意図する使用の定義

デイブは、自分のソフトウェアの意図する使用を定義するために情報を収集する。彼は、オペレータの安全を守り、一定した溶込みを得るには、このプロセスにおけるビジョンの精確さ、動作、電力及びスピードが重要だと知っている。

デイブは、まずソフトウェアの目的と意図から開始して、彼の意図する使用を定義する。

ソフトウェアの目的は、作動しているレーザーとの直接接触から機械のオペレータを保護しながらケースカバーを溶接することである。これには、上述のプロセスの記述のステップ5〜8が含まれる。

リスク分析

デイブは、このプロセスからヒューマンエラーをなくしたいと考えており、レーザー、サーボ及びビジョンの制御がこのプロセスの重要な要素であると知っている。ソフトウェアは、まず、ドアが閉まっているかを確認する。安全上の理由から、ソフトウェアは、ドアが閉まっていることを感知しないと開始しない。ソフトウェアは、レーザーのスイッチが切れていることを確認し、ドアを開けた後に終了する。緊急停止又は予期せずドアが開いた場合、レーザーへの電力が切断する。彼は、このプロセスで得た情報及び溶接が一部であるプロセスの設計の一部として実施された設計リスクマネジメント活動からの情報を使用する。彼は、FMEAを参照し、重要な部品の変数、ハーメティックシール及びユーザーインターフェースの3領域に注目する。デイブは、このプロセスに関して多数のハザードを特定した。まず、オペレータはレーザーに接触して熱傷を負う可能性があった。この製品に関しては、このプロセスで不適切な溶接により不良製品が生産され、漏れが生じ、また最終ユーザーが負傷する可能性があった。デイブはこのプロセスのリスクは高いと判断した。

検証プランニング

デイブはツールボックスの定義ツールを見て、このプロジェクトにはソフトウェア要求事項の定義及び保守文書の作成が必要と判断する。彼のソフトウェア要求事項には、ツール用の設定変数、レーザー照射時間及び電力調整が含まれなければならない。また、ソフトウェア とハードウェアのインターフェースも定義しなければならない。特に、デイブはビジョンシステム、レーザー照射時間及び電力の範囲、動作制御の精確さの要求事項及び、レーザーが作動した場合のハードウェアのドアロックとのインターフェースを含めたドアセンサーの安全装置を入れる。

デイブはさらに、オートメーション技術者、製造技術者及び品質技術者が参加する正式なソフトウェア要求事項審査を実施する必要があると判断する。

このシステムのソフトウェアは市販のパッケージであるが、 デイブはカスタム修正が必要であると知っている。彼は、工場の MES システムにインターフェースを追加する必要がある。

リスクコントロール手段

次に、デイブはリスクに注目する。 彼は、下流の漏れ検査及び定期的な破壊試験による溶込みの検査で十分であると確信していたため、溶接深さ及びその他の重要変数の重大さは低いと考えた。同様に、漏れ検査でハーメチックシールが受容可能であるか確認できる。これで、ユーザーインターフェース領域のリスク、特にドア開放時にソフトウェアがレーザー照射を開始するリスクが残る。デイブはソフトウェアがドアの閉鎖を確認することを知っているが、ソフトウェアが意図した操作を実施しない場合のリスクが大きいため、ドア解放時のレーザー作動を防止するために重複してハードウェアのインターロックを追加する。

検証業務

次に、デイブは検証業務に移る。彼が選択したツールの納入業者は広範なプログラミングツールを支給したため、ソフトウェア要求事項の仕様及び先に作成した審査で設計には十分であり、ツールボックスから設計ツール、開発ツールおよび設定ツールを追加する必要はない。

他にデイブがツールボックスの試験セクションから選択した業務は、試験計画書のソフトウェア環境の詳細と予想試験結果が入ったテスト計画などであった。試験計画書は、デイブ以外にオートメーション技術者、製造技術者及び品質技術者が審査し、承認する必要がある。予想結果と比較した実際の試験結果、合否、試験の特定、並びに問題解決の文書化及び故障があった場合の回帰テストを含む試験結果について、デイブは自分以外にオートメーション技術者、製造技術者、品質技術者及びプロジェクト依頼者の承認を求めた。

導入

溶接機の導入について、デイブはツールボックスの導入ツールを見直し、製造オペレータの手順が必要であり、オートメーション技術者、製造技術者及び品質技術者が審査しなければならないと判断した。オペレータが溶接機の操作方法を確実に理解するために、デイブは、試験を含むオペレータの教育訓練及び認証手順を作成した。彼は、

MES システムでは、認証のないオペレータが溶接プログラムをシステムから取り出すことができないことを知っているため、オペレータ損傷のリスクを軽減できると安心している。

保守

デイブは、設定検査ツールがあることを知っているため、この検証において特に保守計画を作成しない。

例 3：自動溶接プロセス制御システム

この例は、本 TIR の図 2 に示されたプロセスを表す。

		プロセス	反復的リスク分析	検証プランニング及びレポート作成	ソフトウェアシステム

プロセス要求事項の定義（TIRセクション 4.3.1.1 参照）

デバイスコーポレーションはクラス III の医療機器製造業者である。デバイスコーポレーションは、自動溶接プロセス制御システムを実施することにした。医療機器のケースが適切に溶接されるよう、このプロセスは、パラメトリックリリース決定プロセスに基づいた製品隔離方法を支給する。デバイスコーポレーションは、このプロセスの情報を利用して自社の機器履歴簿を支援することも決定した。

