



# 中华人民共和国国家标准化指导性技术文件

GB/Z XXXXX—20XX/ISO/TR 80002-2:2017

---

## 医疗器械软件 用于医疗器械质量体系软件的确认

Medical device software—Validation of software for medical device quality system

(ISO/TR 80002-2:2017, Medical device software—Part 2: Validation of software for medical device quality system, IDT)

(征求意见稿)

(本草案完成时间: 2021-8-2)

在提交反馈意见时, 请将您知道的相关专利连同支持性文件一并附上。

XXXX - XX - XX 发布

XXXX - XX - XX 实施

---

国家市场监督管理总局 发布  
国家标准化管理委员会

## 目 次

前言 .....	III
引言 .....	IV
1 范围 .....	1
2 规范性引用文件 .....	1
3 术语和定义 .....	1
4 软件确认探讨 .....	1
4.1 定义 .....	1
4.2 信任建立活动：工具箱内的工具 .....	1
4.3 批判性思维 .....	1
5 软件确认与批判性思维 .....	2
5.1 概述 .....	2
5.2 确定软件是否在范围内 .....	5
5.2.1 将软件过程和使用的层级定义形成文件 .....	5
5.2.2 监管使用评估 .....	5
5.2.3 与医疗器械监管要求无关的过程和软件 .....	5
5.3 开发阶段 .....	5
5.3.1 确认计划 .....	5
5.3.2 定义 .....	6
5.3.2.1 定义阶段要求 .....	6
5.3.2.2 过程要求 .....	6
5.3.2.3 过程失效风险分析 .....	6
5.3.2.4 制定确认计划 .....	7
5.3.2.5 软件预期用途 .....	7
5.3.3 实现、测试和部署 .....	8
5.3.3.1 所需活动 .....	8
5.3.3.2 软件失效风险分析 .....	9
5.3.3.3 确认计划 .....	9
5.3.3.4 软件实现（设计、开发、构建和测试） .....	9
5.3.3.5 确认报告 .....	10
5.3.3.6 软件发布 .....	10
5.4 维护阶段 .....	10
5.4.1 进入维护阶段 .....	10
5.4.2 维护策划 .....	11
5.4.3 维护阶段内的维护类型 .....	11
5.4.4 过程更改：对风险控制措施的更改 .....	11
5.4.5 紧急更改 .....	11
5.4.6 维护预期用途 .....	12
5.5 退役阶段 .....	12

6 文档 .....	12
7 先决条件过程 .....	13
附录 A (资料性) 工具箱 .....	14
附录 B (资料性) 风险管理和基于风险的方法 .....	18
附录 C (资料性) 示例 .....	21
参考文献 .....	64

征求意见稿

## 前 言

本文件按照GB/T 1.1—2020《标准化工作导则 第1部分：标准化文件的结构和起草规则》的规定起草。

本文件等同采用ISO/TR80002-2:2017《医疗器械软件 第2部分：用于医疗器械质量体系软件的确认》。

本文件做了下列编辑性修改：

——为与现有标准协调，将标准名称修改为《医疗器械软件 用于医疗器械体系软件的确认》。

请注意本文件的某些内容可能涉及专利。本文件的发布机构不承担识别专利的责任。

本文件由国家药品监督管理局提出。

本文件由全国医疗器械质量管理和通用要求标准化技术委员会（SAC/TC 221）归口。

本文件起草单位：。

本文件主要起草人：。

征求意见稿

## 引 言

本文件旨在帮助读者应用批判性思维，使用基于风险的方法，确定医疗器械质量体系中使用过程软件确认的适当活动。

其中包括GB/T XXXXX-20XX/ISO 13485:2016中4.1.6、7.5.6和7.6要求的用于质量管理体系的软件、用于生产和服务提供过程的软件以及用于监视和测量的软件。

本文件汇集了从事此类软件确认和负责建立可审核文档的医疗器械行业人员的经验。本文件开发过程中，始终考虑在进行医疗器械质量体系所用软件确认过程时，我们都会遇到的一些问题和疑问，如：必须完成什么工作？多少工作是足够的？如何进行风险分析？深入讨论后得出的结论是：在每种情况下，都会确定一系列活动（即工具箱中的工具），以便对软件按照预期用途运行的能力提供一定的信任。由于软件的复杂性、所涉及伤害的风险以及外购软件的固有属性（例如质量、稳定性）等因素，活动清单各不相同。

本文件旨在帮助包括制造商、审核员和监管机构在内的利益相关方理解并应用GB/T XXXXX-20XX/ISO 13485:2016中4.1.6、7.5.6和7.6所包含的软件确认的要求。

征求意见稿

# 医疗器械软件 用于医疗器械质量体系软件的确认

## 1 范围

本文件适用于医疗器械设计、测试、组件接收、制造、贴标签、包装、流通以及投诉处理中所使用的任何软件，或GB/T XXXXX/ISO 13485中所描述的某种医疗器械质量体系的任何其他方面的自动化软件。

本文件适用于：

- 用于质量管理体系的软件；
- 用于生产和服务提供的软件；
- 用于要求监视和测量的软件。

不适用于：

- 用作医疗器械组件、部件或附件的软件，或
- 医疗器械独立软件。

## 2 规范性引用文件

本文件没有规范性引用文件。

## 3 术语和定义

GB/T 19000和GB/T XXXXX/ISO 13485界定的术语和定义适用于本文件。

## 4 软件确认探讨

### 4.1 定义

“软件确认”这一术语既可以狭义的解释为测试，也可以宽泛的解释为包括测试在内的广泛活动。本文件使用的术语——“软件确认”——表示建立信任的所有活动，使人相信软件适合其预期用途、值得信赖并可靠。选定的任何活动，均宜确保软件满足其需求和预期目的。

### 4.2 信任建立活动：工具箱内的工具

工具箱内的工具（参见表A.1至表A.5）包括软件生存周期内完成的可降低风险并建立信任的各项活动。

### 4.3 批判性思维

本文件提倡采用批判性思维，确定对特定软件进行确认宜实施的活动。批判性思维是一个分析、评价软件各个方面及其使用环境的过程，以确定在确认过程中所应用的最有意义的一组建立信任的活动。批判性思维避免在未对解决方案进行彻底评价以确定其是否确实可产生预期结果的情况下，采用一刀切确认解决方案的方法。批判性思维承认不同软件确认解决方案可能存在很大差异，并允许在类似情况下同一软件采用不同的确认解决方案。批判性思维挑战提出的确认解决方案，以确保其满足质量管理体系要求的意图，并同时考虑所有关键利益相关方及其需求。当发生软件特性更改、软件预期用途更改或获得新信息时，批判性思维也用于确认解决方案的再评价。

由批判性思维作出的确认解决方案可为制造商建立符合性以确保软件使用安全，生成评审人员认为是充分且适宜的形成文件的证据，使实施确认工作的个人感觉其努力是增值的并代表着达到预期结果的最有效方式。

附录C提供的示例研究，展示了如何将批判性思维应用于不同情况下的医疗器械质量体系所用的软件确认，包括不同复杂程度、不同固有属性和不同风险等级。

## 5 软件确认与批判性思维

### 5.1 概述

在医疗器械质量体系软件的整个生存周期，需要采取适当的控制措施，以确保软件按预期运行。融入批判性思维并应用选定建立信任的一些活动，可建立并保持软件的确认状态。图1为生存周期部分阶段的典型活动和控制措施的概念视图，从决定某个过程实现自动化到软件退役或不再用于医疗器械质量体系。虽然图1描绘的是一个顺序模型，但实际上，此过程在元素定义、风险识别和批判性思维应用上都具有迭代性。

