情報処理学会論文誌 Vol.59 No.2 785–794 (Feb. 2018)

AUTOSAR OS仕様準拠TOPPERS/ATK2を対象とした機能安全規格ISO 26262対応における安全分析事例

毛利 守男[1,a)] 佐藤 秀昭[2] 山下 映[3] 松原 豊[4] 高田 広章[4]

受付日 2017年5月25日, 採録日 2017年11月7日

概要：AUTOSARを用いた車載制御システムを機能安全対応させる場合，OSおよびOS上で動作するアプリケーションの間で，安全要求の違反につながる故障が存在しないことを示す必要がある．そのための取り組みにおいて特に重要となる作業が安全分析である．ソフトウェアを対象とする安全分析手法は過去に提案されているが，AUTOSAR OSを含め，リアルタイムOSを対象とした安全分析の実施方法や事例は，特定の製品や実装への依存度が高く，企業の機密事項にもなりうるため，一般に公開されている情報は限定的である．本論文では，リアルタイムOSの機能安全対応方法の具体的な事例として，AUTOSAR OS仕様準拠TOPPERS/ATK2を対象とした安全分析の実施事例を述べる．産学連携プロジェクトにおいて，TOPPERS/ATK2に対する安全コンセプトの構築，安全分析を実施し，その結果に基づいてアプリケーション開発側とOS開発側が実施すべき安全対策を導出する．加えて，保護機構を含むOSについても分析を行い，対策を導出する．最後に，本事例を通じて得られた知見を整理する．

キーワード：AUTOSAR，RTOS，ISO 26262，機能安全，安全分析，HAZOP，SHARD

Safety Analysis Examples in Compliance with ISO 26262 for TOPPERS/ATK2 Conforming to the AUTOSAR OS Specification

Morio Mori[1,a)] Hideaki Sato[2] Akira Yamashita[3] Yutaka Matsubara[4] Hiroaki Takada[4]

Received: May 25, 2017, Accepted: November 7, 2017

Abstract: In case of adapting the automotive control system using AUTOSAR Platform to functional safety standards, it is necessary to indicate that there are no failures leading to a violation of safety requirements. Safety analysis is a key activity for indication. Nevertheless, the public information about safety analysis is limited. In this paper, we describe an example of the safety analysis for TOPPERS/ATK2 conforming to the AUTOSAR OS specification. In the industry-academia cooperation project, we perform safety analysis for TOPPERS/ATK2, and derive safety measures to be implemented by application development side and OS development side based on the result.

Keywords: AUTOSAR, RTOS, ISO 26262, functional safety, safety analysis, HAZOP, SHARD

1. はじめに

近年，車載制御システム開発においてAUTOSAR [1] への注目が高まっている．ソフトウェアプラットフォームとしてAUTOSARを用いることにより，アプリケーションの複雑性の軽減と再利用性の向上が期待できることがその理由である．名古屋大学大学院情報学研究科附属組込みシステム研究センターではAUTOSAR OS [2] 準拠のリア

1 SCSK 株式会社
SCSK Corporation, Koto, Tokyo 135–8110, Japan
2 株式会社ジェイテクト
JTEKT CORPORATION, Okazaki, Aichi 444–2106, Japan
3 日本電気通信システム株式会社
NEC Communication Systems, Ltd., Minato, Tokyo 108–0073, Japan
4 名古屋大学
Nagoya University, Nagoya, Aichi 464–8601, Japan
a) morio.mouri@scsk.jp

ルタイム OS として，TOPPERS/ATK2 [3]（以下，ATK2 と表記）SC1～SC4 を開発し，リリースしている．SC は Scalability Class の略で，AUTOSAR OS 機能群の実装状況を SC1～SC4 の 4 段階で表している．SC1 は機能群の最小セット，SC2 は SC1 の機能およびタイミング保護機能を中心とする機能群，SC3 は SC1 の機能およびメモリ保護機能を中心とする機能群に対応する．SC4 は SC2，SC3 の全機能に対応することを意味する．

AUTOSAR を用いるシステムは，車載制御システム用途で利用されることから，車載電気・電子システム向け機能安全国際規格 ISO 26262 [4], [5] 対応を求められる機会が増えている．そこで我々は，ATK2 を機能安全規格に対応させる活動を進めている．

AUTOSAR では ISO 26262 への対応状況について，文献 [6] において情報を提供しているが，この 3.4 節において，AUTOSAR 自身が対応しない項目を表明している．これらについては，AUTOSAR のユーザ自身が対応する必要がある．本論文では，非対応事項の中で，特に安全分析に関連する事項を対象とする．

安全関連システムを開発する場合，実装者は安全要求（システムの安全に関する要求事項）を満たすように設計，実装し，そのうえで実際に満たしていることを検証する必要がある．安全要求を仕様化し，実装するために必要となる取り組みが安全分析である．安全分析では，フォールト（故障を引き起こす可能性のある異常な状態）と故障の因果関係を分析する．安全分析を行うことにより，安全要求の侵害を引き起こす原因の特定や，その対策にあたる安全方策の決定が可能となる．安全方策は，安全要求の侵害を防止するための活動（安全分析，レビューやテスト，ユーザに対するマニュアル作成等）と安全機構（ウォッチドッグタイマやメモリ保護ユニット等の技術対策）で構成される．