デバイスコーポレーションは、自動溶接プロセス制御システムを検証するために新しいプロジェクトマネージャーを選任した。このプロジェクトマネージャーは、21 CFR Part 820 の知識にもとづいて、このシステムは 820.70（i）生産及びプロセス制御に従わなければならないことを認識する。したがって、プロジェクトマネージャーは、導入予定の溶接プロセス制御システムに検証が必要であると認識する。

この溶接システムの検証に関する要求事項及びリスクの理解を深めるために、プロジェクトマネージャーは次の通りプロセスを定義する。

1. オペレータは、ロットの最初の部分のロット番号をシステムに入力する。
2. オペレータは、機械の固定具にサブ部品を挿入する。
3. オペレータは、サイクル開始ボタンを押す。固定具は油圧で対応する位置に移動する。
4. 固定されたサブ部品の一定速度の回転とともに溶接サイクルが開始する。
5. 赤外線温度計が、溶接プロセス中の材料の温度をモニタリングする。温度は、各溶接済み部品のロット番号及び部品連続番号とともにファイルに記録する。
6. サイクル終了時に機械が固定具を開く。
7. オペレータは、溶接した部品を取り出し、連続番号に従い、部品をロットトレーの対応する位置に置く。
8. オペレータは、ロットトレーが一杯になるまでステップ 2～7 を繰り返す。
9. オペレータは、ロット終了ボタンを押す。
10. 機械のオペレータインターフェースは、溶接温度がプロセスの範囲を超えた部品連続番号を表示する。
11. オペレータは、ロットトレーから対応する部品番号を破棄する。
12. オペレータは、拒否する部品リストを印刷し、ロットトレー及び報告書を次のステーションに送付する。
13. オペレータは、ステップ 1 を繰り返して新しいロットを開始する。

プロジェクトマネージャーも、主なオートメーション機能を次の通り認識する。

14. ロット番号の保管
15. 連続部品番号毎の溶接温度の保管
16. 溶接中にプロセスの温度範囲を超えた部品の連続番号の表示
17. ロット拒否報告書の印刷

開発	定義	プロセス	**反復的リスク分析**	検証プランニング及びレポート作成	ソフトウェアシステム

プロセス障害リスク分析（TIRセクション 4.3.1.2 参照）

プロジェクトマネージャーは、次に現行のプロセスで起こりうる問題について考える。彼は、このプロセスが故障すれば、適切に溶接されていない部品により患者が未滅菌の医療機器に暴露される可能性があると認識する。溶接プロセス制御システムの誤り又はオペレータの誤りで、不良製品の偶発的なリリースが生じうる。

プロジェクトマネージャーは、次に、このリスクを軽減するためにどのようなリスクコントロール手段が実施されているか考える。彼は、このプロセスグループが、次のプロセスステップで溶接オペレータが正確に部品を不合格にしたかを検証する手順を設定していることを知る。さらに、彼はこの溶接システムが、市販の既製品であることを知る。

開発	定義	プロセス	反復的リスク分析	検証プランニング及びレポート作成	ソフトウェアシステム

ソフトウェア 目的と意図（TIRセクション 4.3.1.4 参照）

ここで、プロジェクトマネージャーは、自動化するプロセスについて基本的に理解したため、溶接プロセス制御システムの目的と意図を作成する準備が整った。

彼は次の通り記載する。

> *この溶接プロセス制御アプリケーションは、溶接したケースの合否について閉鎖ループの品質決定を行う。この決定に基づき、溶接オペレータは、手動で不適合製品を拒否する。*

最後に、プロジェクトマネージャーは、導入予定のシステムが遵守すべき FDA の規制について検討する。そこで、彼は、この重要な事実を反映するために、次の文章を追加する。

> *この溶接プロセス制御アプリケーションは、DHR の一部である記録を保管する。したがって、820.80（d）に従い、このシステムには、最終受入活動の支援に必要な電子記録が含まれる。この従前規則により記録が必要なため、21 CFR 11 の電子記録の要求事項が適用される。*

プロジェクトマネージャーは目的と意図を審査し、このプロセスにおけるソフトウェア の境界を適切に記録する。この審査に基づき、彼は、以下の通り文章の修正を決定する。

> *この溶接プロセス制御アプリケーションは、溶接したケースの合否について閉鎖ループの品質保証決定を行う。この決定に基づき、溶接オペレータは、手動でパラメータが不適合な製品を拒否する。溶接ステーションは、医療機器全体の密封の完全性を制御する唯一のポイントである。*
>
> *この溶接プロセス制御アプリケーションは、DHR の一部である記録を保管する。したがって、このシステムには、820.80（d）に従い最終受入活動の支援に必要な電子記録が含まれる。この従前規則により記録が必要なため、21 CFR 11 の電子記録の要求事項が適用される。*