在开发用于医疗器械质量体系的软件时，从工具箱中选择基本的建立信任活动就是选择软件开发生存周期模型。所选模型宜包括批判性思维活动，以便在各种生存周期活动开展期间选择其他适宜的工具。所采用的分析评价结果推动作出一系列最重要的信任建立活动的选择，以确保软件按预期运行。本文件并不意味着暗示或规定使用任何特定软件开发模型。然而，为了简便起见，本文件以瀑布式开发模型为例，使用各个阶段通用名称来解释批判性思维的概念。其他软件开发模型（例如迭代式、螺旋式）只要融入了批判性思维和适合的工具，均能使用。

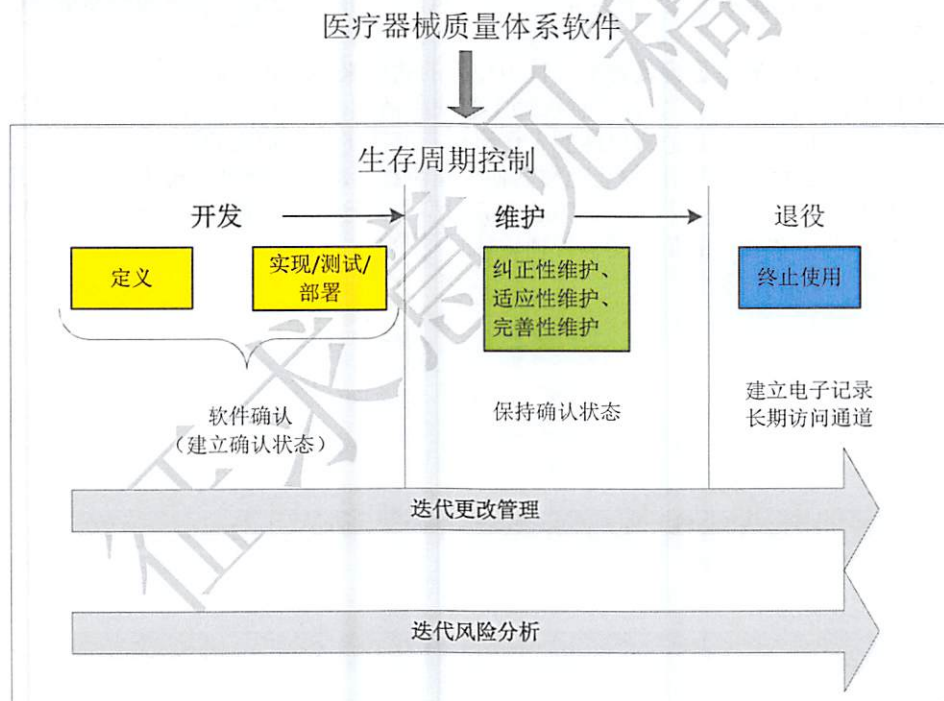


图1 生存周期控制

当考虑在一过程中使用软件时，宜通过调查其预期用途，来确定该软件是否用作医疗器械质量体系过程的一部分。如果是，则宜对软件的预期用途进行确认。尽管本文件描述了医疗器械质量体系软件的确认方法，但同样的方法也是评价软件是否满足定义的需求的良好实践。软件确认最关键的部分是开发/购买正确的软件工具，以便能够按照制造商的预期支持这些过程。这意味着需求宜准确确定，以评价开发/购买的软件能否满足预期用途的需求。适于验证的技术要求与适于确认的过程要求同等重要。在考虑在一过程中使用软件时，该软件可能与其他软件交互或存在接口。

在生存周期的开发阶段，执行风险管理和确认策划任务，以收集信息并在以下四个方面推动作出决定：

- 所投入的工作量以及对文档和可交付成果的审查程度；
- 文档与可交付成果的内容范围；

- 工具箱中工具的选择和应用工具的方法；
- 应用工具方面的工作量。

在这四个方面做出决定的主要推动因素是过程风险和软件风险。但是，其他推动因素也能影响决定，包括软件和过程的复杂性、软件类型和软件的固有属性。

确认策划过程包含两个不同的元素。第一个确认策划元素涉及确定文档的严格程度以及评审由此产生的可交付成果所要采用的审查程度。此要素中的决定主要由过程风险分析结果推动。第二个确认策划元素推动从工具箱中选择工具来实现、测试和部署该软件。工具选择主要由软件风险分析推动。这些策划步骤来自不同类型的风险分析，并在本文件中描述为独立的活动。但是，很多时候这些步骤被合并为一个活动，其中包括风险分析的不同方面以及继续进行确认的最终选择。

在软件生存周期的开发阶段，风险管理和确认策划任务用于定应用于软件的适当工作量级，并确定应用哪些建立信任的工具。这种方法可以完成适当的增值活动和验证任务，这是建立一种已确认状态的基础。一旦执行这些活动和任务，在确认报告中引用这些工具及其相关结果，以支持软件确认的结论。

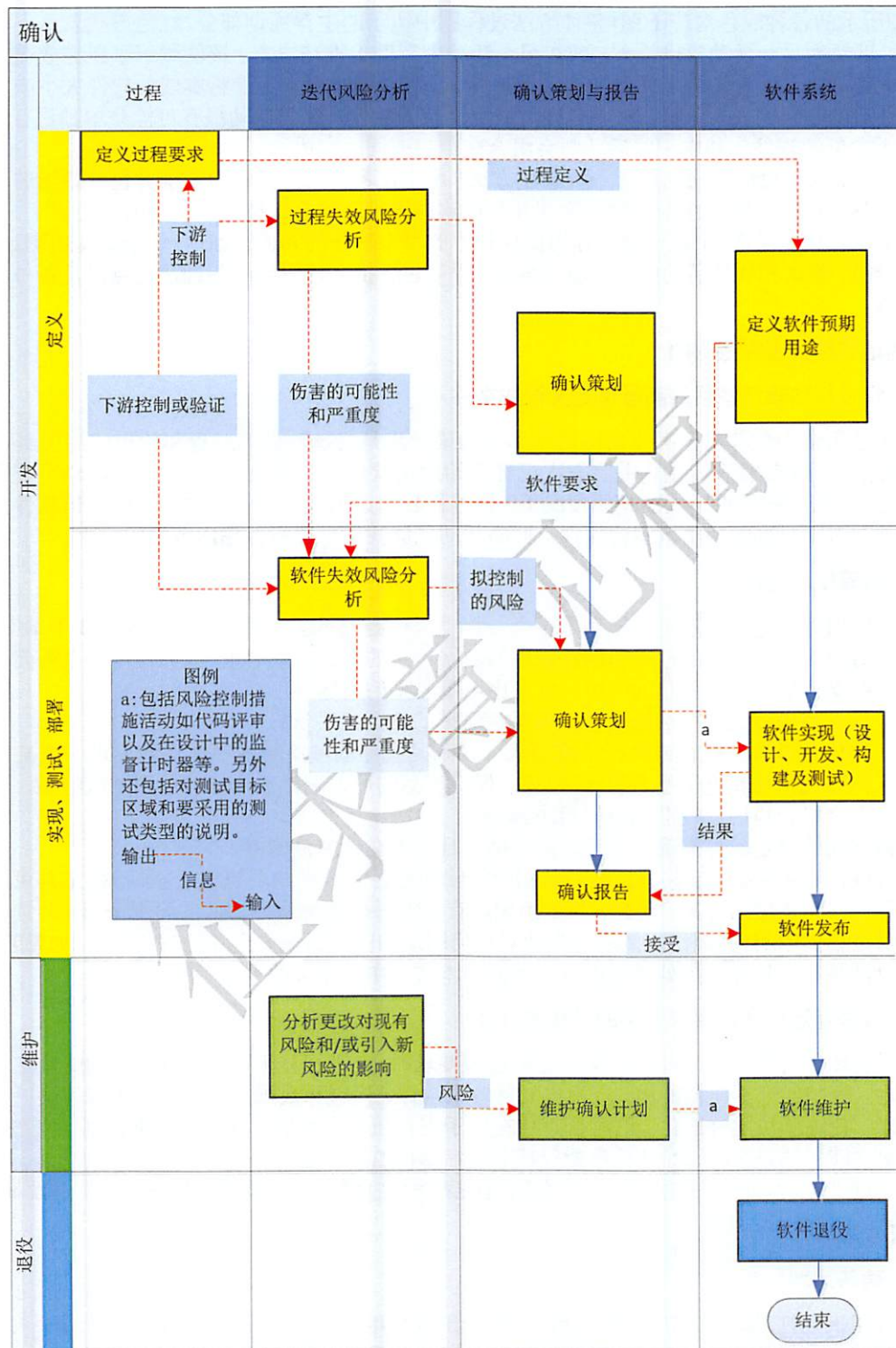
软件部署完成后，将进入软件生存周期的维护阶段。在此期间，软件将予以监视、改善，并按照业务需求或法规要求的更改而更新。更改控制活动使用与生存周期的开发阶段所应用的初始方法相同的概念。然而，此时评估的是更改对预期用途、失效风险、初始开发阶段的风险控制措施以及对软件本身任何功能的影响。

