宇宙機搭載用リアルタイム OS に適用した 高信頼化技術のハンドブック化

○佐藤 伸子†, 石濱 直樹†, 川崎 朋実†, 片平 真史†

ロケットや人工衛星などの宇宙機に搭載するリアルタイム OS(RTOS)を高信頼化するためには，その RTOS をどのように検証しているかを明らかにする必要がある．このため，民間航空機や原子力など信頼性が重視される分野で適用されている技術標準や，過去に宇宙機の搭載計算機で発生した不具合事例を参考に，RTOS に特化した検証要求を整備した．この検証要求を，TOPPERS/HRP カーネルの検証作業で繰り返し適用し，必要な解説を追加するなどの改良を重ね，「リアルタイムOS 高信頼化ハンドブック」として編集したので紹介する．これは，宇宙機だけでなく，信頼性が重視されるシステムの RTOS に広く適用できるものである．

Establishment of a Reliability Handbook for RTOS in Spacecrafts

Nobuko SATO, Naoki ISHIHAMA, Tomomi KAWASAKI, Masafumi KATAHIRA
(Japan Aerospace Exploration Agancy (JAXA))

To improve reliability of Realtime Operating System (RTOS) for spacecrafts such as space vehicles and satellites, it is necessary to clarify how the RTOS is verified. JAXA has organized a verification requirement specialized for RTOS, by reference technical standards that is applied to domains who place emphasis on reliability, such as civilian aircraft and nuclear energy, and past failure cases occurred on spacecraft's onboard computer systems. JAXA applied the verification requirement repeatedly to TOPPERS/HRP kernel, and refined it by adding needful exposition. The paper introduces application of "RTOS Reliability Improvement Handbook". This handbook is possible to apply to not only spacecrafts but also other reliable computer systems.

1. はじめに

ロケットや人工衛星などの宇宙機に搭載するリアルタイム OS(RTOS)は，MPU の高速化やアプリケーションの高度化に伴い，その重要性がますます高まってきている．RTOS を高信頼化するためには，その RTOS がどのように検証されているかを明らかにする必要があるが，アプリケーションと比較して，その基準はあいまいで，RTOS 向けに要求としてまとめられたものはなかった．このため，システム開発者は，RTOS をブラックボックスとして使用することが多く，ひとたびトラブルが発生すると，原因究明が困難になることがあった．

このような状況を改善するため，筆者らは，宇宙機搭載用 RTOS の検証要求をまとめるべく，2003 年ごろから調査・研究に着手した．民間航空機や原子力など信頼性が重視される分野で適用されている技術標準や，過去に宇宙機の搭載計算機で発生した不具合事例を参考に，2004 年までに，RTOS 向けの検証要求を整理した．なお，ここでの「検証」は，ソースコードや仕様書・設計書などのドキュメントを分析する静的検証(レビュー)と，プログラムを実際に動作させて確認する動的検証(テスト)の組み合わせからなり，varlidation(妥当性確認)を含んでいる．

この検証要求が実際の RTOS に対して適用可能であるかを確かめるため，μITRON4.0 仕様の TOPPERS/JSP カーネル(注 1)に，信頼性を向上させる機能を付加した RTOS を開発し，供試体とした．

RTOS の開発者とは異なる検証作業者に対し，上述の要求に基づいた検証を実施することを要求し，最初

† 独立行政法人 宇宙航空研究開発機構(JAXA)

(注 1) NPO 法人 TOPPERS プロジェクトの開発成果

の検証作業を 2006 年までに完了した. 検証作業を実行できたことにより, この検証要求には現実的な手法が伴っており, 単なる理想論ではないことを確認することができたが, 要求の意図や手法について検証作業者に正しく理解させるため, 実際には, 細かな打合せや解説書が必要であった.

2007 年以降, 供試体とした RTOS を改修する度に, 検証要求の適用を繰り返したところ, 当初からこの研究に関わってきた検証作業者は, 打合せや解説書により, 要求の意図や手法を十分理解できるようになった. しかし, 今後, 検証作業者が交代したり, 他の検証作業者が別の RTOS に適用したりすることは, 十分, 想定され, その時に, これまで打合せで補足した内容や, 解説書の存在が忘れられる心配が生じてきた. また, 検証作業は, RTOS 開発者だけの作業ではない. RTOS が標準で提供する BSP(Board Support Package)部は, 評価ボード用なので, ユーザは, 自分が使用する計算機ボード用に, 提供された BSP 部をカスタマイズし, 検証する必要がある. ところが, 上述のように, RTOS の改修に合わせて解説書等を充実させるうちに, RTOS 開発者が求めるような専門性の高い記述が増え, ユーザには読みづらい構成と内容になっていた. そこで筆者らは, この検証要求の他の RTOS への適用や, 他の作業者へのスキル継承をスムーズに進めるため, 2010 年に, RTOS の高信頼化に必要な技術をまとめ, ハンドブックとして整備した.