TOPPERS/ATK2 に対して安全分析を行ううえでの課題について 2 点述べる．1 点目は公知の情報の入手が困難であるという点である．AUTOSAR OS 仕様準拠の製品には，ISO 26262 対応を行ったことを表明しているものがいくつか存在しており，その情報が開示されている [7], [8], [9]．しかし，公開されている情報は，採用した手法とその位置づけの概要にとどまり，詳細な情報は開示されていない（たとえば，文献 [7] では，HAZOP を修正した手法が使用されているが，その具体的な修正内容は明かされていない）．2 点目は，リアルタイム OS（AUTOSAR OS を含む）や通信ミドルウェア等の共通ソフトウェアに対する安全分析の手順，方針，対策の検討方法が確立されていないという点である．前述の文献 [6] や文献 [4] Part 9 においても，複数の分析手法が提示されているだけで，その手法をいかに適用するか，適用した際の課題や知見といった詳細までは提示されていない．

本研究の目的を以下に 2 点述べる．1 点目は，AUTOSAR OS 仕様準拠の TOPPERS/ATK2 に対する機能安全対応の考え方を，事例を通じて明らかにすることである．具体的には，安全を確保するための基本的な考え方（安全コンセプト），安全分析に関する方針の決定，前提の整理，分析対象の精査，およびこれらに基づく分析手法の選定や適用の実際を紹介し，得られた知見を整理する．2 点目は，安全分析の手法の適用性を明らかにすることである．構築した安全コンセプトと，安全分析手法を，実際のリアルタイム OS である TOPPERS/ATK2 の要求仕様書，設計書，ソースコードに適用し，実際のシステムに対する機能安全対応の実現性を確認し，そこで得られた知見を整理する．

本論文の構成は，以下のとおりである．2 章では，本安全分析における方針の整理，前提の整理，分析対象の整理，用いる手法の選定を行う．3 章では，ATK2-SC1 に対する分析を行う．4 章では，保護機構を実装している ATK2-SC3 について，ATK2-SC1 との仕様の違いを示し，安全分析の方針を再整理したうえで，メモリ保護の仕様を例に，実際に行った安全分析を示す．5 章では，まとめとして全体を総括する．

2. 安全分析の方針と手法

2.1 安全分析の方針

ISO 26262 では通常，ハザード分析およびリスクアセスメント（HARA）という工程において，製品を利用することで起こりうる危険事象を分析する．そして，この工程から導出される安全目標をベースに，安全要求を導出する．しかし，ATK2 はソフトウェアの動作環境である性質上，それ自身がどのような製品で利用されるかを規定しない．そのため，HARA を実施し，その結果をもとに安全要求を導出することが困難である．

そこで我々は，ATK2 を文献 [5] 第 9 節に規定されている SEooC（Safety Element out of Context）として扱うこととした．SEooC は，あらかじめどのような製品で利用されるかについて前提を設定したうえで，汎用部品として開発する考え方である．この場合，設定した前提から安全要求を導いた後，その要求を設計情報に組み込む流れとなる．このため，SEooC により開発された汎用部品を，安全性を担保しながら使用するには，あらかじめ設定した前提を満たす必要がある．今回，ATK2 に SEooC を適用する場合の前提として，安全要求，安全要求に割り当てられるレベル，想定システム構成を設定する．

安全要求は，そのリアルタイム OS としての高い汎用性を考慮し，以下 2 点とした．

- ATK2 はいずれの正常系機能が正しく動作しない場合にも製品の安全目標が侵害される．
- ATK2 上のアプリケーションが ATK2 を正しく使用しない場合，製品の安全目標が侵害される．

この 2 点のいずれかを満たさなくなる場合を，安全要求

からの逸脱と見なして，分析を行う方針とする．

安全要求に割り当てられるレベルは，文献 [4] Part 6 第 7 節において，ASIL（Automotive Safety Integrity Level）を用いて A～D の 4 段階で定義する方法が示されている（未対応を示す QM を含めると 5 段階）．ATK2 の安全要求については，あらゆる ASIL のアプリケーションがそのうえで動作することを想定し，最高水準の ASIL D と仮定する．

想定システムの構成は，分析事例の中で設定する．

2.2 安全分析の対象

ATK2 の正常系機能，および ATK2 の使用上の制約は ATK2 外部仕様書 [10] に定義されている．そのため，当該文書から，上記に関連する記述を抽出し，安全分析を行う方針とする．

ATK2 外部仕様書は，AUTOSAR OS の仕様を構成する 2 つの仕様書（AUTOSAR OS SWS，および OSEK/VDX OS Specification）の記述を統合したうえで，一部仕様について加筆修正を行い作成している．

ATK2 外部仕様書では，元となっている AUTOSAR OS SWS の仕様が，タグで管理されていることをふまえ，個々の仕様にタグを採番して管理を行っている．タグを採番する基準は，以下の 2 種とする．

① AUTOSAR OS SWS の各仕様番号に対応する記述は，そのままタグを採番する．
② それ以外の記述は，①と同等の粒度となるようにタグを採番する．

本安全分析は，ATK2 外部仕様書を対象に行った．ATK2 外部仕様書には，正常系機能の記述や，ATK2 使用上の制約以外にも，機能に関する補足説明や，外部ツール（設定情報を生成するジェネレータ等）に関する説明も含まれるため，タグを分類し，対象となる内容のみを分析することとした．分類内容を**表 1** に示す．

この分類に基づき，安全分析の対象を正常系機能と制限事項の 2 種類とする．SC3 については，保護機構に該当する異常系機能も安全要求と見なし，分析対象とする．その理由は，4.2 節で詳細に述べる．

これらの安全要求について，逸脱時の影響を分析する．影響が安全要求を侵害すると判断できる場合，逸脱の原因を分析し，その対策である安全方策を導出する．この分析過程を安全分析シートにまとめる．なお，分析時に設計情報が存在し，かつ安全方策の導出に有効であれば使用する．導出された安全方策は，アプリケーション開発側で実施するものと，OS 開発側で実施するものに分類し，前者を，アプリケーションの開発工程への入力，後者を OS の設計工程への入力とする．

2.3 安全分析の手法

本節では，安全分析の手法について述べる．本安全分析では，具体的な製品を前提としないリアルタイム OS の安全要求を扱う性質上，製品に発生する具体的な障害を扱うことなく，かつ，汎用的に逸脱を導出できる手法が必要となる．

表 1 ATK2 外部仕様書のタグ分類
Table 1 Tag classification of ATK2 external specification.