退役阶段是通过删减过程或替换过程中使用的软件来移除软件的行为。

图1中显示的活动反映了软件生存周期主要控制活动。其他的工作流程包括项目管理、过程开发、供应商管理（若适用）及其他有可能的工作流程，取决于正在实现的软件。

图2描述了软件生存周期控制活动和其他工作流程活动中的批判性思维。批判性思维活动体现在迭代风险分析和确认工作流程中。在组织的业务模型中对这些工作流程进行清晰、正式的定义非常重要，以确保程序从业务和监管的角度妥善管理软件。





注：在使用术语“develop”或“development”时，是关于软件确认状态的开发。

图2 生存周期控制 workflow

图2所示的各种颜色对应于图1整体方法过程图中所示的生存周期部分。红色虚线表示一个活动输出的信息，以及为另一活动提供输入，或在另一活动中有助于推动决定。该图展示了在完成需要输入的活动之前，获取输入信息的需求如何推动活动排序。值得注意的是，不管将实现的软件大小或复杂程度如何，所有活动都要完成。但是，对于更庞大或更复杂的软件，此类活动很有可能是单独进行的；而对于更小或更简单的软件，其中一些活动则会组合在一起或同时完成。

总之，批判性思维方法描述了一种系统方法，用于在各种工作流程中识别并包含适当的建立信任的活动或工具，以支持软件在发布时已确认并且在软件退役之前会保持确认状态的结论。

以下条款提供了图1中描述的生存周期控制中的每个阶段的其他详细信息。条款使用图2中所示的迭代风险分析、确认和软件活动的工作流程描述，提供包含批判性思维的各种决定点以及各种决定推动因素的观点。

## 5.2 确定软件是否在范围内

### 5.2.1 将过程和软件使用的高层级定义形成文件

确定是否认为软件用于医疗器械质量体系的第一步是将该软件的过程和使用的高层级定义形成文件。当很容易知道软件在范围内并且有人已经开始定义软件全部的预期用途时，此活动可能看起来价值很小。然而，在这些假设不太清楚的情况下，将过程和用途形成文件可以清楚地确定该软件是否在范围内。此外，对于已确定超出范围的软件，此类活动能够找出导致软件超出范围的原因。

### 5.2.2 监管使用评估

监管使用评估可用于确定该软件是否为一款“医疗器械质量体系软件”，因此属于本文件的范围。首先确定适用于使用软件的过程和软件管理的数据记录的特定法规要求。能够使用一系列问题帮助充分理解软件在支持法规方面所起的作用。宜考虑以下类型的问题：

- a) 软件的失效或潜在缺陷是否影响医疗器械的安全或质量？
- b) 软件是否自动执行或执行法规要求（特别是医疗器械质量管理体系的要求）的某项活动？示例可包括捕获电子签名和/或记录，保持产品的可追溯性，执行并捕获测试结果，保持 CAPA、不合格、投诉、校准等数据日志。

任何问题的肯定回答都确定了软件需要确认且在本文件的范围内。

有时候，可能很难确定某一过程以及相应的软件是否为质量体系的一部分。一些工具可能与实际医疗器械存在很大程度的分离。因此，每个组织均宜仔细考虑这种边界软件的周围情形，并宜完全理解软件失效对过程的影响，并最终理解软件失效对任何制造的医疗器械安全和功效的影响。如果答案不确定，最好的办法是将该软件视为处于范围内，并采用本文件中所定义的方法。

### 5.2.3 与医疗器械法规要求无关的过程和软件

当过程或软件包含的功能超出医疗器械法规要求时，宜进行分析以确定该软件的哪些部分可认为处于范围内，哪些部分不在范围内。宜根据软件各个组件、模块和数据结构之间的集成程度以及组织的合规性需求，使这些决定合理化。这种合理化在用于支持质量体系的软件尤其重要，例如大型、复杂企业资源规划（ERP）的软件。ERP软件可能包括非医疗器械监管过程（如会计和财务）的功能。但是，这种功能对于商业运营可能至关重要，并且必须满足某些政府要求（例如萨班斯—奥克斯利法案的要求）。

## 5.3 开发阶段

### 5.3.1 确认策划

在应用批判性思维时，确认策划活动的第一部分使用过程风险分析（见附件B）的输入以确定用于文件编制的工作量的基础，并推动从工具箱的“定义”部分（见表A.1至表A.5）选择工具。第二部分使用软件风险分析的输入，以推动从工具箱中选择实现、测试和部署工具。一经执行，活动和软件的确认状态随即建立，确认的证据随即形成确认报告文件。

在开发阶段能够应用许多开发生存周期模型。本文件不提倡或推荐任何一个模型；但是，期望采用某种受控方法。这种受控方法的基础是在实现、测试和部署之前定义需求（包括预期用途）的概念。这对于确定软件满足预期用途的确认至关重要。

### 5.3.2 定义

#### 5.3.2.1 定义阶段要求

在定义阶段完成的活动包括过程的定义、该过程中软件预期用途的定义以及基于该过程中已确定的固有风险对确认工作量的计划。图3描述了所选瀑布模型示例中开发阶段的这一部分。