プロジェクトマネージャーは、次に、溶接システムとインターフェースする他のシステムがあれば、それは何であるか考える。彼は、このソフトウェアは、IR 温度機器、オペレータインターフェース、プリンター及び機械の PLC I/O に接続した PC で作動する単一のアプリケーションであると判断する。

		プロセス	反復的リスク分析	検証プランニング及びレポート作成	ソフトウェアシステム
開発	定義	検証プランニング（TIRセクション 4.3.1.3 参照） ここで、プロジェクトマネージャーは自動化するプロセスを理解し、新しいシステムの意図する使用を決定したため、高レベルの検証プランニングを計画する準備が整った。 既に、プロジェクトマネージャーは、溶接プロセスは検証不可能なプロセスとして実施するため、このプロセスの残留リスクは高いと判断した。したがってプロジェクトマネージャーは、検証の取り組みの広範な審査が必要であると判断する。彼は、プロセスエンジニアリング、品質エンジニアリング及びオペレーションズプロセストレーナーが主要な承認の役割を担うべきだと決定する。次に、彼は、最終製品受入マネージャーも要求事項を承認すべきだと考える。 プロジェクトマネージャーは、彼の品質システムは、他の検証の成果物又はプロジェクトの成果物が承認される前に高リスクシステムとして承認される検証プランニングを必要とするため、検証プランニングの作成開始を決定する。			

		プロセス	反復的リスク分析	検証プランニング及びレポート作成	ソフトウェアシステム
開発	定義	ソフトウェア使用の要求事項及びソフトウェア要求事項（TIR セクション 4.3.1.4 参照） プロジェクトマネージャーは、この検証の取り組みに、高レベルの詳細又は形式を取り入れる必要があると考える。彼は、詳細なプロセス及びソフトウェア要求事項を定義することが重要であると知っている。プロジェクトマネージャーは、ここで、ソフトウェア要求事項を作成する。彼は、ソフトウェアには温度検証の重複及び拒否決定プロセスを入れるべきであると判断する。また、このプロセスには、ラインクリアランス前に拒否報告書を再度印刷するシステムが必要である。 このシステムはパラメトリック値を支援するため、彼は、安全要求事項も、システムアクセスレベルによりどのデータの数値を変更できるかについての詳細なリストとともに入れる。			

開発	インプリメンテーション／試験／導入	プロセス	反復的リスク分析	検証プランニング及びレポート作成	ソフトウェアシステム

ソフトウェア障害リスク分析（TIRセクション 4.3.2.1 参照）

プロジェクトマネージャーは、この溶接システムに対する完全な信頼を確立するためにどんなアプローチを使用すべきかを決定する時点にきている。

プロジェクトマネージャーは、この溶接の設計は業界で一般的に使用される COTS システムを必要とすると述べる。彼は、この製品について過去に生じた問題はすべて製造業者が迅速に特定し、公表したことを知る。

プロジェクトマネージャーは、既に、自動化すべき溶接プロセスが高リスクであると判断したが、まだ、ソフトウェア障害のリスクを正式に分析したい。プロジェクトマネージャーは、彼の洞察を確認するために、彼の会社のリスクモデルに関する質問事項を検討する。

1. ソフトウェアが誤作動した場合、製品の安全に潜在的リスクはあるか。*はい*
   a. どのようなものか。*初期設定温度の限界値に基づき、システムが不良な部品を合格とする。停電後、限界値は初期設定に戻る。*
   b. このリスクの制御には何をすべきか。*各ロットの運転前後にオペレータが限界値を検証する必要がある。*
2. ユーザーが間違えば、製品品質（安全リスク以外）に潜在的リスクはあるか。*ある*
   a. どのようなものか。*手動モードでは、両方の部品センサーが 3 秒間作動すると溶接レーザーが発射しうる。*
   b. このリスクの制御には何をすべきか。*自動モードにおいてのみ発射するという初期設定を変更する。*
3. 記録紛失は規制遵守を示す能力に対する潜在的リスクはあるか。*はい*
   a. どのようなものか。 *デバイスコーポレーションは、使用した溶接変数及びロットの合否に使用した実測データを知らなければならない。*
   b. このリスクの制御には何をすべきか。*プロジェクトマネージャーは、21 CFR Part 11 ソフトウェア要求事項を追加し、ロット報告書の終わりに特定の溶接の限界値を印刷すべきという要求事項も追加する。*

開発	インプリメンテーション／試験／導入	プロセス	反復的リスク分析	検証プランニング及びレポート作成	ソフトウェアシステム

検証プランニング（TIR セクション 4.3.2.2 参照）

プロジェクトマネージャーは、検証を終了するための十分な情報を収集し、ソフトウェア要求事項について理解し、インプリメンテーションアプローチを決定し、ソフトウェアリスクを分析した。この時点で、このシステムについて知っていることすべてを考慮して、この溶接システムがその意図する使用に合うという信頼を確立するにはどのような検証活動が必要かを自問する。

- プロジェクトマネージャーは、第三者がどのようにこのシステムを開発したかを考え、開発者が報告のカスタマイゼーションの要求事項を正確に実施したか懸念する。このシステムは様々なデータ領域に依存するため、開発者の業務の正確性を確認するためにコード審査に検証ステップ活動を追加する。