分類	説明
正常系機能 (安全要求)	・ATK2 が提供する機能
制限事項 (安全要求)	・ATK2 の使用時にアプリケーションが守るべき事項
異常系機能 (一部安全要求)	・制限事項違反に対する技術的対策．安全機構に該当する ・保護機構として扱う機能のみ，安全要求とみなす（詳しくは 4.2 で述べる）
例外処理要求	・システム誤動作対策として，OS が実装する機能(スタックモニタリング，CPU 例外対応等)
非安全機能	・デバッグ用途の機能など，安全関連システムでは不要な機能
ジェネレータ関連機能	・コンフィギュレーションを定数に変換するツールである，ジェネレータが実装すべき機能 ・ATK2 の安全要求には含めず，分析対象外とする
補足説明	・上記以外の補足的な情報

一般に安全分析の手法には，FTA，FMEA，そして HAZOP 等が存在する．しかし，FTA には実際の故障を起点に分析を行う必要があり，FMEA についても，故障モードをあらかじめ特定したうえで分析を行う必要があることから，今回の分析にはそぐわないと判断し，今回は HAZOP を用いて分析を行うこととした．HAZOP は，正常な状態に対して，その状態の逸脱を示すガイドワード（「～ない」や「～の値が小さい」）をあてはめることで，実際に起こりうる逸脱を導出し，その逸脱にともなう影響と，原因ならびに対策を分析する手法である．

ただし，HAZOP 自体は元々化学プラントにおける薬品の流れを対象としているため，ガイドワードがリアルタイム OS に対する分析に向かないという欠点がある．そこで，我々は HAZOP をベースにし，ガイドワードをソフトウェア向けに整理した SHARD（Software Hazard Analysis and Resolution in Design）[11] を用いて分析を行う．SHARD のガイドワードは，真偽値や処理のタイミングを重視する傾向があるため，論理的なコンポーネントの組合せで構成される OS に適していると判断した．

使用したガイドワードを以下に示す．なお，本分析では，SHARD のガイドワードに加えて，リアルタイム OS の要求に対する逸脱を分析するためにガイドワードを新たに付

け加えている．AUTOSAR OS は，車載システムの特に制御用途で用いられることが前提であり，処理によっては数ミリ秒の制約違反が事故につながってしまう可能性がある．そこで，我々は処理時間が長すぎたり，短すぎたりするリアルタイム制約違反に対応するため，ガイドワードとして Longer，Shorter を追加する．

- Omission：機能が提供されない．
- Commission：要求されていないときに機能が提供される．
- Early：期待されるタイミングより早く機能が提供される．
- Late：期待されるタイミングより遅く機能が提供される．
- Value：機能の出力値が間違っている．
- Longer：機能の提供までにかかる時間が長すぎる．
- Shorter：機能の提供までにかかる時間が短すぎる．

3. AUTOSAR OS SC1 に対する分析

3.1 AUTOSAR OS SC1 に対する分析方針

はじめに ATK2-SC1 の分析を行う．SC は，AUTOSAR OS における保護機構への対応状況を示す．AUTOSAR OS SC1 は，OS の基本機能のみ実装し，保護機構は実装していない．本節では，アプリケーションと ATK2-SC1 の想定構成と，それぞれに割り当てるべき安全方策の内容について論じる．

ISO 26262 の要件として，文献 [4] Part 9 第 6.4.5 節に（OS を含む）各アプリケーション間の FFI（Freedom From Interference の略．2 つ以上のエレメント間において安全要求の侵害につながるカスケード故障が存在しない状態を指す）を確保する仕組みが存在しない場合には，各アプリケーションは共存しているアプリケーションの中で最も高い ASIL で開発される必要が生じる旨の記載がある．そのため，SC1 は ISO 26262 対応上，単一 ASIL のアプリケーションのみを扱うこととなる．**図 1** に，ATK2-SC1 上でアプリケーションを動作させる場合の想定構成を示す．

ATK2 上で最高 ASIL にあたる ASIL D のアプリケーションを動作させる場合は，ATK2 とアプリケーションの全体で，ASIL D を満たすように開発する必要がある．文献 [4] Part 6 第 7.4.14 節において，ソフトウェアが ASIL D を満たすためには，ソフトウェアアーキテクチャレベルで以下のエラー検出要求を満たす必要がある旨の記述がある．

- 制御フローモニタリング
- 入出力データの範囲チェック
- 外部の動作監視機構
- プラウシビリティチェック
- ダイバース設計

このうち，制御フローモニタリングについては，AUTOSAR の基盤ソフトウェアにあたる AUTOSAR Basic Software（以下，BSW と表記）モジュールの 1 つ，WdgM（Watchdog Manager）[12] の利用により対応することが可能である．外部の動作監視機構については，システムの対処が必要であるため，対象から除外する．

入出力データの範囲チェック，プラウシビリティチェック，およびダイバース設計については，アプリケーション側でも対応を実施するため，OS でも対応を行うと冗長となりパフォーマンスが低下する．よって，アプリケーションにて対応できないエラーについてのみ OS 側に安全機構として追加する．

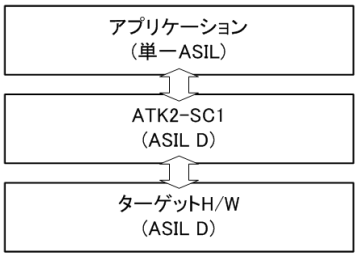


図 1 ATK2-SC1 が動作する想定システム構成
Fig. 1 Configuration of assumed system on which ATK2-SC1 is running.