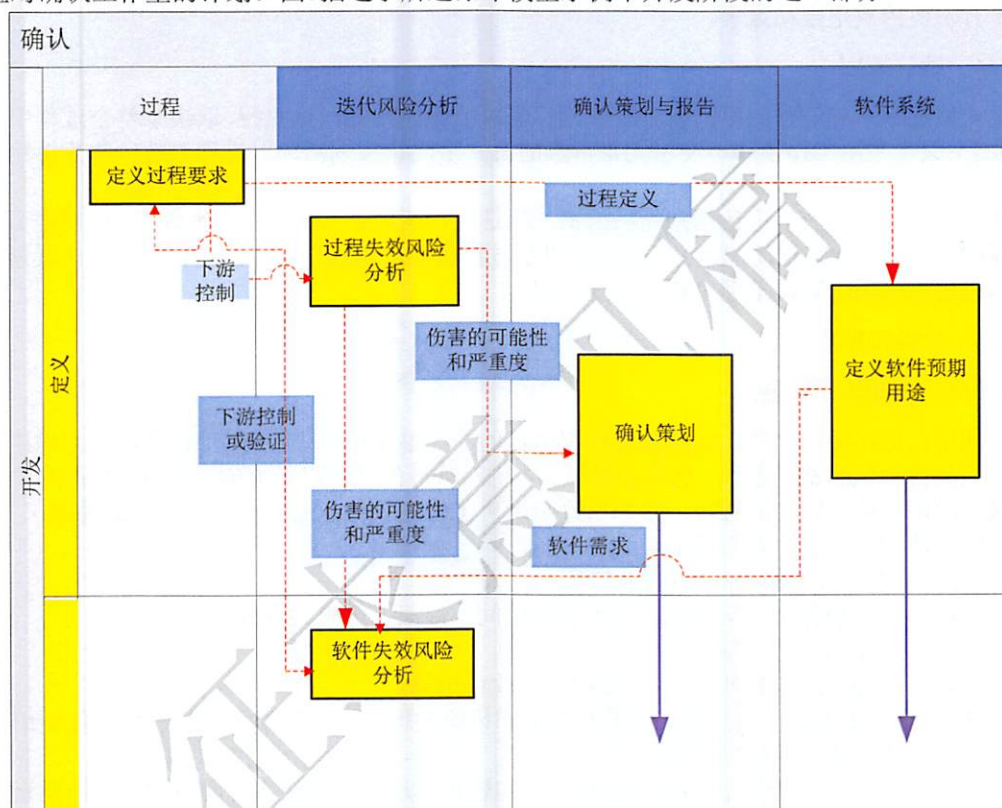


图3 生存周期阶段：定义阶段工作流程

#### 5.3.2.2 过程要求

应用生存周期控制的第一步是定义整个过程的目的和作用，特别是拟由软件控制的部分。最好的实现方法是有适当的主题专家参与，并且定义过程的所有有关方面和活动，不管其是否由软件控制。受益包括：

- 能够清楚地辨别法规要求；
- 能够清楚地辨别在该过程的范围内特定软件的预期用途；
- 能够按照程序或其他方式清楚地识别并处理不受特定软件控制的过程问题和活动；
- 识别软件上游和下游的过程活动，并在软件失效风险评估中和设计软件失效的风险控制中予以考虑。

过程定义活动为在生存周期中后续所做出的决定奠定了基础，对致力于增值的、基于风险的活动上的工作至关重要。

#### 5.3.2.3 过程失效风险分析

在风险分析过程中，将考虑软件与医疗产品最终的安全和有效性之间的关系。还宜考虑以下内容：

- 对人类造成伤害的风险：这包括对患者和用户的直接伤害，以及当控制医疗器械制造或质量的软件出现故障导致医疗器械失效，从而造成的间接伤害。
- 监管风险：如果软件失效可能导致监管机构要求的记录（例如 CAPA、投诉、设备主记录或设备历史记录）缺失或偏离质量体系和制造程序，则要考虑不符合监管要求的风险。
- 环境风险：软件运行环境的风险，包括物理环境和虚拟环境。

其他类型的风险能够纳入本模型，但本文件的范围和讨论的降低风险的工具并不针对这些风险。本文件重点是在过程失效的范围内确定与软件失效相关的人身安全风险、监管风险和環境风险。

宜将风险分析的结果形成文件，因为这些结果是从工具箱中选择工具以及证明确认活动所采用工作量水平有价值的决定推动因素。

#### 5.3.2.4 确认策划

确认程度和客观证据的范围需要确保软件需求能够始终如一地执行，取决于整个过程中软件的临界值。因此，关于采用工作量和可交付元素审查的第一个确认策划活动以过程失效风险分析输入为唯一根据。

该确认策划活动产生确认策划文档的第一次迭代。策划包括选择“工作量”（即决定）以及做出这些选择的理由（即决定推动因素）。理由宜以某过程失效造成的伤害风险为依据。确认策划过程中宜提供批判性思维应用于确认策划过程的客观证据。

#### 5.3.2.5 软件预期用途

##### 5.3.2.5.1 预期用途的要素

预期用途旨在提供软件功能及其在过程中目的的完整概括。具体来说，预期用途系描述和解释软件如何适应其自动化的整个过程、软件的功能，人们对软件的期望以及人们在设计、生产和维护安全的医疗器械方面可以依赖软件的程度。预期用途是用于了解与软件使用有关潜在风险的关键工具。

预期用途的三个主要要素是：

- 与以下内容有关的目的和意图：
  - 软件的使用（例如人员、内容、时间、原因、地点、方式等）；
  - 软件的监管使用；
  - 过程内或与其他软件和/或用户的软件边界。
- 软件使用要求。一般而言，随着复杂性和风险的增加，该要素增加了有关软件使用更详细的信息（例如用例、用户需求等）；
- 软件需求。随着复杂性和风险增加到宜向软件实现者提供明确方向的程度，该要素提供有关软件期望的更具体、更详细的信息。

宜由对法规、质量体系 and 受控过程具有适当技能和经验的人员，对预期用途正式控制和批准。

鉴于我们宜确认“预期用途”，除非软件的预期用途定义充分，否则确认无法进行。

以下条款提供了有关软件预期用途要素的更详细的信息。

##### 5.3.2.5.2 软件目的和意图

软件目的和意图涵盖三个要素的信息：软件使用、监管使用和边界定义。

###### a) 软件使用

- 在定义软件的使用时，宜考虑以下问题：内容、原因、方式、人员、地点、时间。问题的答案探讨如何使用该软件来满足过程要求。这种探讨有助于识别表1所示的软件定义的基本信息。
- 对软件描述有意义的回答宜包含在既定预期用途定义中。



表1 示例问题和答案

问题	答案
软件解决什么问题？	在有效并准确地汇集产品缺陷数据以进行趋势分析中，存在问题。
软件为什么有用？	软件能够实现汇集来自全球各地的数据并进行趋势分析。
软件如何解决问题？	软件推动数据收集过程并自动汇集和计算趋势信息，或者软件并不推动该过程，但被动收集用于汇总和计算趋势信息的数据。
谁使用该软件？	“质量保证与运营部门”使用该软件。
软件在哪里使用？	该软件可通过美国、欧洲和日本的地址访问。
软件什么时候使用？	在全球各地的正常工作时间（即每天，周一至周五）访问该软件。
注：这些示例问题并不详尽。	

## b) 监管使用

——在评价监管使用时，可以进一步探讨所回答的问题，以确定软件是否在范围内（见 5.2）。展开所有为“是”的答案，包括得出这些结论的原因。既然已经确定软件在范围内，则需确定对人类（医疗器械用户以外）或对环境的任何潜在伤害。以下所有问题都将用户考虑指向作为法规的一部分所要求的要素，如公共卫生健康、安全以及电子记录和签名的有效性或真实性等。

- 软件失效或潜在缺陷如何影响医疗器械的安全或质量？
- 软件如何自动操作或如何执行法规（特别是医疗器械质量管理体系的要求）所要求的某项活动？
- 软件如何对人（医疗器械用户以外的）或环境造成伤害？

## c) 软件边界

——定义要通过软件控制的过程部分（过程中的边界）以及定义存在软件接口的位置（与其他软件的边界），有助于促进确认工作的有效性和效率。例如，将多个软件产品作为一组进行确认比单独进行确认可能更有效。还宜考虑，各种分组策略如何影响正在进行的维护阶段活动的效率。

- 过程中的边界；
- 识别过程中软件的边界，清楚地确定了预期用途中要包含的方面。软件可能自动运行一个完整过程或自动运行多个活动的一部分，还能作为该过程的数据存储库。了解软件在该过程内所起的作用，有助于确定与软件某个潜在失效相关的风险；
- 与其他软件的边界；
- 当与其他医疗器械质量体系软件或医疗器械软件进行外部接口时，识别应用程序之间的所有接口非常重要。确认工作通常包括作为方法固有部分的内部接口，但不宜忽视软件的外部接口。软件应用程序之间的所有接口均宜纳入批判性思维过程。

## 5.3.2.5.3 软件使用要求

软件使用要求由良好文件化的且可追溯的元素组成，这些元素为软件目的和、意图提供额外层次的细节。这些要求从用户的角度或产品需求的角度，提供了对系统使用场景的深入见解。能够以用户需求、用例或其他以用户为中心的需求定义等形式捕获用户的视角。产品需求视角捕获的是受系统影响的医疗器械的需求，在某些情况下，可能包括对特定器械要求的引用或该软件可能影响生产线的概述。