本稿では, RTOS 開発者のスキル向上と技術継承, RTOS ユーザの知識向上を目的に, 双方に参照されることを狙ってまとめた, JAXA の「リアルタイム OS 高信頼化ハンドブック」について紹介する.

2. RTOS の「高信頼化」とは

2.1. 「高信頼化」の条件

本稿で述べる RTOS の「高信頼化」は, 次の 2 つの条件がいずれも満たされて, 初めて達成できるものである.

(1) RTOS が信頼に足る機能・性能を有すること.
(2) それらが適切な技法で検証されていること.

宇宙機搭載用に開発した TOPPERS/HRP カーネル(注 2)と Safety カーネルは, これまでに筆者らが高信頼化した唯一の RTOS である. この RTOS を例にとり, 高信頼化を説明する.

(注 2) NPO 法人 TOPPERS プロジェクトの開発成果. JAXA と名古屋大学大学院情報科学研究科組込みリアルタイムシステム研究室(高田・冨山研究室)の共同研究による.

2.2. 信頼に足る機能・性能

TOPPERS/HRP カーネルと Safety カーネルの信頼性機能の検討にあたっては, ハードウェア, RTOS 及びアプリケーションからなる計算機システム全体の信頼性を, RTOS の機能を使って向上させることを目標とした. その着眼点は, 次の 3 点にある.

(1) 過去に発生したソフトウェアの不具合事例と同様の不具合の防止.
(2) 計算機システムの状態の常時監視と, 異常発生時の終了処理や再起動処理の安全な実行.
(3) 宇宙機搭載用計算機システムに共通する機能でありながら, 従来, アプリケーション側で実装していた機能で, 処理方法の統一化により計算機システムの信頼性が向上できるものは, RTOS へ編入.

これら 3 つの着眼点で検討した信頼性機能(5.1 項参照)を実装することで, TOPPERS/HRP カーネルと Safety カーネルは, 宇宙機搭載用 RTOS として, 高信頼化の第一の条件を満たしているといえる.

2.3. 適切な技法での検証

TOPPERS/HRP カーネルと Safety カーネルの検証にあたっては, 従来からある一般的なソフトウェア向けの検証要求ではなく, 次章から説明する, RTOS に絞った明確な検証要求を新たに設定し, 適用した.

検証は, ソースコードや仕様書・設計書などのドキュメントを分析する静的検証(レビュー)と, プログラムを実際に動作させて確認する動的検証(単体テスト, 結合テスト, システムテスト)の組み合わせからなる. テスト結果などの検証エビデンスは, 閲覧性・検索性を確保して保管しており, ユーザの求めに応じ, 最新の状態で提示できる状態にある. よって, TOPPERS/HRP カーネルと Safety カーネルは, 高信頼化の第二の条件も満たしているといえる.

3. 検証要求の設定

アプリケーションと比べ, RTOS は, 必要な処理時間の予測を行ったり, 複数の処理要求が同時に発生した場合でも目的の時間内に処理を完了させたりといった, リアルタイム性に関わる機能を求められるのが特徴的である. また, 組込み機器では, 汎用のコンピュータとは異なり, 厳しいリソース制約が要求されることが多く, 搭載される RTOS に対しても効率的なリソースの使用が求められる.

