ホワイトペーパー

ISO 26262 準拠のために気を付けたい 5 つの落とし穴

モデルベースデザインを実践するためのヒント

MathWorks®

はじめに

ISO 26262 は、自動車の安全関連システムを開発するための事実上の業界標準として急速に普及しています。現在、安全関連機能に携わるエンジニアのほとんどが ISO 26262 を採用しています。

開発組織は、このガイダンスを利用し、個々のプロセスや目標に適応させることで、ISO 26262 への準拠を加速させることができます。その際には、以下の点を考慮することが必要です。

- ISO 26262 規格で推奨される機能安全活動
- アプリケーションの自動車用安全度水準 (ASIL)
- 開発プロセスで使用するツール

このホワイトペーパーでは、組織の ISO 26262 要件遵守の妨げとなり、ISO 26262 の評価での失敗を引き起こす可能性のある、よくある落とし穴について説明します。また、これらの落とし穴を回避するための推奨事項についても説明します。

ISO 26262 プロセスの落とし穴

ISO 26262 規格は、機能安全の要件管理からソフトウェア開発プロセスで必要な属性まで、12 のセクションで構成されています。ソフトウェア開発に関連する要件のほとんどは、規格の第 6 部に定められています(ISO 26262-6: 2018)。開発プロセスにモデルベースデザイン (MBD、モデルベース開発) を使用することを計画している組織は、どこで、どのようにモデルベースデザインを使用するかを決定しなければなりません。この作業を容易にするために、MathWorks IEC Certification Kit は、MATLAB® および Simulink® 製品を使用して完全な ISO 26262-6 準拠のワークフローを実現する方法についてのガイダンスを提供しています。

IEC Certification Kit は、プロセス移行の取り組みの開始点として使用する必要があります。このキットには、モデルベースデザインを使用して ISO 26262-6 に準拠する方法についての概要、ドキュメンテーション テンプレート、リファレンス ワークフローが用意されています。

内容を理解したうえで、以下の手順で既存のプロセスと ISO 26262-6 に準拠したプロセスとのギャップ分析を行います。

1. 開発する電気システムとコンポーネントの ASIL 要件を決定する。
2. ISO 26262-6 規格のどのプロセス要素に従うかを決定する。
3. これらのプロセス要素を組織の既存の開発プロセスにマッピングする。
4. ギャップや改善すべき領域を分析して理解する。
5. プロセス改善戦略と実装計画を定義する。

ギャップ分析は、あらゆるプロセス変革活動に不可欠な手順です。しかし残念なことに、多くの組織は ISO 26262 プロジェクトを開始する際に求められる厳密さを過小評価しています。このような組織は「アップデートしながら進める」という考え方をしているため、プロジェクトが遅れることになったり、製品の認証時に手順を踏んでいないことに気づいたりします。

MathWorks Consulting Services は、モデルベースデザインの ISO 26262 ギャップ分析を数多く実施してきました。このサービスの目標は、開発組織の ISO 26262-6 に準拠したプロセスやツールの使用方法を、実装レベルまで客観的に把握することにあります。このサービスは、プロセスのコンプライアンスに関する問題点を指摘するだけでなく、ツールの推奨される使用方法を提案します。これらの推奨事項には、多くの場合プロセスにとどまらず、モデル化やツールの適用に関するベストプラクティスも含まれています。

MathWorks は、組織と協働する中で、以下の 5 つの落とし穴を発見しました。

1. 設計と実装のためのソフトウェア アーキテクチャ戦略が欠如している
2. 明確に定義およびマッピングされた ISO 準拠のプロセスがない
3. 自動化インフラストラクチャの環境がない
4. 作業成果物を記録する戦略がない
5. サプライヤー提供のツール検定キットを活用していない

1. 設計と実装のためのソフトウェア アーキテクチャ戦略が欠如している

機能安全チームとアーキテクチャチームは、製品の機能安全目標を達成するための戦略を持つことが重要です。ISO 26262 では、ソフトウェアの開発には主に次の 2 つの方法が認められています。

