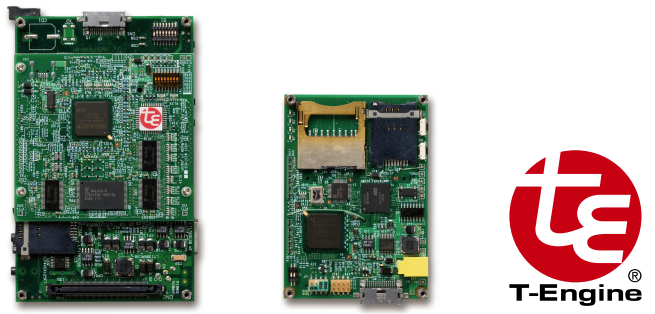


図 1.1　T-Engine（左から T-Engine ボード、μT-Engine ボード、T-Engine ロゴ）

ところで、実際の組込み機器を開発する場合、標準 T-Engine ボードやμT-Engine ボードをそのまま最終的な製品に組み込んで使う必要はない。最終的な組込み製品のハードウェアは、その製品の性能、機能、大きさ、コストなどに合わせて量産用に開発された別のボードによって実現されるのが一般的である。

これに対して、標準 T-Engine ボードやμT-Engine ボードは、最終製品向けの量産用ボードができる前の段階において、ソフトウェアの先行開発に利用する開発評価用ボードである。標準 T-Engine ボードやμT-Engine ボードには、この上で動く T-Kernel や T-Kernel Extension、デバイスドライバやミドルウェアがレディメイドの状態で整備されているため、開発者の意図するシステムを短期間で開発できる。また、この作業は、最終製品に組み込むべき量産用ボードの開発と並行して行えるため、ハードとソフトを含めた製品全体の開発期間を短縮できる。この様子を図 1.2 に示す。

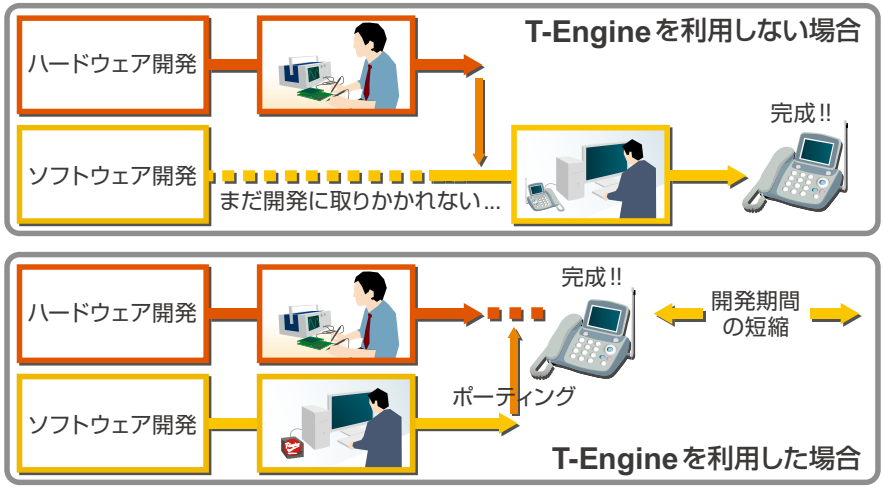


図 1.2　T-Engine を使った製品開発期間の短縮

T-Kernel や T-Kernel Extension を使った組込み機器は、T-Engine Appliance（T-Engine 応用製品）と呼ばれる。組込み機器にはいろいろな種類のものがあり、その見かけやハードウェア構成も多種多様であるが、この点については T-Engine Appliance もまったく同様である。すなわち、OS が T-Kernel であるという共通点はあるものの、各機器のハードウェアはそれぞれの組込み製品に最適な構成となっており（図 1.3）、T-Kernel の利用が機器のハードウェア構成を制約するわけではない。T-Engine フォーラムが標準化しているのは、あくまでも開発評価用ボードである標準 T-Engine ボードやμT-Engine ボードのハードウェア仕様であり、最終的に開発される組込み機器のハードウェア構成の多様性を阻害するものではない点に注意されたい。

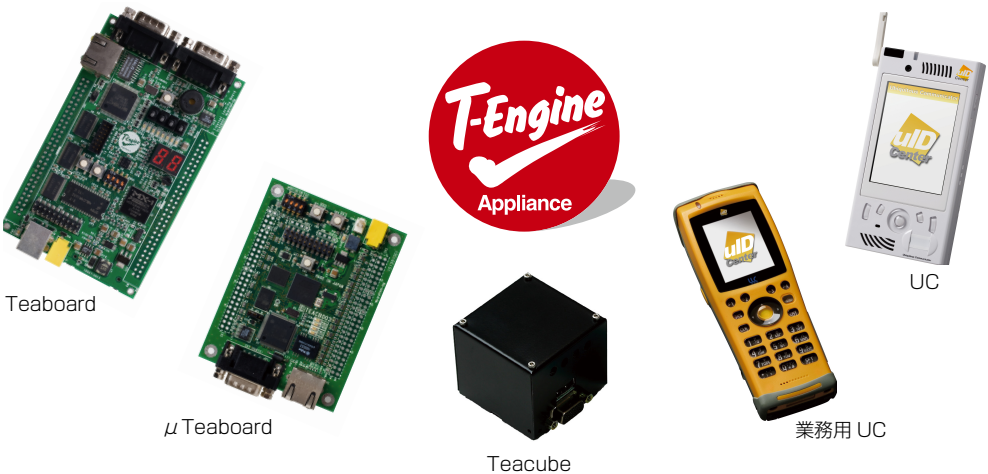


図 1.3　T-Engine Appliance の製品例

■スケーラビリティへの対応

一口に組込みといっても、その範囲はきわめて広いし、規模の違いも大きい。非常にわずかなリソースで動作しなければならない小さなシステムから、ファイルシステムや通信機能を持つ中規模なシステム、さらには GUI による画面表示を含むような大規模システムもある。組込みシステムの OS は、機器の用途の違いに加えて、こういったシステムの規模の違いにも柔軟に対応できなければならない。