ソフトウェア一般の検証要求を RTOS に「特化する」

とは，“RTOS” というソフトウェアのこのような機能や使い方に着目し，適用外の要求を削除して不足している要求を追加することと，ソフトウェア一般向けに抽象化された要求を RTOS に限定して具体化することの両方を指している．

RTOS に特化した検証要求を設定するにあたり，産業全般と，信頼性・安全性が重視される 5 つの産業分野(医療機器，原子力発電所，鉄道，軍事機器及び民間航空機)の技術標準を調査した．調査結果を分析した結果，各分野それぞれから，ソフトウェアに対する実践的なテスト技法とテスト基準について示している 6 つの技術標準[1]-[6]を参考にすることとした．

これらに記述されている検証要求とテスト技法は，それぞれの分野で使用される全てのソフトウェアに適用できる半面，RTOS のように，ある特定の目的をもったソフトウェアに適用するには，情報が過多であったり抽象的であったりした．そこで，アプリケーションとは異なる “RTOS” というソフトウェアの機能や使い方に着目して，これらの技術標準から，RTOS の特徴に合う検証要求とテスト技法を抽出し，具体化した．さらに，過去に宇宙機の搭載計算機で発生した不具合事例を分析し，同様の不具合を発生させないために必要なテストをこれに加味し，RTOS に特化した検証要求として設定した．表 1 に，抽出したテスト技法をテスト項目で分類して示す．なお，テスト技法の名称は，一般的な和名に置き換えている．

表 1 6 つの標準から抽出したテスト技法

テスト項目	テスト技法
要求事項のテスト	外部要求事項のテスト
	内部要求事項のテスト
入力域のテスト	代表値のテスト／同値分割
	境界値分析／異常入力域のテスト／極端な値のテスト
	典型入力組み合わせテスト
	入力値と出力値の全対応のテスト
	入力値の閾値テスト
	計算式の閾値テスト
	入力値と内部状態の全組み合わせテスト
システム構造のテスト	データ結合および制御結合の網羅テスト
	状態遷移テスト
	ソフトウェアインタフェーステスト
	ハードウェア・ソフトウェアインタフェーステスト
	ハードウェア障害対応テスト
	エラー推測テスト
モジュール内構造のテスト	MC/DC テスト
	条件網羅テスト(C1)
	命令網羅テスト(C0)
	ループテスト(最小・最大・中間回数)
	データフローパステスト
メモリ関連のテスト	メモリ割当てテスト
	メモリ参照テスト
割込み・タイミング・負荷のテスト	割込みシーケンスの最大組み合わせテスト
	負荷テスト

検証要求を満たすため，これらのテスト技法を使用し，検証する RTOS の特性を考慮して，テストケース設計を行う．RTOS では，特に，メモリやタイミングの検証を重要視する．

4. 高信頼化ハンドブック

4.1. ハンドブック化の目的

前章で，検証要求の骨格はできたが，有効な検証を実施するためには，テストケースを設計する検証作業者が，検証要求の意図を正しく理解している必要がある．ハンドブック化にあたっては，検証作業者が，単純なチェック作業に陥ることなく，自ら考えることが重要と考え，テストの目的設定やテスト対象の分析の方法などの記述を充実させた．また，対象読者を明確にイメージした章構成をとり，記述レベルに配慮した．

4.2. 対象読者と前提知識

ハンドブック化にあたり，対象とする読者と，読者に求める知識レベルを次のように設定し，記述レベルや構成に反映した．

【対象読者】

(1) RTOS の開発や検証に従事するソフトウェア技術者
(2) RTOS を組み込んで利用するシステム開発者

【前提知識】

(1) RTOS の基本的な知識
(2) 一般的なソフトウェアの検証に関する基本的な知識

4.3. 検証プロセスが対象とする開発モデル

このハンドブックが検証の対象とするのは，ウォータフォール V 字型モデルで開発される RTOS である．検

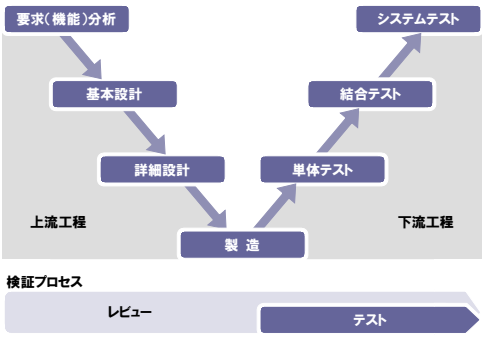

図 1 V 字型モデルと検証プロセス

証プロセスでは，まず，対象となる RTOS を分析し，静的検証(レビュー)と動的検証(テスト)が必要な要素を識別する．レビューとテストは開発と平行して行い，上流工程ではドキュメントのレビュー，下流工程ではプログラムを動作させてのテストが主体となる．図 1 に，開発プロセスと検証プロセスの関係を示す．