1. オプション 1: すべてのソフトウェアユニットとコンポーネントを、システムに必要な ASIL の最高評価で開発する。
2. オプション 2: 開発プロセスを、異なる ASIL と品質管理 (QM) 評価に分ける。

すべてのソフトウェアユニットに単一のプロセスを使用できるという単純な理由で、オプション 1 を選択する組織もあります。これは最初は良い選択のように思えますが、実際にはそのようなアーキテクチャ設計のシンプルさよりも、作業量の多さの方が問題となります。そのため、多くの機能安全チームやアーキテクチャチームはオプション 2 を選択しています。オプション 2 では、依存性エラーおよび故障モード影響解析 (FMEA) などの安全解析手法を使用して ASIL A から D に分類される可能性のある高評価のユニットをいくつか保有し、残りのソフトウェアを QM ユニットとして扱うことができます。

解析後に、異なるレベルの ASIL コンポーネントおよび QM コンポーネントが適切に分割されるように、電気システムとソフトウェアを構築する必要があります。この分割の概念は、無干渉と呼ばれ、ISO 26262-6: 2018 付表 E に詳細が記載されています。無干渉は、ある ASIL が他の ASIL と相互作用して他の ASIL の機能を低下させたり侵害したりしないことを確保するために必要です。付表 E には、考慮すべき 3 つの主要な概念が示されています。

- タイミングと実行
- メモリ
- 情報の交換

リアルタイム OS ベースの組み込みシステムのタイミングと実行は、詳細に解析する必要があります。メモリと情報の交換は、ソフトウェアユニットの設計および開発中に慎重に検討し、管理する必要があります。経験上、開発組織は概念的な設計を作成することはできても、設計を Simulink 内の正しい実装に変換することができないということがよくあります。モデル参照、メモリセクション、情報交換のためのカスタムメソッドなど、分割の目的を満たすために必要なモデル構造やモデル構成があります。MathWorks は、これらの推奨されるモデル構造およびモデル構成を *Simulink による ISO 26262 アプリケーション開発のための 11 のベストプラクティス*に文書化しています。

「Simulink による ISO 26262 アプリケーション開発のための 11 のベストプラクティス」で取り上げられたトピック

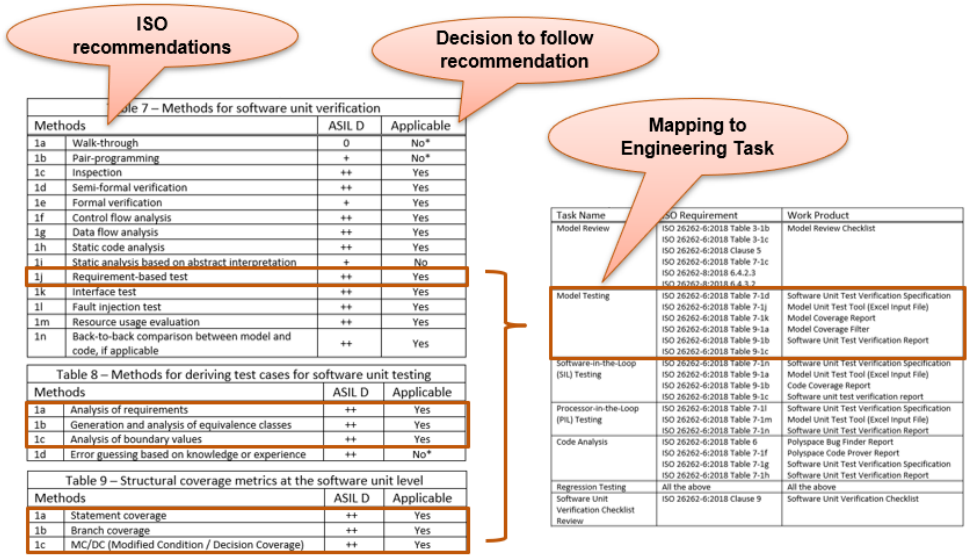
このホワイトペーパーで紹介されているベストプラクティスは、安全度水準が混在するアプリケーションなど、さまざまなアプリケーションのニーズに基づいて、必要に応じて使用し、カスタマイズすることができます。このようなガイダンスがない場合、多くの組織は、無干渉の概念を達成するために、大幅な修正を必要とする状態に陥ります。これを開発の開始前に管理することで、やりなおし作業や検証作業を大幅に削減できます。