開発	インプリメンテーション／試験／導入	プロセス	反復的リスク分析	検証プランニング及びレポート作成	ソフトウェアシステム

ソフトウェアのインプリメンテーション（設計/開発/構築/試験セクション 4.3.2.3 参照）

ソフトウェアを内部で開発するのではなく購入するという決定は、コマーシャル・オフ・ザ・シェルフ（COTS）の能力にもとづいて行う。しかし、プロジェクトマネージャーは、意図する使用は高リスクに分類されているため、溶接制御ソフトウェアが妥当性の確認されたソフトウェアの開発ライフサイクルのもとで開発された事をデバイスコーポレーション の品質部門に照明する必要があることを知っている。

彼は、COTS 供給者とこの問題について協議した後、供給者の SDLC プロセスが最近独立監査会社の監査を受けたことを知った。プロジェクトマネージャーは、そこで、COTS 供給者に連絡し、SDLC 査察報告書の写しを購入できた。その結果、品質部門は、COTS 供給者が有効なライフサイクルモデルのもとでこのソフトウェアを開発したと確信した。

開発	インプリメンテーション／試験／導入	プロセス	反復的リスク分析	検証プランニング及びレポート作成	ソフトウェアシステム

検証レポート（TIRセクション 4.3.2.4 参照）

プロジェクトマネージャーは、検証レポートを記入し、承認を得る。

開発	インプリメンテーション／試験／導入	プロセス	反復的リスク分析	検証プランニング及びレポート作成	ソフトウェアシステム
		ソフトウェアのリリース（TIRセクション 4.3.2.5 参照） プロジェクトマネージャーは、彼の正式な構成管理システムのソフトウェアが彼の検証レポートで引用されたソフトウェアと一致することを検証する。			

開発		プロセス	反復的リスク分析	検証プランニング及びレポート作成	ソフトウェアシステム
		変更の分析 プロジェクトマネージャーは、彼の検証プランニングのもとで、検証後の溶接システム変更を管理する正式な変更制御プロセスが彼の会社にあることを検証した。			

開発		プロセス	反復的リスク分析	検証プランニング及びレポート作成	ソフトウェアシステム
		保守のプランニング（TIRセクション 4.4.1 参照） プロジェクトマネージャーは、システムが引き続きその意図する使用を満たすことを確認するためにどの活動が適切であるかを前もって考える。このシステムは高リスクであるため、彼は、3ヵ月に1回校正を行い、温度の実測値及びロット報告書に印刷された温度が正確かつ厳密であるという承認をしなければならないと決定する。プロジェクトマネージャーは検証プランニングにこれを記録する項を加え、このシステムが生産段階に入ったら3ヵ月に1回の審査が確実に実施されるように校正/認証手順の開発及び実施を請求する。			

開発		プロセス	反復的リスク分析	検証プランニング及びレポート作成	ソフトウェアシステム
		ソフトウェアの保守 プロジェクトマネージャーは、彼の検証プランニングのもとで、溶接システム及びプロセスがその意図する使用とは異ならないことを保証する定期的審査プロセスが彼の会社にあることを検証した。			

開発		プロセス	反復的リスク分析	検証プランニング及びレポート作成	ソフトウェアシステム
		ソフトウェアの廃用 プロジェクトマネージャーは、彼の検証プランニングのもとで、溶接システムの廃用を管理する正式なソフトウェア廃用プロセスが彼の会社に存在することを検証した。			

ツールボックスの選択:

設計、開発及び設定ツール

- プロセス要求事項の定義
- 正式なソフトウェア要求事項の審査
- 製造/ビジネスプロセス内のリスクコントロール手段の特定
- プロセス開発審査
- トレーサビリティマトリックス（要求事項の仕様に内在）

試験ツール

- テスト計画
- ソフトウェアシステムテスト
- ソフトウェア設定制御

導入ツール

- ユーザー手順の審査
- アプリケーションの内部教育訓練
- 据付時適格性確認
- プロセス検証

例 4：C/C++言語コンパイラ

背景

クラス III の医療機器会社が、組み込みシステムについて自社の OTSS C/C++言語コンパイラの検証をしたい。このコンパイラは DHF（設計履歴ファイル）に入った製品ソフトウェア（ソフトウェアソースコード及び実行可能ソフトウェア）を作成するため、コンパイラは調整されていると判断された。

品質システムプロセスの記述

このケーススタディには、2 つの品質システムプロセスがある。一つ目は、クラス III の医療機器ソフトウェアインプリメンテーション全体の品質システムプロセスである（図 1 参照）。二つ目は、ソフトウェアの設計を実施し、OTSS C/C++言語コンパイラを含むすべてのソフトウェア要求事項を満たすプロセスである。（図 1 の“ソフトウェアのインプリメンテーション”の項参照）。