また，冒頭で述べたように，検証は，RTOS 開発者(ベンダ)だけの作業ではない．ベンダが標準で提供する BSP 部は評価ボード用のサンプルなので，ユーザは，自分が使用する計算機ボード用に，提供されたサンプル BSP 部をカスタマイズし，検証する必要がある．図 2 を用いて説明する．

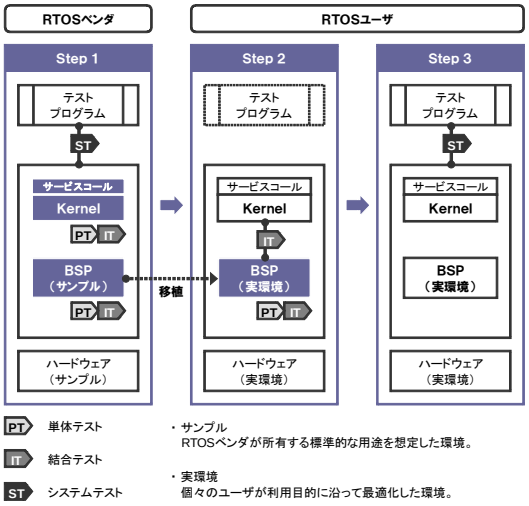

図 2 検証におけるベンダとユーザの役割分担

STEP1 において，ベンダの主目的はカーネル部の検証である．サンプル BSP 部を検証する目的は 2 つあり，1 つは，カーネル部の検証の信頼性を高めるため，もう 1 つは，ユーザの BSP 開発を助けるためである．ユーザが開発する実用の BSP 部は，STEP2 でユーザ自身が検証しなければならない．提供されたサンプル BSP 部をカスタマイズしたのであれば，差異のない部分については，サンプル BSP 部の検証結果を参照して差し支えない．STEP3 では，実際に使用する計算機ボードと BSP 部を用いて，特に，評価ボード，サンプル BSP 部との差異がカーネル部に与える影響を考慮し，ユーザがカーネル部も含め，計算機システム全体を再検証する．

ハンドブックの記述や構成では，このように，RTOS のベンダとユーザでは，検証の目的と範囲が異なることに配慮している．

4.4. ハンドブックの構成

RTOS の高信頼化は，開発の上流工程から始めなければならないが，テクニックを必要とするのは，やはり，下流工程でのテストである．このハンドブックでは，検証作業者が，検証要求の意図を正しく理解したうえで，テストを円滑かつ確実に進めることができるように，目的設定，分析，設計，そして実施と報告について，サンプルを交えつつ説明している．表 2 に，ハンドブックの構成を示す．

表 2 高信頼化ハンドブックの構成

章タイトル	各章の概要	対象読者
第 1 章 概要	RTOS の構造と機能，RTOS の開発，RTOS の高信頼化に必要なテストの進め方について	全ての読者
第 2 章 単体テスト 第 3 章 結合テスト 第 4 章 システムテスト	単体テスト，結合テスト，システムテストの詳細について (テストの目的，テスト対象の分析やテストケースの設計と実施，テスト結果のまとめ)	主にソフトウェア技術者
第 5 章 BSP テストケース設計時の留意点	BSP (Board Support Package) のテストケースを設計する際に留意すべき点について	主にシステム開発者
付録 A テスト技法の参考情報	―	主にソフトウェア技術者
付録 B テスト項目の検討経緯	―	全ての読者

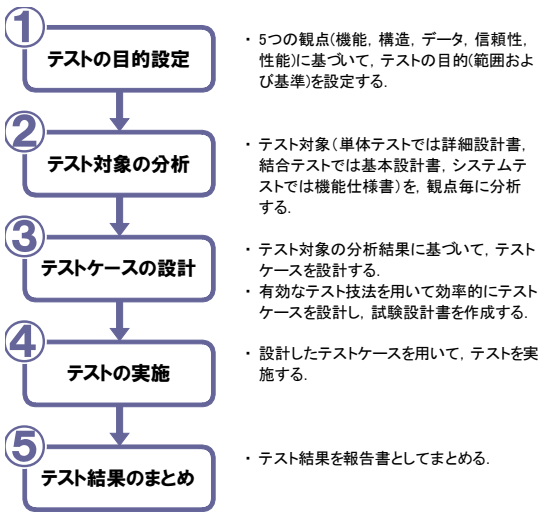

図 3 テストの進め方

第 2 章から第 4 章では，各テストフェーズ(単体テスト，結合テスト，システムテスト)でのテストの進め方を，それぞれ，図3のような流れで整理している．テスト対象の分析ポイントやテストケースの設計ポイントを記述するにあたっては，RTOS 向けに具体的で分かりやすい説明を心がけた．後述の 5.2 項で，システムテストを取り上げて例示している．