2. 明確に定義およびマッピングされた ISO 準拠のプロセスがない

自動車業界では、Tier 1 サプライヤーや OEM メーカーの多くが、高品質なソフトウェアを開発できる経験豊富なエンジニアを擁しています。ただし、これら多くの開発組織には、設計全体、開発プロセス、またはその両方を説明する適切なドキュメンテーションがありません。代わりに、組織はエンジニアの「正しい行い」に頼っています。このような開発組織は ISO 準拠の監査を受けた場合、おそらく不合格になるでしょう。なぜなら、エンジニアは通常の活動を重視し、設計上の意思決定および結果の根拠にあまり関心がないからです。完成度およびデューデリジェンスは、優先度や重要度を決めるエンジニアの経験に左右されることになります。この問題を回避するため、開発組織は、事前に ISO 準拠のプロセスを定義し、定義されたすべての活動が確実に実行されるようにし、かつその結果が活動目標の結果と一致していることを証明する根拠を示さなければなりません。

MathWorks が目にするもう一つのよくある問題は、開発組織が ISO 要件の詳細を確認せずに、既存のプロセスを文書化する傾向があるということです。これは、通常、製品が成熟し、評価の最終準備段階に達したときに発覚します。この手法では、やり直し作業、再設計、または生産プログラムの遅延が起きることがよくあります。機能的な安全規格を対象としたアプリケーションでは、プログラムを開始する際に、開発組織のプロセスを実際の規格にマッピングすることが重要です。例えば、ISO 26262-6:2018 には、約 90 の異なる原則およびメソッド、ならびに組み込みソフトウェア開発に関する幅広い推奨事項があります。開発組織はまず、ISO 規格のどの活動を実行するかを決定し、それらの活動を内部プロセスにマッピングし、それぞれの活動の証拠としてどの作業成果物を記録すべきかを指定する必要があります。

以下のサンプル文書で、ISO 26262 セクション 6 のさまざまな表と、それを開発組織の内部プロセスにマッピングした例を示します。



ISO 26262 セクション 6 を開発組織のプロセスにマッピングした例。

このようなマッピング文書は、不足している活動や機能を表示するギャップ分析サービスに含まれる場合があります。MathWorks Consulting Services では、量産プログラムを継続する場合に使用する実装プランに、お客様と一緒に取り組むことが可能です。このような手順が重要なのは、開発組織が監査を受ける場合に、この文書が ISO 26262 の活動が行われていることを実証する手段となり、また安全計画の証拠の一部としても使用することができるからです。この文書には、いくつかの活動が省略された理由と、対応する作業成果物が構成管理システムのどの場所に格納されるかに関する情報が記載されます。