## 5.3.2.5.4 软件需求

由定义元素活动组成，这些元素记录完整并可追溯，规定了软件为满足其目的、意图和使用要求需要执行的操作。软件需求包括系统设计的输入，系统配置输入以及测试活动输入。

## 5.3.3 实现、测试和部署

## 5.3.3.1 所需活动

在实现、测试和部署阶段中完成的活动包括：

- 设计中确认严密程度的策划；
- 开发和配置；

- c) 软件的构建;
- d) 基于已识别风险的软件测试。

### 5.3.3.2 软件失效风险分析

软件失效风险分析的关键是确定并记录与软件失效相关的固有风险，并确定任何控制措施（包括正在分析的软件之外的过程和软件控制）。然后使用该分析得出现实有效的确认方法。

在评审可归因于软件失效的风险时，考虑正在分析的软件之外的任何过程控制，其构成风险控制措施。这种风险控制措施能够减少软件失效的影响，从而减少对软件的依赖，进而减少对测试（检查）和文档（客观证据的收集）的依赖，以确保软件的安全运行。包括这些考虑因素将有助于确保在整个过程的背景下审查软件。

附录B中的模型并不代表一个包罗万象的公式。结果分析为从工具箱中选择用于软件确认的工具提供了输入。

### 5.3.3.3 确认策划

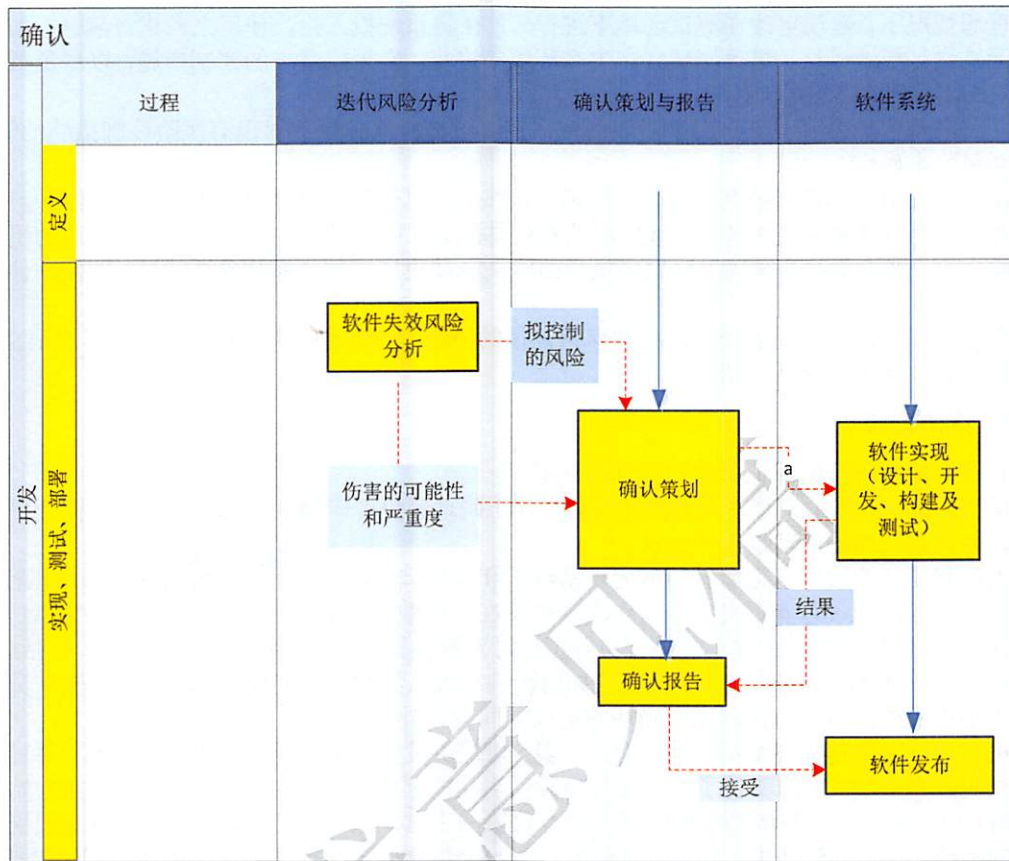
此活动使用预期用途定义和软件风险分析结果作为输入，以确定风险控制措施并从工具箱中选择用于确认软件的工具。

工具选择过程由能胜任的人员参与这一点很重要。能胜任的人员不必是软件专家，但理解失效对过程的影响以及该过程自动化软件失效的固有风险。来自不同学科（监管、质量、临床等）的人员宜参与对于任何高度复杂的软件或存在与其失效相关的高风险的软件的策划过程。

确认策划活动生成一个形成文件的计划，该计划描述做出的选择（决定）以及做此选择的原因（决定推动因素）。确认策划提供了用于选择增值建立信任活动的理由的形成文件的证据，这些活动用以确保软件如预期运行。

### 5.3.3.4 软件实现（设计、开发、构建和测试）

本阶段包括工具箱中许多工具的实际应用。工具（确认计划中确定的所需活动）在设计、开发、构建和测试步骤中进行（见图4）。



<sup>a</sup> a 包括风险控制措施活动如代码评审以及在设计中的监督计时器等。另外还包括对测试目标区域和要采用的测试类型的说明。

图4 生存周期阶段：实现、测试和部署阶段工作流

### 5.3.3.5 确认报告

为确保软件按预期实施，建立充分信任的活动（包括从工具箱选择的工具）一经完成，宜在最终的确认报告中引用活动和（有可能）活动结果。报告的正式评审和批准为所有形成文件的客观证据提供一份参考摘要，这些证据支持该软件已对其预期用途完成确认的结论。

### 5.3.3.6 软件发布

确认结束后，宜形成一个正式受控方法来发布软件。确定的控制措施宜确保并确认：投入使用的软件与通过确认报告中所引用的信任建立活动评估的软件相匹配。否则，合理的理由和控制措施宜确保并证实结果足以代表已发布软件在其预期环境中的性能。

## 5.4 维护阶段

### 5.4.1 进入维护阶段

阶段进入准则：软件发布后，软件维护阶段随即开始。

活动范围：维护阶段活动包括在适应、管理和控制不同类型更改的同时，确保软件保持在经过确认的状态。有些类型可能只涉及软件使用所处过程的更改。

对任何已确认系统的更改宜根据方针和程序以受控的方式进行。

理想情况下，建议更改在测试环境下进行，并在将系统投入生产使用之前进行确认。如果在测试环境下无法进行更改确认，更改测试宜在生产环境中进行，宜采取适当的控制措施，以尽量减少对生产环境或直接对产品的不良影响。

从工具箱中选择哪些工具用于更改确认，宜通过分析软件更改对现有风险控制措施的影响或/和引入的新风险来确定。

由于软件的实际使用或其配置可能随着时间变化，尽管努力对其进行控制，使用维护阶段专用工具（例如：定期监视实际使用情况，或实时监视软件配置）可能是适宜的。如果预期用途的更改导致更高的风险水平，则该更改能够触发比原来执行的确认活动更广泛的一系列确认活动，即使软件没有发生更改。

关于选择开展更广泛的确认活动并保留证据的决定，宜作为确认策划的一部分形成记文件，以提供证据证明软件保持经确认的状态。

#### 5.4.2 维护策划

宜在维护阶段开始前记录维护策划的证据。

理想情况下，维护策划在开发阶段开始。宜正确理解更改影响软件确认的方式，检查更改对风险的影响，并策划适当的活动以维护确认。

大型和复杂的软件有可能必须适应日常维护和性能调整活动，而不影响软件按预期执行的能力。在开发阶段进行维护策划能定义哪些操作活动能够在不影响确认的情况下完成，哪些更改需要开展确认。在软件进入维护阶段之前，宜策划并讨论确定何时对软件开展进一步确认活动的方法，包括底层软件（例如操作系统、数据库管理系统）中的更改如何影响已确认的软件。这有助于培训软件操作员识别这些边界并识别正常操作活动与任何需要确认的更改之间的差异。