第 2 章から第 4 章は，C 言語で記述されたカーネル部への適用を想定しているが，BSP 部はハードウェアに依存するため，アセンブラ言語で記述される部分やハードウェアで制約される部分があり，第 2 章から第 4 章の内容が適用できない場合がある．このため，第 5 章では，BSP 部のテストケースを設計する際の留意点として，次を挙げている．

【アセンブラ言語で記述されている関数】

- C 言語の命令網羅テスト相当として，ブラックボックステストを実施すること．
- ループ処理の途中に実行がジャンプしてくる可能性を考慮すること．

【ハードウェアの制約】

- 模擬ハードウェアを使用するなどにより，実施できないテストケースがある場合は，理由を明確にして記録を残すこと．

なお，C 言語で記述された BSP 部については，第 2 章から第 4 章の内容が適用できる．

5. 適用事例

設定した検証要求を，TOPPERS/HRP カーネルと Safety カーネルの検証に，繰り返し適用した．実際に適用することで得られた知見を，このハンドブックに反映している．適用結果として，以下に，TOPPERS/HRP カーネルと Safety カーネルの概要を示し，具体的な実践例として，TOPPERS/HRP カーネルのシステムテストを取り上げる．

5.1. TOPPERS/HRP カーネルと Safety カーネル

2.1 項でも触れたとおり，TOPPERS/HRP カーネルと Safety カーネルは，高い信頼性を要求される宇宙機に搭載されることを前提に開発された RTOS である．図 4 に，この RTOS を模式的に示す．TOPPERS/HRP カーネルは，RTOS の主体をなす．Safety カーネルは，TOPPERS/HRP カーネルとアプリケーションの状態を監視し，HRP カーネルに異常が生じた場合には，TOPPERS/HRP カーネルに代わって，アプリケーションの実行を制御する．

【TOPPERS/HRP カーネルの特長】

TOPPERS/HRP カーネルは，計算機システム全体の信頼性に寄与する RTOS の一般的な機能(μITRON4.0 仕様のスタンダードプロファイル)に加え，高信頼性機能として，以下の機能を有する．

- メモリプロテクション(メモリの破壊及び誤アクセスの防止)
- ミューテックス(プライオリティインバージョンの防止)
- アラームハンドラ(デッドロックの防止)
- オーバーランハンドラー(タスクの暴走の防止)

【Safety カーネルの特長】

TOPPERS/HRP カーネル上，又は TOPPERS/HRP カーネル自身に復旧不可能な異常が生じ，全てのソフトウェアが動作しない状況下に陥った場合，メモリがいかなる状態でも(仮に壊れていたとしても)，予め設定したイベント処理を行い，計算機システムの安全性を確保することができる．イベント処理の動作ログを記憶し，復旧後に読み出すことができる．

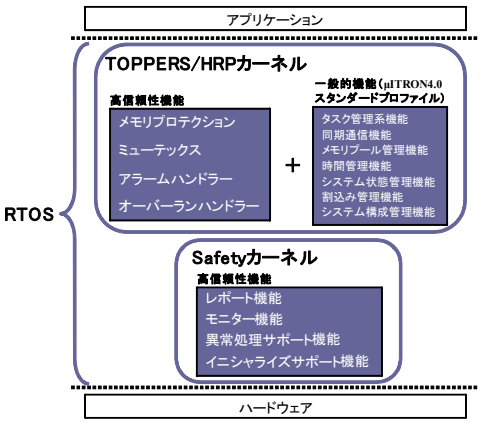

図 4 TOPPERS/HRP カーネルと Safety カーネル

5.2. 検証の実践例：HRP カーネルのシステムテスト

5.2.1. システムテストの狙い

システムテストの狙いは，結合テストの完了後に，ハードウェア上で RTOS 全体を対象にテストを実施し，RTOS の提供する機能が正しく実装されているか，信頼性や性能が要求を満たしているかを確認することにある．

5.2.2. テストの目的設定

ハンドブックに示された観点(機能，信頼性，性能)に基づき，TOPPERS/HRPカーネルの特性を考慮して，システムテストの目的を設定した．今回は，日本ノーベル(株)製の「μITRON4.0 評価システム」(注 3)上に独自に構築した環境を使用した機能テストと，検証用に製作したアプリケーションモデルを使用した信頼性・性能テストを実施することとした．