そのため、開発組織がすべてのエンジニアリング活動について、以下の 3 つの主要な手順を計画することが非常に重要です。

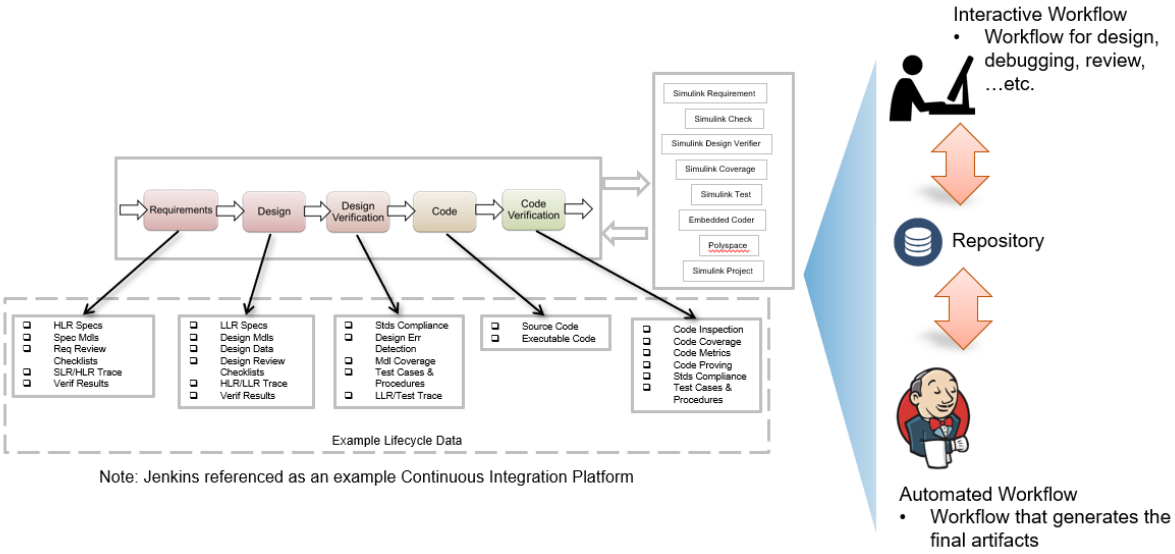
- **定義:** 開発サイクルの各段階で必要な活動について記述したプロセス文書を作成する。
- **実行:** これらの活動を継続的に実行する。コンプライアンスを確保するために、チェックリストや設計のレビューが必要です。
- **記録:** 開発サイクルの各段階で記録する必要のある作業成果物のリストを特定する。

ISO に準拠したプロセスは、製品開発サイクルの可能な限り早い段階で定義する必要があります。これらのタスクは、設計上の意思決定や実装方法を推進する場合があります。早い段階でプロセスを定義することで、評価中や量産開始前の不要なやり直し作業を回避することができます。

3. 自動化インフラストラクチャおよび環境がない

初めて ISO に準拠したプロジェクトを実行している開発組織からの一般的なフィードバックは、必要とされる計画と実行の厳密さが彼らの想定を大きく上回っていたというものです。いくつかの活動が開発組織にとって初めてであるというだけでなく、タスクの実行や作業成果物の管理に関するプロセスが追加されると、組織が他の業務にかける時間的余裕に制約が加わる可能性があります。そのため、ISO 全体の戦略の一貫として、自動化を考える必要があります。開発プロセスは、開発者の時間の大半がアルゴリズムの作成と、そのアルゴリズムを要件に照らして検証するためのテストケースの作成に費やされるように最適化する必要があります。それ以外の部分はすべて自動化します。自動化は、ISO 準拠を向上させるための業界共通の手法です。

自動化の手段または前提条件は、前述のようにインプット、活動、アウトプットを明確に定義し、文書化した定義済みのプロセスがあることです。ここで重要なのは、自動化は 2 つの異なるモードで行う必要があるということです。まず、ユーザーが取り組むべき内容を理解できるようにするには、デスクトップ自動化ツールも考慮した対話型のワークフローを含める必要があります。これを実現する簡単な方法は、プロジェクトのニーズに基づいて作業成果物を自動的に構成する API および GUI のセットを用意することです。これらには、モデル/コードのガイドライン、モデルまたはコード生成の構成、またはレポートのテンプレートが含まれる場合があります。第二に、デスクトップの自動化を通してサポートされる一連の活動は、継続的インテグレーション (CI) 型のワークフローでも利用できることが必要です。