3.2 基本機能の外部仕様レベル分析

ATK2-SC1 に対する SHARD を用いた実際の分析について，基本機能の 1 つであるリソース機能の仕様を例に説明する．リソース機能は，AUTOSAR OS における処理単位（タスク，C2ISR）間で排他制御を行う．本節では，ATK2 外部仕様書から，以下の仕様を選び分析を行う．

- タスク，C2ISR（Category 2 Interrupt Service Routine の略で OS の機能を利用する ISR）は同一のリソースをネストして獲得することはできない．

本仕様は，排他処理用のシステムサービス（OS の機能を呼び出す関数）GetResource に関して，ATK2 を使用するアプリケーションが守るべき事項，制限事項に該当する．表 1 の分類より，制限事項は安全要求として扱うことから，本仕様からの逸脱について分析し，その影響と原因，対策を導出する．分析結果を**表 2** に示す．表の各列の読み方は以下のとおり．

- 仕様：分析対象となる安全要求．対象となるタグを ATK2 外部仕様書から抜粋して記述する．
- ガイドワード：ガイドワードを記述する．今回の例ではスペースの都合で Omission 以外のガイドワード記述を省略している．
- 逸脱：仕様にガイドワードをあてはめた結果を記述する．これにより，どのような形で要求を満たさないかを記述する．ガイドワードをあてはめた結果，無意味な文章が生成された場合は逸脱に追加しない（例：本

表 2 リソース機能の外部仕様分析例（抜粋）
Table 2 Analysis example of external specification related to resource function.

仕様	ガイドワード	逸脱	影響	原因	対策
タスク，C2ISR は同一のリソースをネストして獲得することはできない	Omission	タスク，C2ISR が同一のリソースをネストして獲得する	OS が意図しない動作となる (安全要求を侵害する可能性がある)	OS 外の不具合	GetResource 呼出し時に指定されたリソースが，既に占有されている場合，E_OS_ACCESS を返す (COS3837)
					E_OS_ACCESS が返された場合の適切な対応をユーザ側で検討する(マニュアル ID0010)
					・・・
				H/W の不具合	・・・

仕様に Longer をあてはめた場合，「タスク，C2ISR は同一のリソースをネストして獲得することはできない時間が長すぎる」となる）．

- 影響：逸脱によりどのような形で安全が損なわれるかを記述する．もしその程度が軽微である場合は，対処不要と見なし，そこで当該逸脱に対する分析を打ち切る．
- 原因：逸脱の原因を導出する．これは，逸脱を防ぐために対処すべき点を示している．今回の例では，表 2 に抜粋した「OS 外の不具合」の他に「H/W の不具合」も原因として想定する．もし，原因を掘り下げる必要がある場合は，その理由を何段階にもわたって掘り下げる．
- 対策：逸脱内容および原因をふまえて，逸脱への対策を立案し表に記述する．

前節の議論をふまえ，ATK2-SC1 が動作するシステムでは，原則としてアプリケーション側で逸脱への対応を実施し，OS 側には安全方策を追加しない方針とする．具体的には，アプリケーション側で対処すべき内容が導出されるため，これを開発者向け安全マニュアルに記載することとする．また，ATK2-SC1 の別の仕様（主に異常系機能）において，逸脱への対処が行われている場合は，これも対策として対策欄に記載することとする．

上記の方針に従い，ATK2-SC1 の各外部仕様について，逸脱の影響の精査から，原因の分析，対策の導出までを行った．

3.3 AUTOSAR OS SC1 に対する分析結果

ATK2-SC1 に対して行った安全分析の作業をまとめると以下のとおりとなる．

- 各異常系機能が，安全要求（正常系機能，制限事項）の逸脱に対応する安全機構として機能することを確認した．
- 各制限事項をアプリケーション側で守るべきルールとしてまとめ，安全マニュアルに記述した．
- 逸脱原因のうち，OS 自身の不具合や，ハードウェア不具合によるものについては，アプリケーション側で実装すべき安全機構の指針としてまとめ，安全マニュアルに記述した．

本作業を ATK2-SC1 の外部仕様全体に対して行ったことにより得られた知見を以下に示す．

- 分析対象としたほぼすべての要求仕様について，Omission による影響導出が有効であった．この理由は，元の要求仕様を単純に否定するガイドワードなので，影響が重大な結果につながることが多いためであると考えられる．

4. AUTOSAR OS SC3 に対する分析

4.1 AUTOSAR OS SC1 との仕様の違い

次に ATK2-SC3 の分析を行う．AUTOSAR OS SC3 は，同一のアプリケーションを構成する OS オブジェクトをグループ化するための仕組みである OS アプリケーション（以下，OSAP と表記）を導入している．