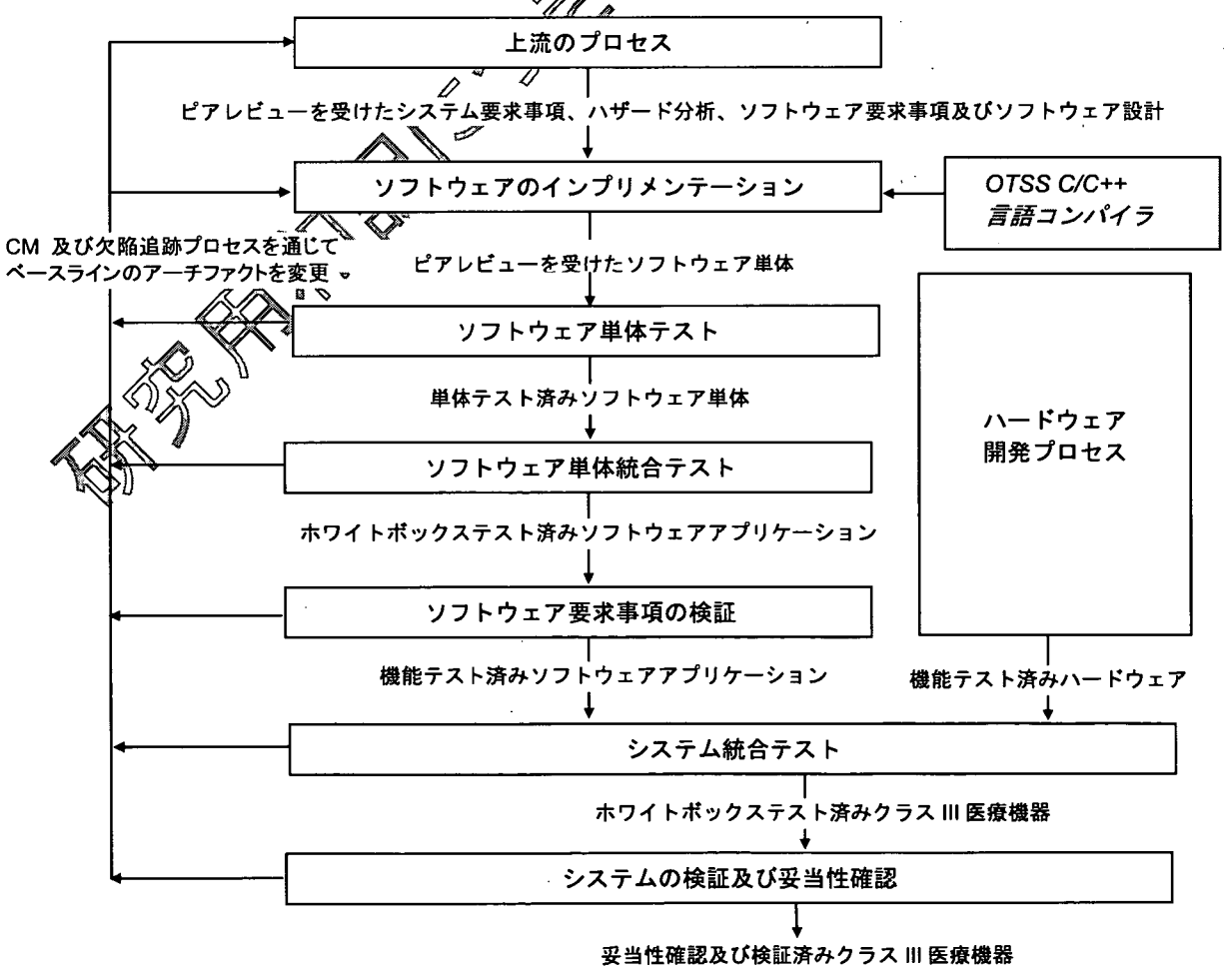

図 C1－クラス III 医療機器ソフトウェアのインプリメンテーション

上流のプロセス

ソフトウェアインプリメンテーションの上流プロセスには、開発する医療機器を特徴付けるシステムレベル文書（要求事項、設計、ハザード分析など）の開発プロセスがある。次に、ソフトウェアに実装したシステムの部分について、ソフトウェア要求事項、ソフトウェア設計及び他のソフトウェア文書又は計画の開発プロセスを通じて特徴を付与する。ソフトウェアの開発と並行して、医療機器ハードウェア開発のためにも追加プロセスが実施される。

ソフトウェアインプリメンテーションプロセス

使用した正式なソフトウェア言語は C/C++ソフトウェア言語である。OTSS C/C++言語コンパイラを使用して高レベルのソフトウェアステートメントを実行可能なマシンコードにコンパイルする。ソフトウェアインプリメンテーションプロセスの出力は、他の技術担当者が完全性及び正確性をピアレビューしたベースラインのソフトウェア単体である。ソフトウェア単体をピアレビューするには、ソフトウェア単体は、最高のコンパイラレベルで誤りなくコンパイルされ、コンパイラの警告はすべてピアレビューで説明しなければならない。