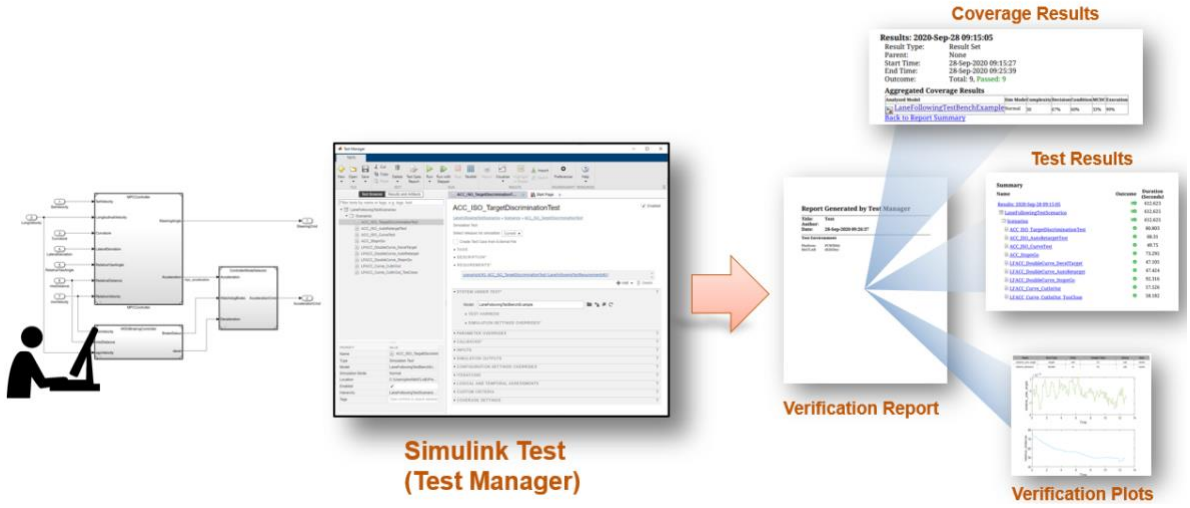


対話型のワークフローと自動化されたワークフローの両方に対する自動化のサポート。

デスクトップ自動化のサポートにより、ユーザーはネイティブな Simulink 環境でモデルを操作することができ、「ワンクリック」レベルの自動化を通して必要な構成を簡略化することができます。こうすることで、開発者はアルゴリズムの作成に集中し、設計ガイドラインに基づいてアルゴリズムの見直しまたは更新を行うことができます。対話型のワークフローによってセキュリティ層が追加されるため、最終的な作業成果物が正しく生成され、かつ定義されたガイドラインやメトリクスに照らして客観的に検証されることが確実となります。

4. 作業成果物を記録する戦略がない

ISO 26262 に準拠したワークフローを導入する場合は、どの作業成果物を構成管理システムに格納すべきかを決定することが重要です。また、CI サーバーでプロセスの一部が自動化されている場合や、手動で結果を確認している場合でも、これらの活動が完了したことを証明するために、構成管理システムにこれらの活動の証拠を保存する必要があります。これを行うためには、各チェックリストやプロセス文書で、どの成果物を保存すべきかを詳しく説明する必要があります。



作業成果物を伴う Simulink Test ワークフローの例。

上のスクリーンショットは、テストケースの実行を証明するために、構成管理システムで生成および保存が可能な検証レポートの例を示しています。この作業成果物は、ISO 監査中に、リリース チェックリストの一部として、または指定されたリリースがテストされたことを確認する必要がある場合に、後でレビューすることができます。使用された作業成果物や再現が必要な場合を詳細に説明するベースライン戦略が必要ですが、段階的な設計や検証レポートがすべて必要なわけではありません。

コンサルティング サービスの現場では、お客様が必要な作業成果物をすべて取得することに困難を感じているのをよく目にします。ただし、この情報は、ISO 26262 パート 6、モデルベースデザインのワークフロー、それに対応する MathWorks ツールチェーンの出力を十分に理解しさえすれば得られるものです。ここで、MathWorks IEC Certification Kit が有益なリソースとして役立ちます。このキットでは、トピック、原則、メソッドの表に対して、適用可能なツールをマッピングしています。また、開発者が高水準の ISO 26262-6 ソフトウェア開発のワークフローを可視化するのに役立つリファレンス ワークフローも用意されています。マッピング表とリファレンス ワークフローの例を以下に示します。