OSAP は権限レベルに応じて信頼 OSAP と非信頼 OSAP の 2 種類に分類される．信頼 OSAP は原則 OS 上のすべての OS オブジェクトにアクセス可能だが，非信頼 OSAP は原則自 OSAP に属する OS オブジェクトへのアクセスしか認められず，システム全体に影響を及ぼす操作（全割込み禁止，OS シャットダウン等）を行うことも認められない．この権限レベルを活用することにより，信頼度が異なる複数のアプリケーションを管理することが可能になる．

SC3 では OSAP のほかに，保護機構を導入している．AUTOSAR は，複数の開発元による OSAP が同一のプロセッサ上で共存することをプラットフォームの要求に含めている．保護機構は，ある OSAP のフォールトが無関係な OSAP に伝播することを防ぐことにより，OSAP の共存を

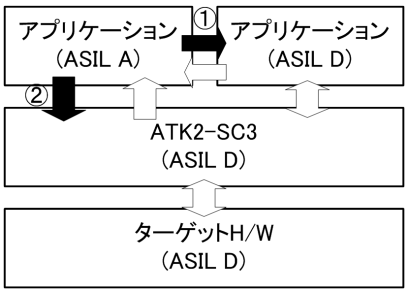


図 2 ATK2-SC3 が動作する想定システム構成
Fig. 2 Configuration of assumed system on which ATK2-SC3 is running.

可能にするための仕組みである．SC3 で追加される保護機構は，主にサービス保護とメモリ保護の 2 つに大別される．

(1) サービス保護

サービス保護は，システムサービスレベルでのアクセス保護を行う．サービス保護は大きく 3 種類に大別される．1 つ目は，非信頼 OSAP 等，OSAP の処理単位（タスクまたは C2ISR）から，信頼 OSAP を含む他 OSAP に対し，アクセス権を付与されていない状態でシステムサービスを実行した場合に，その処理要求を拒否する仕組みである．2 つ目は，システムサービスを不適切な形（割込み禁止状態での呼び出し，不適切な引数の使用等）で呼び出した場合，その処理要求を拒否する仕組みである．3 つ目は，非信頼 OSAP から他の全 OSAP に影響を与えるシステムサービスの実行（OS シャットダウン要求，全割込み禁止処理等）を拒否する仕組みである．

(2) メモリ保護

メモリ保護は，メモリレベルでのアクセス保護を行う．各非信頼 OSAP の単位で，アクセス（読み，書き，実行）可能なメモリ領域をあらかじめ定義しておき，非信頼 OSAP がアクセスを許可されていない領域に対して，アクセスを行った場合に，MPU（Memory Protection Unit）によりこれを検知し，アクセスを遮断する仕組みである．

4.2 AUTOSAR OS SC3 向け安全分析の方針

複数の OSAP が ATK2-SC3 上で共存しつつ動作する構成について分析を行う．ここでは，2 種類の ASIL アプリケーション，具体的には低 ASIL（ASIL A 等）と高 ASIL（ASIL D 等）のアプリケーションが共存するケースを想定する．想定構成を**図 2** に示す．

OSAP 共存にあたっては，高い ASIL を割り付けられた OSAP に対し，より低い ASIL を割り当てられている OSAP が影響を及ぼさないことを担保する必要がある．図 2 において，低 ASIL のソフトウェアが，高 ASIL のソフトウェアに対してアクセスするパターンは 2 種類存在する．

① 低 ASIL 側の OSAP ⇒ 高 ASIL 側の OSAP

② 低 ASIL 側の OSAP ⇒ OS（ASIL D）

それぞれのパターンについて，AUTOSAR が提供する保護機構を整理し，FFI を満たすための条件を確認する．なお，高 ASIL 側の OSAP が OS より ASIL が低い場合，高 ASIL 側の OSAP（ASIL C）⇒ OS（ASIL D）のようなパターンも生じうるが，今回は②に包含する形で分析を行う．図 2 における①と②に対してそれぞれサービス保護，メモリ保護を実現することにより，権限を持たない非信頼 OSAP に対して，API レベルでのアクセスを拒否したうえで，API によらない手段で直接メモリにアクセスしてくることを排除することが可能となる．

今回の分析は，FFI の実現にあたり，OS が持つ保護機構に含まれる安全要求を分析し，安全方策を導出することを目的とする．保護機構の仕様は異常系機能に分類されるが，仕様から一度逸脱するだけで即座に安全要求に違反することから，安全分析の対象として逸脱時の影響を分析する．分析の結果導出した安全方策に対応することにより，ATK2 上で複数 ASIL のアプリケーションを安全に共存させることが担保可能になる．

4.3 メモリ保護の外部仕様レベル分析

ATK2-SC3 に対する SHARD を用いた実際の分析について，メモリ保護の仕様を例に説明する．本論文では，ATK2 外部仕様書のメモリ保護の章から以下の仕様を選び分析を行う．

- 非信頼 OSAP の専有コード領域に対して，他の非信頼 OSAP が読み出し，書き込み，実行アクセスすることを禁止する．

外部仕様レベルの分析結果を**表 3** に記載する．分析手順を以下に示す．仕様に対し，ガイドワードを用いて逸脱にともなう影響および逸脱の原因を導出する．例では，書き込み処理について，Omission をあてはめた場合の影響を導出している．影響を分析した結果，アクセス先の OSAP に干渉し，安全要求に反することが確認できるため，以降で原因の導出を行う．表 3 では，原因 1～原因 3 まで 3 段階分掘り下げて分析を行っている．ここで導出した原因 1，2 について，原因 3 において OS に起因する不具合と H/W に起因する不具合（H/W の中で MPU については独立で扱う）に分ける．H/W に関する対策はユーザに対応を委ね，OS に関する対策は OS 開発者側に対応を委ねる方針とする．

分析対象の安全要求の粒度によっては，原因を分析した結果，有効な対策が導出できない場合がある．この場合，さらに詳細な設計情報を参照して，原因の分析，対策の導出を行う．その時点で具体性をともなった対策を導出できた場合は，当該安全要求に対する分析を終了する．導出できなかった場合は，対策が導出できるまで，設計の粒度を細かくして分析を繰り返す．

表 3 メモリ保護の外部仕様に関する分析例（抜粋）
Table 3 Analysis example of external specification related to memory protection function.