5.2.3. テスト対象の分析

ハンドブックに示された観点毎の分析ポイントで，テスト対象であるTOPPERS/HRP カーネルの API 機能仕様書を分析した．分析結果の一部を表 3 に示す．表中の「○」は分析内容に該当する記述が API 機能仕様書にあって，テストケース設計が必要なもので，「△」は記述はないが，テストケース設計が必要なものである．

5.2.4. テストケースの設計

API 機能仕様書の分析をさらに進め，ハンドブックに示されたテストケース設計ポイントがテスト対象に存在するかを分析し，抽出したテストケース設計ポイントを一覧にした(表 4)．テストケースの設計にあたっては，このポイントが，いずれかのテストケースで必ず確認できることを確認し，μITRON4.0 評価システムやアプリケーションモデルに組み入れた．

5.2.5. テストの実施～結果のまとめ

設計したテストケースを使用してテストを実施した．テストケースの実例として，アプリケーションモデルを使用したテストケースの一部を表 5 に示す．ここに示す小項目毎にテストを実施し，結果を記録した．他のテストについても同様に実施し，MS Word や Excel を使用して，試験結果を報告書にまとめた．

表 3 API 機能仕様書の分析結果(一部)

No.	分析内容	分析結果
1	同時に複数の割込みが入った場合の動作が定義されている.	○
2	割込みハンドラ実行中に別の割込みが入った場合の動作が定義されている.	○
3	割込みハンドラ動作中に，待ち時間の終了や，待ち解除関連のサービスコール発行により，通常であれば処理が切り替わるような状況となった場合の動作が定義されている.	○
4	最大割込み禁止時間についての要件が明確である.	△
5	同時に使用できる資源の数についての要件が明確である.	△
6	タスクスイッチ時間についての要件が明確で	△
11	高負荷状態での性能条件に 明確である.	△
12	ハードウェアまたはソフトウェアで異常が発生した場合の動作が網羅されている.	△
13	異常発生時に動作不安定となる場合がある.	△
14	資源を共有している処理があり，競合する可能性がある.	△
15	グローバルデータを共有している処理があり，競合する可能性がある.	△
16	要求する単位で正確な時間を刻み続ける仕組みがない.	○
17	資源の解放漏れがある.	△
18	アプリケーションがメモリを破壊してしまった場合の挙動が定義されている.	○

表 4 システムテストのテストケース設計ポイント

観点		テストケース設計ポイント	テスト技法
機能	割込み	割込みハンドラ動作中に，待ち時間の終了や，待ち解除関連のシステムコール発行により，通常であれば処理が切り替わるような状況となった場合の動作が，仕様どおりであることを確認する.	外部要求事項のテスト
	ディスパッチ	ディスパッチ禁止中に，待ち時間の終了や，待ち解除関連のシステムコール発行により，通常であれば処理が切り替わるような状況となった場合の動作が，仕様どおりであることを確認する.	外部要求事項のテスト
信頼性	負荷	目標とする動作条件の範囲内では，正常動作することを確認する.	負荷テスト
		目標とする動作条件の範囲を超えた場合の，動作を確認する.	
		目標とする動作条件の範囲内で，性能に対する要求がある場合は，要求仕様通りであることを確認する.	
		その他，高負荷状態での振舞いについて，要件を満たしていることを確認する.	
	異常	ハードウェアで異常が発生した場合の動作が，仕様どおりであることを確認する.	外部要求事項のテスト
		ソフトウェアで異常が発生した場合の動作が，仕様どおりであることを確認する.	
		その他異常が発生した場合，動作が仕様どおりであることを確認する.	
	競合	資源を共有している処理がある場合，競合がおこらず，仕様どおりに動作することを確認する.	外部要求事項のテスト
		グローバルデータを共有している処理がある場合，競合がおこらず，仕様どおりに動作することを確認する.	
		ディスパッチ禁止を多重に行った場合の動作が，仕様どおりであることを確認する.	
		サービスコールの発行タイミングを，さまざまなパターンに変更し，いずれも問題なくサービスコールの要求が有効となっていることを確認する(正常終了しても実際はサービスコールの要求が有効とならないことはないことの確認).	
		資源を共有している機能同士を組み合わせ，資源解放漏れがないことを確認する.	
		割込みが入った場合の動作が仕様どおりで	