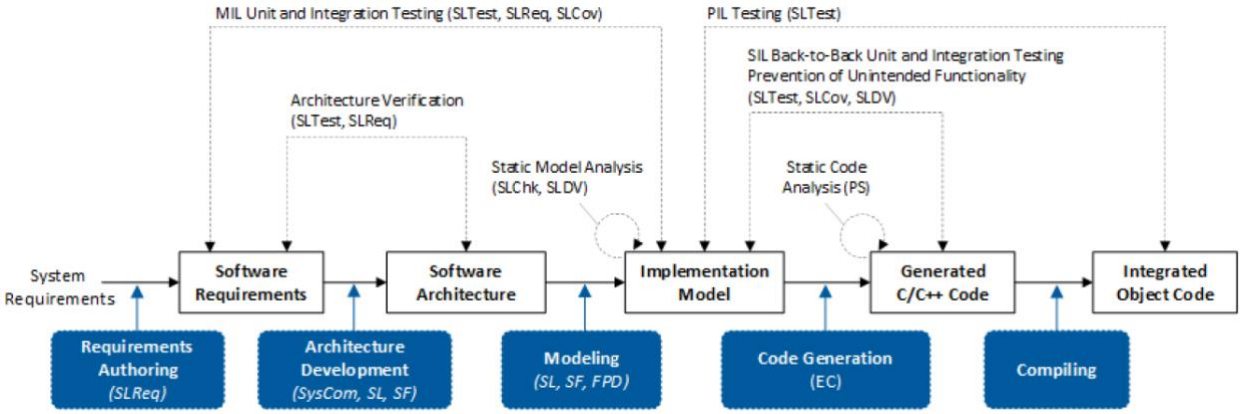
Table 9 — Structural Coverage Metrics at the Software Unit Level

Methods		ASIL A	ASIL B	ASIL C	ASIL D	Applicable Model-Based Design Tools and Processes	Comments
1a	Statement coverage	++	++	+	+	Simulink Coverage – Model coverage analysis Simulink Coverage – Code coverage analysis	During model testing, Simulink Coverage can collect execution coverage at the model level. During SIL and PIL execution, Simulink Coverage can measure the statement coverage of the generated code.
1b	Branch coverage	+	++	++	++	Simulink Coverage – Model coverage analysis Simulink Coverage – Code coverage analysis Simulink Desi… – Test case ge…	During model testing, Simulink Coverage can collect decision coverage (also known as branch coverage) at the model level. During SIL and PIL execution, Simulink Coverage can measure the decision coverage of Simulink Design Veri… can generate test cases … the model level.