仕様	ガイドワード	逸脱	影響	原因 1	原因 2	原因 3
非信頼 OSAP の専有コード領域に対して，他の非信頼 OSAP が読出し，書込み，実行アクセスする事を禁止する	Omission	書込みアクセスを禁止しない	タスク，C2ISR が自身の所属する非信頼 OSAP とは異なる非信頼 OSAP の専有コード領域に対して書込みを行い，アクセス先が意図しない動作となる	当該アクセスを許可すべきと誤認した	アクセス元の OSAP が信頼 OSAP であると誤認した	OS の不具合
						H/W 故障
					別の信頼 OSAP からのアクセスであると誤認した	OS の不具合
						H/W 故障
				書込みアクセスを禁止しなかった		OS の不具合
						MPU 故障
						H/W 故障

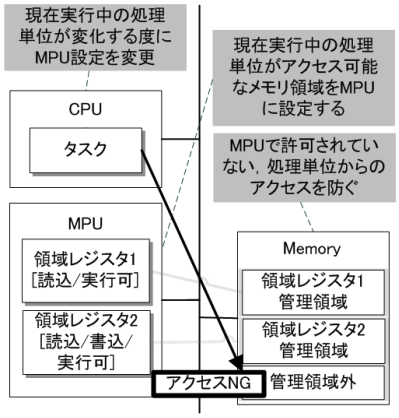


図 3 メモリ保護のアーキテクチャ設計情報
Fig. 3 Architecture design information of memory protection function.

4.4 メモリ保護のアーキテクチャ設計レベル分析

外部仕様に対する分析の結果，OS の不具合に起因する逸脱の可能性についてさらに分析を行うこととした．OS の不具合の内容をより明確にし，具体的な対策を検討するためには，保護機構の設計に対する分析が必要である．本節では外部仕様の分析で導出された原因をより詳細に分析するため，アーキテクチャ設計を分析することとする．アーキテクチャ設計レベルのメモリ保護の概要を**図 3** に示す．

メモリ保護に関する設計情報の中で，本外部仕様に関連する内容をまとめた内容を以下に示す．

- ATK2-SC3 では，タスクをはじめとする OS の処理単位は MPU を介してメモリ領域にアクセスする．MPU で許可されていない領域にアクセスした場合，アクセスが遮断され CPU 例外が発生する．
- 非信頼 OSAP の処理単位を実行中は，非特権モードでプロセッサを動作させ，その処理単位からアクセス可能なメモリ領域を MPU に設定することで，メモリアクセスを制限する．

これらは，OS 機能の 1 つであるディスパッチャが処理単位を切り替える際に以下の一連の処理を順に実行することにより実現する．

1) MPU にアクセス可能なメモリ領域をセット．
2) 動作モードを CPU のステータスレジスタにセット．
3) return 命令を用いて動作モードを切り替える．

切替え後の処理単位からアクセス可能なメモリ領域は，以下の 2 種類である．

- その処理単位の持つユーザスタック
- その処理単位が属する OSAP からアクセス可能なメモリ領域

以上の設計を分析した結果，メモリ保護仕様の逸脱の原因となりうる不具合を（ア）～（ウ）に分類した．

（ア） 定数設計の誤り
（イ） ジェネレータ設計の誤り
（ウ） 処理設計の誤り

（ア）と（イ）は，システムコンフィギュレーションにおける不具合である．（ウ）は，メモリ保護機構の設計の不具合である．後者については，具体的な対策を検討するために，次節で，さらに詳細な設計に対して分析する．

（ア）と（イ）では，コンフィギュレーションが適切な OS の定数に変換されない，という設計上の不具合を扱う．（ア）では，設計時に適切な定数を定義していないために，ユーザが意図する設定情報を格納する定数がない，という不具合を扱う．（イ）では，ジェネレータの設計が誤っているために，ユーザが意図した設定情報が本来格納されるべき定数に格納されない，という不具合を扱う．

これらの不具合に対して，ユーザの意図どおりの管理データが生成されているか否かを以下のいずれかの方法で確認することを提案する．これらの方法は，**図 4** に示されるように，レビューの対象とタイミングが異なるだけであり，本質的には同じである．

- 生成されたメモリマップ情報（アクセス可能なメモリ領域情報）を取得しレビュー．
- 生成されたメモリマップ情報に基づきアクセス可否をテスト（領域境界に対してアクセスし，メモリ保護例

外が発生するかどうかをテストする）．

本対策の実施により，コンフィギュレーションの意図に基づくメモリ保護を行うことが可能となる．概要を図 4 に示す．

（ウ）の対策について検討したが，対策にあたり抑えるべき内容が「切替え先に応じて正しく動作モード，MPU 設定を行うこと」であるため，まだ設計情報として粒度が粗く，対応する安全対策を導出することは困難と判断する．このため，ユニット設計レベルについて，さらに分析を行うこととする．