(注 3) http://www.jnovel.co.jp/service/itron/index.html

表 5 アプリケーションモデルを使用した
テストケース(一部)

大項目	中項目	小項目
割込み	同時に複数の割込みが発生した場合の動作確認	INT2, INT3の割込みが同時に発生した場合，INT2_hdr, INT3_hdr の順で起動すること
		INT1, INT2の割込みが同時に発生した場合，INT2_hdr, INT1_hdr の順で起動すること
		INT2, INT4の割込みが同時に発生した場合，INT2_hdr, INT4_hdr の順で起動すること
		INT2, INT3, INT5の割込みが同時に発生した場合，INT2_hdr, INT3_hdr, INT5_hdr の順で起動すること
		INT0, INT1, INT2の割込みが同時に発生した場合，INT2_hdr, INT1_hdr, INT0_hdr の順で起動すること
		INT2, INT4, INT5の割込みが同時に発生した場合，INT2_hdr, INT5_hdr, INT4_hdr の順で起動すること
		INT0, INT1, INT2, INT3, INT4, INT5の割込みが同時に発生した場合，INT2_hdr, INT3_hdr, INT5_hdr, INT1_hdr, INT0_hdr, INT4_hdr の順で起動すること
		INT0, INT1の割込みが同時に発生した場合，INT0_hdr, INT1_hdr が起動しないこと
		INT0, INT1, INT2の割込みが同時に発生した場合，INT0_hdr, INT1_hdr, INT2_hdr が起動しないこと
		INT0, INT3の割込みが同時に発生した場合，INT3_hdr が起動すること
	割込みハンドラ実行中に割込みが発生した場合の動作確認	INT3_hdr 実行中に INT2が発生した場合，INT3_hdr→INT2_hdrの順で処理を行うこと
		INT2_hdr 実行中に INT3が発生した場合，INT2_hdr→INT3_hdrの順で処理を行うこと
		INT4_hdr 実行中に INT2が発生した場合，INT2_hdr→INT4_hdrの順で処理を行うこと
		INT2_hdr 実行中に INT1が発生した場合，INT2_hdr→INT1_hdrの順で処理を行うこと

5.3. 実践結果

5.2 項のシステムテストと同様の進め方で，TOPPERS/HRP カーネルの単体テストと結合テスト，Safety カーネルの単体テスト，結合テスト及びシステムテストを実施した．いずれにおいても，テストケース設計の観点と具体的なポイントが，ハンドブックにガイドラインとして示されているため，検証作業者は，必要なテストケースを全て，その必要性の根拠を理解したうえで，設計することができた．

6. まとめ

本稿では，RTOS の高信頼化の考え方を示し，それを具現化するためのテクニックをまとめたハンドブックを紹介した．信頼性の高い検証を実施するためには，検証作業者に検証要求を十分に理解させる必要があるという課題に対し，ハンドブックに示したガイドラインを，実際に宇宙機搭載用 RTOS の検証作業に適用したところ，検証作業者は，テストの必要性の根拠を理解してテスト設計をすることができ，このハンドブックの有用性を確認することができた．このハンドブックを適用した RTOS は，現在のところ，TOPPERS/HRP カーネルと Safety カーネルのみであるが，今後は，宇宙機に搭載する別の RTOS にも適用を広げ，高信頼化を図りたい．また，このハンドブックの内容は，宇宙機だけでなく，信頼性が重視される計算機システムの RTOS に広く適用できるものであるため，民生分野において参照されることを期待する．ハンドブックの入手方法については，JAXA 情報・計算工学センター(prjedi@jaxa.jp)へ問い合わせられたい．

参考文献

[1] JIS C 0508-7 (IEC61508-7), 電気・電子・プログラマブル電子安全関連系の機能安全 - 第7部：技術及び手法の概観, 2000

[2] General Principles of Software Validation; Final Guidance for Industry and FDA Staff, 2002

[3] IEC880 (IEC60880) / Edition 1, Software for computers in the safety systems of nuclear power stations, 1986

[4] IEC 62279 / Edition 1, Railway applications - Communications, signalling and processing systems - Software for railway control and protection systems, 2002

[5] DEF STAN 00-42 (PART 2) / Issue 1, Reliability and Maintainability Assurance Guides Part 2: Software, 1997

[6] RTCA DO-178B, Software Consideration In Airborne Systems and Equipment Certification, 1992