下流の検査プロセス

ソフトウェア単体は、次の通り、複数の試験プロセスで試験又は検証する。

- ソフトウェア単体テスト。個々のソフトウェア単体の論理的正当性及び境界条件を試験する。この試験は、開発システム又は標的システム（医療機器のハードウェア）で実施する場合がある。単純なソフトウェア単体については、コードのピアレビューが単体の論理的な誤り検知に十分であると判断された場合にはこの検査を実施しなくてもよい。
- ソフトウェア単体統合テスト。ソフトウェア単体を統合及び試験し、ソフトウェアの設計が正しく実装され、設計がテストされたことに関する境界条件が試験されたことを保証する。この試験は標的システムで実施する。
- ソフトウェア要求事項の検証。完全な一組のソフトウェア要求事項に照らし合わせて、完全にソフトウェアアプリケーションを検証する。検証は標的システムで実施する。
- システム統合テスト。医療機器のソフトウェア及びハードウェアをテストし、システムの設計が正しく実装され、システムの設計が試験されたことに関する境界条件が試験されたことを保証する。
- システムの検証及び妥当性確認。医療機器は、システム要求事項レベルで検証し、また、その意図する使用の妥当性を確認する。

プロセス障害リスク分析

このプロジェクトは、自社のプロセスリスクアセスメント手順に従った。クラス III 医療機器ソフトウェアインプリメンテーションの全体の品質システムプロセス（図 1 のすべてのプロセスを含む）は、クラス III 医療機器内で機能するソフトウェアを作成するため、本質的に高リスクである。

ソフトウェアインプリメンテーションプロセスの一部である OTSS C/C++言語コンパイラは、次に基づき低リスクと評価された。

- 患者、オペレータ又は第三者に対する直接重大な損傷又は死亡原因にはならない。
- ツール（例えば、ソフトウェア単体テスト、ソフトウェア単体統合テスト、ソフトウェア要求事項の検証、システム統合テスト、システムの検証及び妥当性確認）の出力（ソフトウェアソースコード及び実行可能なソフトウェア）の下流での検証。

意図する使用の定義

上述のソフトウェアインプリメンテーションプロセス内の OTSS C/C++言語コンパイラの目的と意図は、組み込みシステムソースコードを作成し、コンピレーションプロセスを実行し、クラス III 医療機器の実行可能なソフトウェアを作成することである。

ソフトウェア使用の要求事項

1. ツールは、C 及び C++コードをクロスコンパイルして、選択した納入業者のオペレーティングシステムを用いたRISCプロセッサで作動しなければならない。
2. コンパイラにはソースコードデバッガがなければならない。
3. コンパイラは、ANSI C 及び C++対応でなければならない。
4. コンパイラは、様々な承認された業界標準の統合開発環境（IDE）を統合しなければならない。
5. 納入業者は、検索可能な既知のバグリストを公表しなければならない。このリストは、必要に応じて相談の参照として使用しなければならない。
6. 納入業者は、規制産業内に大規模なユーザー基盤が必要である。

ソフトウェアの障害リスク分析

このOTSS C/C++言語コンパイラのリスク分析から、誤りがあった場合に次が発生しうることがわかった。

- リスク_1：OTSS納入業者は、技術の明らかな特徴及び機能に加え、適切なビジネスプロセス、開発方法及びサポートを提供できない。
- 軽減_1：納入業者選択プロセス。（下の項参照）
- リスク_2：間違った実行可能な文章を作成する。
- 軽減_2：検証プランニング（下の項参照）
- リスク_3：ユーザーによる誤使用。最も厳格なレベルのエラーチェックを実施していない。
- 軽減_3：教育訓練及び手順/作業指示書

納入業者選択プロセス

このプロジェクトは自社の納入業者選択承認に関する品質システム手順に従い、この情報はプロジェクトのDHFに保管する。この手順には、納入業者のSDLCに関する方針、手順、業務及び活動を審査するオンサイトでの評価が含まれた。OTSS納入業者のC/C++言語コンパイラの能力は、上記のソフトウェア使用の要求事項を満たすことが検証された。

検証プランニング

このOTSS C/C++言語コンパイラでは、下流の検証アプローチを選択した。納入業者の選択プロセスで、この納入業者がすべての文書化されたソフトウェア使用の要求事項を満たすと判断された。このコンパイラは、納入業者において重要なランタイムがあり、このプロジェクトで実施するデバッグ及び試験でも重要なランタイムがある。コンパイラの出力は、下流のプロセスにおける次の動的テストを受ける。

- ソフトウェア単体テスト
- ソフトウェア単体統合テスト
- ソフトウェア要求事項検証テスト
- システム統合テスト
- システムの検証及び妥当性確認

検証レポート

検証レポートの内容

- OTSS の記述
- ソフトウェア使用の要求事項
  - ハードウェア要求事項
  - ソフトウェア要求事項
  - パッチ
- リスクアセスメント/ハザード分析
- 納入業者選択
- インストール活動
- 検証
  - ソフトウェア使用の要求事項テストケース/結果
- 既知のバグリスト
- 設定制御
  - 教育訓練
  - インストール位置
  - 保守
  - 廃用プロセス

ツールボックスの選択

- 定義:
  - 意図する使用
  - 検証プランニング
  - リスクマネジメント計画（リスクアセスメント）
- インプリメンテーション:
  - リスクコントロール手段
  - 納入業者監査
- 導入
  - 据付時適格性確認
  - アプリケーションの内部教育訓練
  - 最終受入試験
- 保守段階
  - 保守計画
  - 既知の問題の分析