ISO 26262 Recommended Activities

IEC Certification Kit Recommendations

ISO 26262-6: 2018 表 9 に関するツールのマッピングの例。

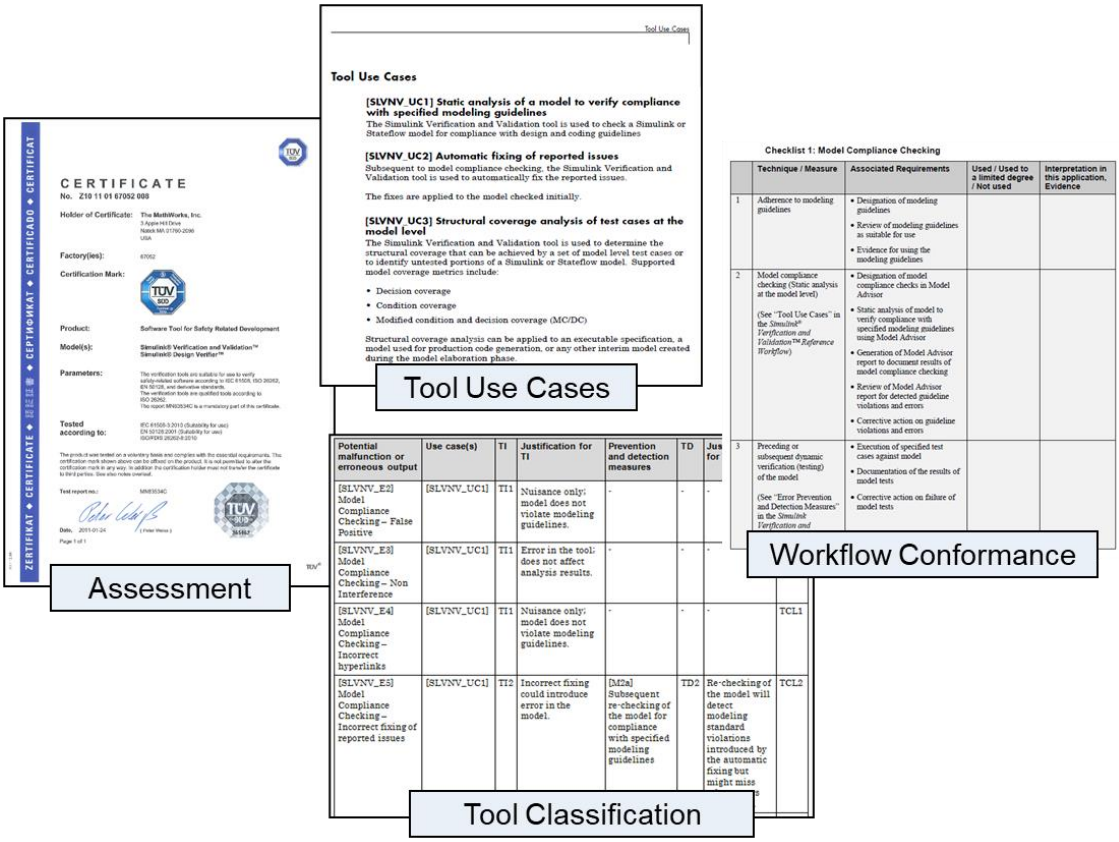


IEC Certification Kit リファレンス ワークフロー。

5. サプライヤー提供のツール検定キットを活用していない

ISO 26262-8: 2018 では、ISO 26262 に準拠したアプリケーションの開発時に使用されるサポートプロセスについて説明しています。また、このセクションでは、自動車業界では馴染みのない主要な概念である、ツールの適格性確認についても取り上げています。ツールの適格性確認プロセスがないと、自動化されたツールが最終製品にエラーをもたらす可能性があります。ASIL 評価のアプリケーションの開発にツールが使用される場合、ツールはそのツール信頼度 (TCL) に基づいて分類される必要があり、その後、安全に使用できることを証明するために必要な手順に従う必要があります。

この作業は面倒ですが、多くのツールベンダーがツール認証キットを提供しているため、負荷を軽減することができます。これは MathWorks に関しても同様です。



IEC Certification Kit で事前に適格性を確認した作業成果物の例。

上のスクリーンショットは、MathWorks IEC Certification Kit ドキュメンテーションの詳細です。このキットには、使用例、潜在的な不具合、エラーの検出と防止方法に基づいて各ツールを分類するために行われた解析が用意されています。TCL 1 以外に分類されたツールについては、ツール開発プロセスの独立した評価や、テストケース経由のソフトウェアツールの検証などの詳細な作業成果物も用意されています。

MathWorks の ISO ギャップ解析により、多くのお客様がこのような認証キットの利用を見落としており、結果的に多くの作業が繰り返されていることがわかりました。作業を見直してみると、ツールの適格性確認作業の多くで特定の作業成果物を見逃していたか、誤った解析、誤った正当化、またはその両方が行われていたことがわかりました。これらの要因はすべて、評価中に必要のないやり直し作業を行う原因になる可能性があります。

まとめ

最近の ISO 26262 プロジェクトで発見されたよくある落とし穴についてご紹介しました。この情報が、ISO 26262 への準拠を必要とする自動車エンジニアリング チームのお役に立つことを願っています。これらの落とし穴を回避する鍵は、計画、プロセスの取得、ソフトウェア アーキテクチャ、レポート/記録、自動化、およびツールの適格性確認です。

問い合わせ先

皆様のご意見、ご感想をお待ちしております。

袁 帥, syuan@mathworks.com

関連情報

ISO 26262 プロセス導入アドバイザリサービス - 技術コンサルティング サービスの概要

MATLAB および Simulink における ISO 26262 のサポート - リソース