T-Kernel の場合、T-Kernel 本体は機能を絞ったコンパクトな OS となっており、T-Kernel 本体だけを使えばリソースが少なく規模の小さいシステムに対応できる。一方、ファイルや GUI を含むような中規模、大規模なシステムを作る場合には、T-Kernel に T-Kernel Extension を追加したり、その他のミドルウェアを自由に追加して利用すればよい。すなわち、システム規模の大小に対しては、T-Kernel Extension やミドルウェアの着脱によって対応していくというのが T-Kernel の考え方である。

規模の大小に対して柔軟に対応できることを「スケーラビリティ」と呼ぶ場合があるが、T-Kernel では、リアルタイム OS の本体と T-Kernel Extension などによって実現される拡張機能を階層分けすることにより、このスケーラビリティを実現している。

■ T-Monitor の機能

では、T-Kernel を使った組込みシステムのソフトウェア構成を、ハードウェアに近い順に説明しよう。まず、T-Monitor は ROM 上に搭載される基本モニタである。電源投入時にはまず T-Monitor がハードウェア初期化を行い、それから T-Kernel のブート処理を行う。[1)]

T-Kernel 起動後も、割込みはまず T-Monitor の割込みエントリルーチンが受け付けて、それぞれの割込みハンドラに振り分ける形になっている（6.2.2 項参照）。

また、コマンドライン上でメモリ操作やブレークポイント設定などを行うデバッグ機能を持つ。C 言語レベルのデバッグ（GDB や Eclipse 版統合開発環境）も T-Monitor のデバッグ機能を用いて実現できる。

さらにプログラムから呼び出せるモニタサービス関数（`tm_xxxx`）も用意されている。

■ T-Kernel の機能

次に T-Kernel は T-Kernel/OS（Operating System）、T-Kernel/SM（System Manager）、

1） この意味では T-Monitor はパソコンでいう BIOS に相当する。

T-Kernel/DS（Debugger Support）の３つの部分から成り立っている。それぞれの機能を表 1.1 に示す。

表 1.1　T-Kernel の機能

T-Kernel/OS	タスク管理機能	＊1
	タスク付属同期機能	＊1
	タスク例外処理機能	＊1
	同期・通信機能	＊1
	拡張同期・通信機能	＊1
	メモリプール管理機能	＊1
	時間管理機能	＊1
	割込み制御機能	＊1
	サブシステム管理機能	＊2
T-Kernel/SM	システムメモリ管理機能	＊3
	アドレス空間管理機能	＊3
	デバイス管理機能	＊3
	割込み管理機能	＊3
	I/O ポートアクセスサポート機能	＊3
	省電力機能	＊3
	システム構成情報管理機能	＊3
T-Kernel/DS	カーネル内部状態参照機能	＊4
	実行トレース機能	＊4

（注）ここで示した＊1 ～＊4 は本書での説明のために独自に分類したものであり、仕様書上の分類ではない。

このうち＊1 をつけて示した部分は、タスクやセマフォといったオブジェクトを管理する、リアルタイム OS としての最も基本的な部分である。この部分は組込み分野で長年の実績のある μITRON を機能的にも仕様的にも発展的に継承している。

T-Kernel としての重要な点は、μITRON にはなかった、＊2、＊3、＊4 の機能が新たに追加されていることである。T-Kernel ではこれらの部分を新たに標準化することによって、ミドルウェアの流通を促進し、ミドルウェアやアプリケーションの共通の土台となる標準プラットフォームを目指している。

特に＊2 は、OS の機能を拡張するためのサブシステムという非常に重要な枠組みであり、＊3 の機能や T-Kernel Extension は、このサブシステムの枠組みを利用して実現されている。

＊3 は、メモリ空間の管理やデバイスドライバ、ハードウェア操作などの実際のシステム開発で必要になる機能であり、この部分の標準化は、プログラムの開発効率や移植性の向上に欠かせないものである。

また、＊4 は、デバッガを実装するために必要な機能であり、標準化によりデバッガや各種ツールの実装を容易にし、かつ移植性の向上に貢献している。

本 PDF は『T-Kernel 組込みプログラミング強化書』のお試し版です。

本書掲載のプログラムコードがダウンロードできます。
詳しくは正式版をご覧ください。

T-Kernel 組込みプログラミング強化書（お試し版）
開発現場ですぐに役立つ実践テクニック

2007 年 12 月 20 日　初版 1 刷発行（書籍版）
2011 年 11 月 21 日　PDF 版 発行

監　修　　坂村健
編　著　　パーソナルメディア株式会社
発行所　　パーソナルメディア株式会社
〒 141-0031　東京都品川区西五反田 1-29-1　コイズミビル
TEL　：03-5759-8303
FAX　：03-5759-8306
E-mail：pub@personal-media.co.jp
http://www.personal-media.co.jp/

PMBK-246-PDF-00-01-sample