例 5： 自動ソフトウェア試験システム

背景

この例では、製造業者はクラス II の医療機器製造業者である。この製造業者が製造した医療機器はソフトウェアで制御する。そのソフトウェアは、アーキテクチャ上、オペレータコンソール及びリアルタイム組み込み制御ソフトウェアという 2 つの重要な要素で構成される。オペレータコンソールは、このシステムへの主なヒューマンインターフェースである。リアルタイム組み込み制御ソフトウェアは、電気機械制御、データ収集、タイミングなどすべてを実施するソフトウェアである。オペレータコンソールソフトウェア（業界の標準オペレーティングシステム及びデータベース搭載 PC に内蔵）及びリアルタイム組み込みソフトウェア（オンボード組み込み CPU カードに内蔵）は、標準 TCP/IP ハードウェア及びプロトコルインターフェースを用いてインターフェースしている。

このプロジェクトのソフトウェアマネージャーは、ソフトウェアの自動試験を導入してソフトウェアの開発・試験プロセスを改善することは価値があると判断した。ソフトウェアマネージャーは、まず、オペレータコンソールソフトウェアの自動ソフトウェア試験のみを実施することにした。自動ソフトウェア試験は、統合テスト及びソフトウェアシステムテストの両方の時点で実施する。

ソフトウェアが調整されているかの判断

自動試験ソフトウェアは、この製造業者のソフトウェア開発手順に必要な試験を実施するために使用し、統合テスト及びシステムテスト時点の回帰テストに必要な証拠を提供するため、開発プロセスの一部を自動化し、21 CFR 820.70 (i) –生産及びプロセス制御-自動プロセスの検証要求事項の対象であったことがわかった。

プロセスの定義

オペレータコンソールの自動ソフトウェア試験導入の要求事項及びリスクの理解を深めるために、ソフトウェアマネージャーは、ソフトウェア開発プロセスにおける自動試験ソフトウェアの使用を次の通り定義する。

医療機器のソフトウェア開発中に、様々な時点でシステムソフトウェアに様々なモジュールの統合を予定している。また、既にシステムに統合されているモジュールは、欠陥修正及び要求事項の修正のために変更される。自動試験システムは、統合システムソフトウェアの回帰テスト及びシステム中の特定のモジュールの最終試験への使用が計画されている。このソフトウェアプロジェクト計画では、週に 2～3 回のモジュ

ール統合又は更新が必要になる。自動試験は、このような各統合ポイントで実施し、新しい機能が正常に作動し、それまでの機能に追加したコード又はあるビルドで変更したコードが悪影響を及ぼしていないことを確認する。自動試験は、検証、そして最終的には顧客への最終リリースの候補であるビルドのソフトウェアシステム試験レベルで実施する。また、自動試験は、予定の手動試験を保管する回帰テストレベルを提供するための修正が必要な最終開発段階で欠陥が発見された場合にも使用される。

リスクの分析

ソフトウェアマネージャーは、ここで分析プロセスに進み、自動試験ソフトウェアについて問題が生じた場合の影響を判断する。

まず、ソフトウェアマネージャーは、自動試験プロセスの故障、自動試験ソフトウェアの故障、又は自動試験ソフトウェアエラーによる誤りが最終的に医療機器の故障になり、患者、オペレータ、第三者、サービス担当者又は環境への潜在的危害が生じるかどうかを評価する。

- ソフトウェアマネージャーの最大の関心は、自動ソフトウェア試験システムが、検査中のオペレータコンソールソフトウェアに対して実際には欠陥があるのに正常に作動しているという誤った指示を与える可能性があるということである。
- 検出されない欠陥がソフトウェアの重要な部分にあった場合、医療機器が誤作動し、危害が生じうる。
- ソフトウェアマネージャーは、この潜在的リスクは、自動試験ソフトウェアの誤った管理又は誤った自動試験ソフトウェア使用、若しくは自動試験ソフトウェア自体の故障により発生しうると理解する。
- ソフトウェアマネージャーは、自動ソフトウェア試験システム使用時期、及び使用目的に境界条件を設定し、ソフトウェア開発・試験チームがこのシステムに過度に依存しないようにすることが重要であると判断する。
- 自動試験ソフトウェアの設定、プログラミング及び操作に関わる者は、その役割について教育訓練を受ける必要がある。
- ソフトウェアマネージャーは、これらの要素を制御することで、関連する潜在的リスクを受容可能なレベルにまで軽減できると感じる。

ソフトウェアの意図する使用の定義

ここで、ソフトウェアマネージャーは自動試験ソフトウェアの使用の可能性を分析し、関連するリスクについて理解したため、自動ソフトウェア試験システムの目的と意図を作成する準備が整った。

- 自動試験システムは、開発プロセスにおける統合テスト時点のソフトウェアのビルドを試験するために使用する。
- 自動試験システムは、ソフトウェアのシステムテスト時点で検証と候補のリリースビルドを試験するために使用する。
- 自動試験システムは、システムの回帰テストを行い、新たに導入したソフトウェア又は変更したソフトウェアによりワークフローに悪影響が出ていないかを確認する。
- 自動試験システムの一般的な役割は、実施される手動試験を保管する回帰テストの実施である。
- 複雑度が低く、予測可能なワークフローには、特定のプロトコルが、相当する手動試験と同等であると検証されている場合、自動試験システムをソフトウェアの正確性の最終決定要因として使用できる。
- 自動試験システムは、ソフトウェアシステム又は医療機器全体の安全装置（リスク軽減）を提供するソフトウェアを実行する。