可追溯性分析是管理维护活动的有用工具。可追溯性分析通常是初始确认的基石，并且通常通过可追溯性矩阵来推动。矩阵映射了测试或其他验证活动的要求、风险控制措施等。如果在初始实现期间表现良好，可追溯性分析将成为维护阶段一个有价值的工具，有助于识别更改的影响以及识别适当的确认更改的活动。在简单软件中，此类分析可能包含一个由要求至实现和验证的单级追溯。但是，复杂软件可能需要一个多级的矩阵，将顶级的功能分解为较低级别的要求，进而再分解为实现和验证。也可以嵌入其他信息，例如，可在追溯矩阵内指定认为风险极高的软件部分，可能还指出了其他确认活动。

#### 5.4.3 维护阶段内的维护类型

软件在发布使用后可能会发生变化的原因有很多。其中较常见的维护更改类型包括：

- 纠正性维护更改，以纠正软件中的错误和缺陷；
- 完善性维护更改，以增强性能、可维护性或其他软件属性；
- 适应性维护操作，以更新软件操作环境（例如操作系统更改、与软件连接的系统硬件或其他应用程序的更改）。

#### 5.4.4 过程更改：对风险控制措施的更改

当由软件完全或部分控制的过程发生更改时，宜进行影响分析以重新评价风险控制措施。

由软件完全或部分控制的过程可能独立于软件而进行更改。当一个过程发生更改时，理解该更改如何影响软件确认状态是非常重要的。过程更改能够影响软件的预期用途或其他有关软件支持信息的预期用途。

过程更改也可能影响作为确认基本原理一部分的软件现有风险控制措施。由于软件是过程的一部分，因此下游控制可能是软件的重要风险控制措施。如果下游控制被正确识别为软件确认基本原理和过程定义的一部分，则提议过程更改的影响分析更容易进行。以对软件和软件运行过程建立信任的方式进行维护，影响分析对其是必不可少的。

#### 5.4.5 紧急更改

紧急更改宜遵循批准的过程。这些过程宜对制定和实施的理由、获取和记录更改部署的授权机制、确保风险得到适当评估和控制的规定以及调用紧急更改所需的任何活动（如培训、沟通、产品评审和处



理等)明确要求。在此情况下,执行适当评估和控制风险的规定代表了满足在发布之前更改确认满足法规要求所需的最小活动集。

软件更改可能需要在紧急情况下进行。通常,如果软件、操作系统或数据的完整性受到损害,或为了减轻潜在的有害情况,需要进行此类更改。

此外,有可能需要的在紧急更改发生后开展一系列活动,以充分评价更改的所有影响。依据由过程失效构成的综合风险,过程输出(数据或产品)可能需要进行额外控制,直到紧急更改后的所有活动全部完成为止。

造成过程中断的软件问题通常很明显,而找出细微的、隐蔽的问题比较困难。定期评估错误日志、帮助中心的请求、顾客反馈和其他缺陷报告,可以指出隐蔽的问题。此类监视技术能够发现那些不至于明显到导致错误报告,但却是可纠正的软件问题。因此,有可能需要开展维护活动,通过在未来发布版本中实施纠正来处理已识别的问题。此外,可以主动管理已发布软件中归因于此类软件问题的问题。

在利用维护活动纠正未来发布版本的问题后,宜评审已发布软件中所识别缺陷的历史影响并管理其后果。

如果软件确认取决于通过培训确保软件的正确使用,定期评价用户培训有效性则是另一种有助于保持确认状态的监视技术。

#### 5.4.6 维护预期用途

如果软件的预期用途发生变更,则宜确认新的预期用途,或停止使用新用途。对于后者,风险评估是为了确保在未经授权使用期间不会引入任何风险。

预期用途的变更是需要特别注意的,因为更改可能是细微且难以发现的,也可能非常明显。如果是细微的,目的和意图或软件使用要求发生变更,不一定导致具体的软件需求要素发生变更(见5.3.2.5)。这种更改可能是有意为之,或只是因为在新模式下简单地使用现有软件,而未意识到预期用途受到了影响。预期用途可能会随着时间的推移缓慢更改,或者用户可能会以非初始预期的方式开始使用该软件。由于这种转变,所部署的软件不再处于确认状态。

每次启动对已确认的软件进行更改时,均宜重新评审预期用途,确保预期用途与软件的实际用途保持一致。

#### 5.5 退役阶段

在退役阶段,宜记录软件的退役情况,并建立可以在任何需要的记录保留期内访问任何有关电子记录的方法。

软件退役活动高度依赖于要退役软件的类型。有些软件只执行一项活动并不存储任何数据。其他软件则可能像批量可追溯性或文件控制系统一样复杂,覆盖大量与产品相关的数据和与合规性相关的数据。对于需要存储数据的软件,宜生成一份用于处理数据的计划。需要考虑的问题包括以下内容:

- 是否有软件取代退役软件?
- 数据能否迁移至新软件?
- 是否宜将数据迁移至便携格式以便长期保留?
- 此类数据的数据保留要求是什么?
- 数据是否会存储于持久介质上?
- 如果是,存储说明或存储程序是什么?能否检索包含所有有关数据要求的数据?
- 维护可读的持久介质和软件的程序是什么?
- 是否会存储已归档的硬件平台,用以使用和检索已退役的应用程序?
- 如何维护存储的硬件?
- 作为投诉或CAPA调查的一部分,访问退役软件是否需要?
- 是否需要平台和应用程序重新创建软件程序?

## 6 文档

宜确保与软件生存周期控制活动有关的所有信息均得以形成适当文件并予以控制。

具有优质并高效的文档可产生两大好处：

- a) 完整的软件定义在文档中清晰明确地表达，能够充分了解软件的预期用途和期望的性能，并且能够了解对软件所做的任何和所有更改所带来的全部影响；
- b) 确认策划和实行情况的记录，为批判性思维所做决策提供了形成文件的证据。文档围绕所实施的评估或分析以及针对基于风险和有意义的建立信任活动的工具选择，可以对关于所实施的确认的有简洁理解。通过总结如何满足验收标准，文档提供的证据表明，已完成的活动可确保软件按预期执行，并为其自动化过程引入了可接受的风险级别。

生成文档的范围与软件确认的工作量直接相关。工作量宜与风险相称。因此，本文件中讨论的软件确认方法，根据过程失效的影响，决定文档的范围。该过程对人员或环境造成伤害的风险越大，文档预期的范围就越大。此外，较高的伤害风险，宜推动多个跨功能同行、更高层次的管理层或两者一起开展更高级别的文件审查。

将生存周期控制信息组织到文档中可能由于许多因素而有所不同，例如使用的技术以及软件的大小或复杂性。

宜以方便信息审核的方式组织信息，同时要有能力在软件生存周期的维护阶段保留确认状态的证据。

生存周期控制信息的捕获和记录方式，取决于确认实施各方的偏好和既定政策。有关如何将生存周期控制的客观证据打包呈现在文档中，软件确认各方拥有自由裁量权。从合规性评审角度看，宜建立确认计划和报告文档，以提供计划和执行的所有增值、建立信任活动的汇编，以确保软件按预期运行。本质上，该文档是基于输入（决定推动因素）所做选择（决定）的关键记录。这些输入体现了确认所采用的批判性思维过程，表明符合法规要求的完整软件解决方案已经开发完成，而且考虑了所有关键利益相关方及其需求。