4.5 メモリ保護のユニット設計レベル分析

メモリ保護機構の処理設計の誤りの原因について，ユニット設計レベルでの分析を行う．ここでは，アーキテクチャ設計における「1) MPU にアクセス可能なメモリ領域をセットする」について，処理の流れを取り上げる．

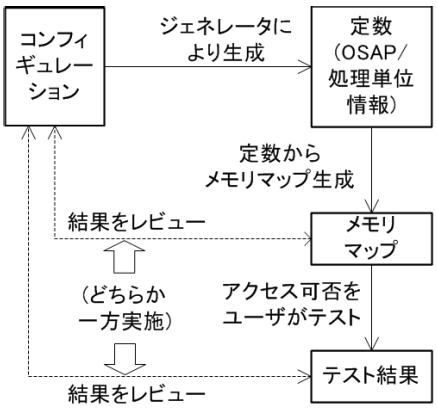


図 4 定数，ジェネレータの安全方策
Fig. 4 Safety measures of constants and generators.

ユニット設計レベルまでの分析結果を**表 4** に記載する．原因 4 にはアーキテクチャ設計レベルで導出した原因，原因 5 にはユニット設計レベルで導出した原因，対策には導出された安全方策を記載している．以下，原因 4 から原因 5 を導出するまでの流れを説明する．

ユニット設計によると，MPU へのメモリ領域セット処理は以下の流れで行われる．

1) 実行中の処理単位が属する OSAP が切替え先の OSAP と等しいかどうかをチェックする．等しい場合は，切替え処理を行わずに終了する．
2) MPU にメモリ保護対象として設定した処理単位と現在動作している処理単位が等しいかどうかをチェックする．
3) 2) の結果，等しくない場合は，処理単位レベルの MPU 設定レジスタに切替え先処理単位の MPU 設定情報をセットする．
4) MPU にメモリ保護対象として設定した OSAP 情報と現在動作している OSAP 情報が等しいかどうかをチェックする．
5) 4) の結果，等しくない場合は，OSAP MPU 設定レジスタに，切替え先 OSAP の MPU 設定情報をセットする．

上記過程において発生しうる不具合は，以下の 2 点に収束する．なお，動作モード設定処理で発生しうる不具合も，

表 4 メモリ保護のアーキテクチャ設計およびユニット設計に関する分析例（抜粋）
Table 4 Analysis example of architecture design and unit design related to memory protection function.

逸脱	原因 2	原因 3	原因 4	原因 5	対策
書込みアクセスを禁止しない	アクセス元の OSAP が信頼 OSAP であると誤認した	OS の不具合	ジェネレータ設計誤り		メモリマップレビュー又はテストをユーザにて実施し，意図通りである事を確認する
			定数設計誤り		
			処理設計誤り	所属 OSAP の属性情報取得間違い	左記観点のレビュー・テストを実施する
				・・・	
	別の信頼 OSAP からのアクセスであると誤認した	OS の不具合	ジェネレータ設計誤り		メモリマップレビュー又はテストをユーザにて実施し，意図通りである事を確認する
			定数設計誤り		
			処理設計誤り	所属 OSAP の情報取得間違い	左記観点のレビュー・テストを実施する
				・・・	
	書込みアクセスを禁止しなかった	OS の不具合	ジェネレータ設計誤り		メモリマップレビュー又はテストをユーザにて実施し，意図通りである事を確認する
			定数設計誤り		
			処理設計誤り	MPU レジスタへの設定間違い	左記観点のレビュー・テストを実施
				・・・	

分析の結果以下の 2 点に収束している．

- 変数判定（所属情報の確認処理）の間違い
- レジスタの設定間違い

この 2 点の根本原因に対する対策として，以下の 2 種類が考えられる．

（ア） 安全機構の追加
（イ） 設計とソースコードに対する検証（レビューおよびテスト）で確認

今回我々は，この対策として（イ）を選択する．これは，以下の理由による．

- レビューで確認できる内容，量であると判断したため．
- 実装の複雑化を避けるべきと判断したため．

そこで，これらのレビューの観点をまとめたレビューチェック表を作成し，OS の設計工程のインプットとする．

4.6 AUTOSAR OS SC3 の安全分析結果

ATK2-SC3 のメモリ保護機能に対して行った安全分析の作業をまとめると以下のとおりとなる．

- メモリ保護機構に関連しないタグについては，ATK2-SC1 と同様の分析を実施．
- メモリ保護機構に関連するタグについては，外部仕様レベルからアーキテクチャ設計，ユニット設計まで掘り下げて分析を実施．
- 逸脱を防ぐための安全方策として，OS 開発側で必要となるレビュー項目を洗い出し，チェックリスト化．

サービス保護や保護機構に属さない各機能についても，上記と同様の方針で安全分析を行った．

本作業を ATK2-SC3 の外部仕様全体に対して行ったことにより得られた知見を以下に示す．

- 分析対象となる安全要求の粒度によっては，逸脱の原因を分析した後，具体的な対策まで導出できないことが生じる．本事例ではすでに設計情報が存在したため，これをベースに対策を導出することができた．
- もし，設計情報が存在しない場合は，本分析で対策が必要となった箇所について，設計段階で対策を検討して実現する流れとなると考えられる（この場合，対策欄に記載される内容は「OS の設計で対策する」という抽象的な表現にとどまることになる）．