検証プランニング

ソフトウェアマネージャーは、自動化すべきプロセス、特定の自動試験システムの意図する使用、及び関連する潜在的リスクを明確に理解している。彼は既にこのソフトウェアの使用に関して何らかの制御を設置すべきと判断した。彼は、自動ソフトウェア試験システムが適当な制御とともに使用された場合、使用によるリスクは受容可能レベルになると判断した。

この場合、彼は、“ 適切に使用した”場合、この自動ソフトウェアシステムが原因で医療機器が故障するリスクはほとんどない、又は全くないと判断した。彼は、“ 適切に使用した”場合とは、ソフトウェア開発・試験チームがソフトウェアの正確さの判定に際しこの自動試験システムに過度に依存しないことと定義した。彼が低リスクと判断したため、ソフトウェアマネージャーは、このシステムの検証の要求事項は、ソフトウェア試験システムに関しては最低のレベル及び厳格さとなると判断した。

検証の文書化 –" 検証レポートアプローチ"

ソフトウェアマネージャーが採用を決定したアプローチは、システムに必要なレベルの信頼を獲得するためのすべての活動の要約を含む自動ソフトウェア試験システムのソフトウェア検証レポートの作成である。

批判的思考

ソフトウェアマネージャーは、ここで、システムが適切に使用され、医療機器の深刻な欠陥の原因とならない必要な信頼レベルの最良の獲得方法を決定するプロセスに移る。

彼は、システムに必要レベルの信頼を獲得するために最も重要な要素のひとつは次の通りであると判断する。

適当な意図する使用の厳格な遵守

- ソフトウェアの開発及び試験に関わる要員すべてがシステムの境界条件及び適当な意図する使用を確実に理解するようにする。
- 文書化：検証レポートに、特定の意図する使用及び特定のプロジェクトのソフトウェア開発計画を通したその連絡方法を記述する項を加える。

適当な注意

- その企業の試験システムが同レベルの重要度又はより重要度の高いアプリケーションに使用されている信頼できる納入業者から、業界標準の自動ソフトウェア試験システムを購入する。
- システムの意図する使用を納入業者とともに審査し、その意図する使用が適切かを判断する。
- 市販前にその納入業者のソフトウェアの検証の方法に関する情報を入手する。納入業者の QA 組織から、市販のソフトウェアが納入業者による検証を受けていることを確認する文書を入手する。これにより、自動ソフトウェア試験システムが納入業者により適切に試験されたという信頼を獲得し、その後にソフトウェアマネージャー及び彼のチームが実施する追加的活動の基盤ができる。
- 納入業者との関係を構築し、ソフトウェアマネージャー及び彼のチームが使用する試験ソフトウェアの版についての既知の問題及び欠陥に気づくようにする。
- 納入業者の今後のソフトウェアの更新計画を理解し、新版のソフトウェアへの移行計画及び再検証活動を予測する。

- 文書化：検証レポートに、納入業者の自動ソフトウェア試験システムの検証情報、納入業者の欠陥（バグ）リストへのアクセス方法及び新版のソフトウェアへの移行計画など、納入業者の適当な注意についての活動の結果に関する項を加える。

インストールテスト

- コンピュータの計算環境が納入業者の仕様に適合することを確認する。
- ソフトウェアが正確にインストールされたことを確認するために、最初の高レベルプロトコルを確立する。
- 文書化：検証レポートに、インストール確認活動結果に関する項を加える。

リスクマネジメント

- ソフトウェアマネージャーがソフトウェアの目的と意図で定義した通りにのみシステムが使用されるようにする。
- 自動試験システムを使用するプロジェクトのソフトウェア開発計画に特定の自動試験システムの受入可能境界条件を入れる。
- 手動試験で自動ソフトウェア試験システムが取り扱わない領域を取り扱うよう、試験システムの正確な適用領域を特定するために分析を実施する。
- 文書化：検証レポートに、初回リスク分析で特定されたリスクに関する項を加え、これらのリスクがどのように軽減されるかを示す。

ソフトウェア使用の要求事項

- ソフトウェア開発チーム及びソフトウェア試験チームと協力するソフトウェアマネージャーは、使用しようとする自動試験システムの機能性のリストを作成する。
- このリストを、使用する機能を表す“ ソフトウェア使用の要求事項” リストと呼ぶ。
- 文書化：検証レポートに、“ ソフトウェア使用の要求事項” リスト及び各ソフトウェア使用の要求事項の記述の項を加える。

自動試験システムの検証

- “ ソフトウェア使用の要求事項” に基づき、ソフトウェアマネージャーは、初回自動試験スクリプト/プロトコルを 3 つ選び、同一のプロトコルによる手動試験の“ 並列” 試験を実施し、必要レベルの信頼を獲得できると判断した。
- 3 つの初回試験スクリプト/プロトコルは、チームが使用するすべての機能性を実行する。