注：术语“documentation”是指记录的信息主体，无论是记录在实际文档中还是记录在捕获信息的工具（如需求管理工具）中。

## 7 先决条件过程

本文件中介绍的方法旨在一个有效的质量管理体系内全面运作，以便更加有效。

能够对批判性思维方法的成功产生最积极的影响的质量体系内容包括：资产与基础设施管理（人力和硬件）、更改管理（包括配置管理）和供应商管理。这些内容的详细说明不在本文件的范围内；其中每项内容在其他行业标准和文件中都有阐述（参见参考文献）。此外，本文件无意将特定角色或职能（例如质量保证、管理和制造）与本文件中的活动相关联。每个公司的理念和人力资源基础设施将为实施确认活动决定可接受的角色。

## 附录 A (资料性) 工具箱

### A.1 综述

工具箱提供了一系列建立信任的活动，开展这些活动能够满足确认要求意图。工具箱并不是满足此目的的所有可用活动的详尽列表，但它提供了基于当前软件工程知识体系的初始程序集。其中一些活动是重叠或协同工作的，例如，正常的用例测试往往是软件系统测试的一部分，但这里关注的是活动的价值。这些活动将作为确认策划执行的基础。

活动的选择和执行宜适合于软件有关的风险。为了支持此选择，工具箱中的活动按照以下方案进行分类和标记。

- 必选：在任何情况下都要按照要求开展此活动。
- 可裁剪：选择并开展此活动的适当部分。
- 可选择：适当时，选择并开展活动。

工具箱能够量身定制以定义组织中使用的活动，并能够随着技术变化和教训的积累而不断发展，从而整合新的软件工程最佳实践。适当时，有些活动也将在标准程序中按程序进行。

### A.2 工具箱结构

为方便起见，这些活动被组织成五个最重要的软件生存周期过程活动。依据软件的范围和性质，宜在软件生存周期的各个阶段应用批判性思维，以识别和选择最适合于该软件的活动。

列表中出现的每个已命名的活动都有简短的定义以及关于该活动价值（即该活动对确认工作的贡献值）的描述。定义栏还包含可用于完成该命名活动的方法。

表A.1 开发阶段：定义

活动	定义
定义过程要求 (必选)	定义拟通过软件实现部分或全部自动化的过程（某个制造过程或某个质量体系过程）的活动。活动还描述了在过程执行或软件风险分析时可能予以考虑的过程内的任何验证或预防措施。 此活动的输出能记录在过程图或要求说明中，这些过程图或要求说明定义了业务、制造或质量体系过程中实施的活动。
过程失效风险分析 (必选)	确定过程失效对医疗器械安全性和有效性、制造人员、环境或质量体系的影响的活动。
预期用途 (可裁剪)	对于简单软件，活动能由几个句子或段落组成。对于大型复杂软件，活动能包括跨多个文件的扩充文档，或许还可包括具体的软件需求。风险也是决定预期用途定义深度的重要因素。 预期用途要素： ——软件目的和意图； ——软件使用要求； ——软件需求。
确认策划 (必选)	确认策划分两个阶段进行： ——在开发一阶段，定义确认文档中预期的详细程度和工作量、审查级别，并且选择定义阶段要包含的活动； ——稍后，在实现阶段，根据在定义阶段做出的决定和有关的风险分析活动，选择适当的确认活动。 确认策划输出的是一个计划，描述了将要执行的活动，为软件始终满足其预期用途要求而进行建立信任。
正式的软件需求评审 (可选择)	利益相关方基于预期用途对软件需求进行评审并达成一致的活动（过程、会议等）。

表A.1 开发阶段：定义（续）

活动	定义
软件开发生存周期模型选择 (可选择)	定义在软件全生存周期的开发阶段所使用的生存周期方法和控制的活动。通常仅用于复杂或有风险的软件。YY/T 0664—2020/IEC 62304:2006/ AMD1:2015 作为过程标准，对于一些软件有可能特别适合。
风险管理计划 (必选)	有关计划如何对软件进行风险管理的活动。风险管理计划的输出是一个计划，它定义了用于分析与软件风险相关的软件关注领域，以及选择风险分析的方法，如失效模式与影响分析（FMEA）、故障树分析或其他工具。
确定制造过程或业务过程的风险控制措施 (必选)	该活动是一种识别风险控制措施或危害控制措施（例如程序控制）的机制。包括持续监视以确保控制就位并正常运行。

表A.2 开发阶段：实现

活动	定义
软件失效分析 (风险分析) (必选)	软件失效分析是指确定软件失效对过程和过程失效分析中所确定关注领域的影响。
软件架构文档与评审 (可选择)	软件架构定义了软件元素的高级结构及其相互关系，同时记录架构，并对正确性、完整性以及软件功能执行能力进行评审。
设计规范 (可选择)	设计规范是对如何实现软件需求的精确阐述，通常包括软件或组件结构、算法、控制逻辑、数据结构、数据集使用信息、输入输出格式、接口描述等。
开发设计评审 (可选择)	开发设计评审是为评估一个或多个配置项所选定设计方法的进度、技术充分性和风险解决方案而进行的评审。
确定软件设计中的风险控制措施 (必选)	该活动确定了风险评估过程中识别的风险或危险的控制措施。识别风险控制措施宜是一个迭代过程，以允许持续监视并确保控制措施到位且正常运行（如程序控制、硬件冗余）。
代码评审或代码验证 (可选择)	代码评审或代码验证包括对软件源代码的同行评审，旨在查找和消除缺陷，提高整体代码质量。通过建立和遵守一组通用编码标准，能够增强代码评审和整体代码质量。
可追溯性分析 (可选择)	可追溯性分析系指对设计、代码、测试、风险或危险分析要求以及风险控制措施要求的可追溯性。还可能包括对过程要求的可追溯性。
供应商审核 (可选择)	供应商审核是指对达到必要级别的软件供应商体系进行评估，使买方确信供应商有足够的能提供安全和可用的软件。多种供应商审核方法都可能采用。

表A.3 开发阶段：测试

活动	定义
测试策划 (可选择)	测试策划宜定义测试活动的整体方法，以帮助软件满足其预期用途建立信任。但是，软件测试本身可能不足以确定软件是否适合其预期用途。其他验证技术可能需要与测试相结合，以确保确认方法综合、全面。 测试级别宜基于风险推动因素和风险因素，并宜提供适当的置信水平，以证明软件按照适当的测试方法满足要求和设计规范。这种测试可能包括开发人员测试、单元测试、集成测试、用户测试、负载测试、操作测试等。
单元测试 (可选择)	为验证某个软件元素（如某个单元或模块）或软件元素集合的设计实现而进行的测试。
数据验证 (可选择)	数据验证是指为确认数据的正确性而完成的活动。数据验证可以作为数据迁移、转换或测试工作的一部分或独立完成，并且适当时，也可能包括统计抽样。
集成测试 (可选择)	集成测试是一种有序的测试过程，此过程期间，软件元素、硬件元素或两者被组合并测试，以评价其交互作用，直至软件集成完成。