5. おわりに

我々は，ATK2 に対して，今回提案する安全分析方法を適用することにより，安全方策を導出した．

ATK2-SC1 の基本機能に対して分析を行い，アプリケーション側で対処すべき安全方策を導出し，安全マニュアルにまとめた．ユーザがこのドキュメントを参照して開発することにより，OS を機能安全に対応させることが可能になった．

また，ATK2-SC3 の保護機構に対して分析を行い，OS が対処すべき安全方策を導出し，開発者向けのチェックリストにまとめた．OS 開発者がこのチェックリストに基づいて開発を行うことにより，複数のアプリケーションが共存するケースにおける FFI を担保することが可能になった．

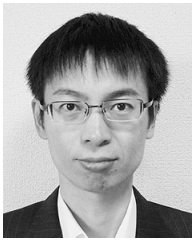
本研究の課題について示す．今回分析した内容は AUTOSAR OS に限定されるため，Platform 全体として機能安全対応を行うには，AUTOSAR RTE [13] と連携して SW-C（AUTOSAR Platform におけるアプリケーションの基本単位）を動作させるケースの分析や，通信ミドルウェア等のリアルタイム OS 以外の共通ソフトウェアの分析が必要になる．

参考文献

[1] AUTOSAR, available from ⟨http://www.autosar.org/⟩ (accessed 2016-09-07).
[2] AUTOSAR, Specification of Operating System AUTOSAR Release 4.2.2 (2015).
[3] TOPPERS プロジェクト，TOPPERS/ATK2，入手先 ⟨https://www.toppers.jp/atk2-download.html⟩（参照 2016-06-21）．
[4] ISO, ISO 26262: Road vehicles – Functional safety, Part 1〜9 (2011).
[5] ISO, ISO 26262: Road vehicles – Functional safety, Part 10 (2012).
[6] AUTOSAR, Overview of Functional Safety Measures in AUTOSAR (2014).
[7] exida, Results of the ISO 26262 and IEC 61508 Functional Safety Assessment, available from ⟨http://www.exida.com/2015/EBA_13-04-113_R012_V1R3_Assessment_OS_1_1_x.pdf⟩ (accessed 2016-11-05).
[8] exida, Results of the ISO 26262 Functional Safety Assessment, available from ⟨http://www.exida.com/2016/uploads/Vector_1410-091-C_R004_Assessment_Report_SafeBSW_and_OS_V1R0.pdf⟩ (accessed 2016-11-05).
[9] exida, Results of the ISO 26262 Functional Safety Assessment, available from ⟨http://www.exida.com/2016/uploads/KPIT_1309-020-C_R002_Assessment_Report_OS_V1R0.pdf⟩ (accessed 2016-11-05).
[10] AP コンソーシアム：ATK2 外部仕様書，入手先 ⟨https://www.toppers.jp/docs/tech/ATK2-0010_ATK2_spec_120.pdf⟩（参照 2016-06-21）．
[11] David, J.P.: The Principled Design of Computer System Safety Analyses, Ph.D. Thesis, The University of York (1999).
[12] AUTOSAR, Specification of Watchdog Manager AUTOSAR Release 4.2.2 (2015).
[13] AUTOSAR, Specification of RTE AUTOSAR Release 4.2.2 (2015).

毛利 守男

2001年株式会社CSK（現，SCSK株式会社）入社．主に通信キャリア向けエンタープライズシステム，および車載制御システムの開発に従事．2014年10月より2017年3月まで名古屋大学大学院情報科学研究科付属組込みシステム研究センター共同研究員として，車載制御システム向け高品質プラットフォームに関する研究に従事．



佐藤 秀昭

2008年株式会社ジェイテクト入社，電動パワーステアリングのソフトウェア開発に従事．2014年9月より名古屋大学大学院情報科学研究科附属組込みシステム研究センター共同研究員として，車載制御システム向け高品質プラットフォームに関する研究に従事．2016年6月復任，現在に至る．車載ソフトウェアプラットフォーム開発に従事．

山下 映

1992年日本電気通信システム株式会社入社．主にキャリア向け交換システムのリアルタイムOS開発，保守に従事．2015年4月より2017年3月まで名古屋大学大学院情報科学研究科付属組込みシステム研究センター共同研究員として，車載制御システム向け高品質プラットフォームに関する研究に従事．

松原 豊

名古屋大学大学院情報学研究科附属組込みシステム研究センター助教．2006年名古屋大学大学院情報科学研究科博士前期課程修了．同大学院情報科学研究科附属組込みシステム研究センター研究員，特任助教を経て，2013年より現職．2015年4月～9月米国ワシントン大学およびカリフォルニア大学サンディエゴ校客員研究員．組込みシステム向けのリアルタイムOS，リアルタイムスケジューリング理論，安全技術，セキュリティ等の研究に従事．博士（情報科学）．IEEE，USENIX，電子情報通信学会各会員．

高田 広章

名古屋大学未来社会創造機構教授．同大学大学院情報学研究科教授・附属組込みシステム研究センター長を兼務．1988年東京大学大学院理学系研究科情報科学専攻修士課程修了．同専攻助手，豊橋技術科学大学情報工学系助教授等を経て，2003年より名古屋大学大学院情報科学研究科情報システム学専攻教授．2014年より現職．リアルタイムOS，リアルタイムスケジューリング理論，組込みシステム開発技術等の研究に従事．オープンソースのリアルタイムOS等を開発するTOPPERSプロジェクトを主宰．博士（理学）．IEEE，ACM，電子情報通信学会，日本ソフトウェア科学会，自動車技術会各会員．