表A.3 开发阶段：测试（续）

活动	定义
用例测试 (可选择)	用例测试是功能测试的一种形式，其忽略系统或组件的内部机制或结构，侧重于响应所选输入和执行条件而生成的输出。每个用例都能具有与之关联的输入参数，每个参数都可能具有一组标识值，以模拟实际使用条件。使用描述完成某个目标的序列的预定过程能够连接一系列用例。
接口测试 (可选择)	接口测试是指确认软件应用程序之间的接口，考虑从输出到输入的整个数据传输路径。接口测试能够通过直接测试或 100%数据验证来完成。测试活动宜包括确保接口在规范限制或按正常和异常情况边界条件下按要求执行的策略。
回归测试 (可选择)	重新运行以往已正确执行过的测试用例，以检测软件开发和维护期间所做的更改或更正所产生的错误。
供应商提供的测试套件 (可选择)	供应商提供的测试套件能够测试软件解决方案的全部功能，并能在最终使用环境下提供对软件性能的极大信任。但是，宜评估此类套件是否适合所定义的预期用途及测试的完整性，包括对任何已采取的风险控制措施的测试。使用这种套件可能需要一份协议，要求供应商在软件的整个生存周期内维护测试套件。
软件系统测试 (可选择)	软件系统测试是对一个集成的硬件和软件系统进行测试以验证软件是否满足其规定要求的过程。这种测试能够在开发环境和目标环境下进行。 软件确认与软件系统测试不同，因为软件确认是验证软件是否适合在其预期环境中使用及其预期用户使用。软件系统测试仅验证软件需求是否已成功实现。 对于由软件控制的生产系统，过程确认测试能够涵盖部分或全部这些测试。对于质量体系应用程序，执行软件说明书中所要求的所有步骤能够涵盖软件测试要求。
用例测试 (可选择)	用例测试是指基于用例进行的测试，包括用例中定义的替代过程和错误条件。
正常情况测试 (可选择)	正常情况测试是指使用常规输入进行测试。
稳健性测试 (压力测试) (可选择)	稳健性测试宜证明：软件产品在得到意外的、无效的输入时表现正常。稳健性测试用于评估某个系统或组件达到或超过其规定要求的界限。 识别充分的此类测试用例集的方法包括等价类划分、边界值分析和特殊用例识别（错误猜测）等。
输出强制测试 (可选择)	输出强制测试是指选择测试输入，以确保系统正确生成所选（或所有）输出。输出强制涉及编制一组测试用例，这些测试用例被设计用来从系统中生成特定输出。重点是创建所需输出，而不是启动系统响应的输入。
输入组合测试 (可选择)	输入组合测试是一种测试技术。通过该技术，使软件单元或系统在运行期间可能遇到的输入的组合。
Beta 测试 (可选择)	Beta 测试是由供应商在实时环境中针对一小部分客户进行的测试。
性能测试 (可选择)	性能测试衡量软件系统按照其所需的响应时间、中央处理器（CPU）使用情况以及其他运行中的量化特征的执行程度。

表A.4 开发阶段：部署

活动	定义
用户程序评审 (可选择)	用户程序审核是对与软件使用相关的用户程序和说明的评审。这样的评审确保正确定义软件的用途。
软件应用的内部培训 (可选择)	内部培训是指特定于软件的形成文件的培训活动。
安装鉴定 (可选择)	安装鉴定是建立软件安装和运行符合形成文件的安装说明的信任。
运行鉴定与性能鉴定(实施过程确认时) (可选择)	运行鉴定建立了制造过程和有关系统能够始终如一地在既定限制和公差范围内运行的信任。 性能鉴定确定了过程的有效性和再现性。
最终验收测试 (可选择)	最终验收测试是指在最终部署之前对系统进行的测试。又称上线测试。
操作员认证(可选择)	操作员认证是确认已接受培训的人员在培训中表现出能力的证明。

表A.5 维护阶段

活动	定义
维护策划 (可裁剪)	维护策划相关方法如下： 前瞻性策划。此方法涵盖软件更改的前瞻性规划和预期。这种方法能够在软件进入维护阶段之前的初始实现期间使用，但也能够在维护阶段的任何时间使用。 更改待定策划。此方法涵盖对软件某一更改待定时所做计划。该规划通常侧重于更改待定的活动，该策划在软件维护阶段完成。
已知问题分析 (可选择)	已知问题分析是一个过程，通过该过程供应商已知的软件任何问题和所有问题都将评估其对已安装软件的使用或确认状态的影响。
兼容性测试 (可选择)	兼容性测试是确定两个或多个软件系统信息交换能力的过程。
基础设施兼容性分析 (可选择)	基础设施兼容性分析是确定软件基础设施的更改如何影响已安装软件的过程。更改可能包括对硬件或系统位置的更改。
系统监视 (可选择)	系统监视包括用于在软件生存周期的维护阶段评估软件系统总体健康状况的技术。系统监视方法可能包括以下： <ul style="list-style-type: none"> <li>——定期评估预期用途是否发生变更；</li> <li>——最终用户的实际使用状况；</li> <li>——培训效果评价；</li> <li>——缺陷分析；</li> <li>——数据审核。</li> </ul>
备份和恢复过程 (可选择)	备份和恢复过程包括系统备份，备份介质的存储和保留，以及从备份介质还原数据的恢复过程。
运行控制 (可选择)	除备份与恢复过程、监视和报告之外，还能通过运行控制来帮助确保软件按预期运行。常用方法包括： <ul style="list-style-type: none"> <li>——信息安全措施；</li> <li>——访问权限管理；</li> <li>——数据库管理；</li> <li>——归档；</li> <li>——应急策划。</li> </ul>
回归分析 (可选择)	回归分析包括可追溯性分析或影响分析等任务。其目的是确定维护系统已确认状态所需的活动。

## 附录 B (资料性) 风险管理和基于风险的方法

### B.1 综述

如本文件核心部分所述，确认内容和严格程度由与软件相关的风险所决定。

为了扩展这一概念，请参考ISO 14971标准。ISO 14971描述了适用于医疗器械的风险管理过程，但基本原则及术语能够应用于遵循ISO 14971的软件。

### B.2 术语

以下定义或引用于ISO 14971，或基于ISO 14971中的定义。

- 危险：可能导致伤害的潜在根源；
- 危险情况：人员、财产或环境暴露于一种或多种危险中的情形；
- 风险：伤害发生概率和该伤害严重度的组合；
- 伤害：对人健康的损伤或损害，或对财产或环境的损害；
- 严重度：危险可能后果的度量；
- 风险控制措施：将风险降低或维持在规定水平的措施。

### B.3 基本原则

基本原则是将与软件相关的风险降低到可接受的水平。为了满足这一原则，制造商需要识别使用软件可能出现的危险情况、估计有关风险，并评价这些风险是否符合接受准则，如果没有其他规定（如法规要求），接受准则由制造商定义。

特别是由于软件本身不会受到伤害，因此软件控制的整个过程均需进行风险管理。

### B.4 识别危险情况和估计风险

遵循ISO 14971标准的方法，从预期用途开始识别可能的危险和危险情况，并估计相关风险。但是，所考虑可能的伤害与ISO 14971考虑的内容有很大不同。

生产和质量体系失效很少对使用由软件控制制造或质量的医疗器械的患者或用户造成直接伤害。在这种情况下伤害几乎都是间接的，对器械的伤害最终会成为医疗器械患者或用户伤害的来源。这并不是说间接伤害在任何方面都不那么严重。事实上，在某些方面，生产和质量体系失效的严重度可能被认为更严重，因为生产体系和质量体系中的一次失效可能导致许多医疗器械失效，最终在被检测发现之前伤害许多患者。单个器械的软件失效一次只能伤害一位患者。

生产或质量体系失效都可能导致直接和间接的多重伤害。请注意，以下列表内的伤害并不相互排斥。每种伤害都有可能对患者或医疗器械用户造成间接伤害。示例包括但不限于以下内容：

- 对医疗器械的伤害：
  - 机器工具不产生临界公差；
  - 校准系统错误校准了药物输送装置；
  - 灭菌器控制器失效，致使非灭菌组件产生；
  - 最终测试系统失效，检测不到潜在器械缺陷。
- 对制造过程的伤害：
  - 一个由软件控制的过程失效，由于采用人工解决办法而降低生产率；
  - 一个由软件驱动的过程失效导致很大比例的超差部件产生。
- 对合规性的伤害：
  - 投诉处置系统误报失效统计数据，从而使现场报告缺陷无法察觉；
  - 医疗器械服务或维修系统未能突出显示以前未被发现的缺陷问题趋势；
  - 植入医疗器械的数据库发生完整性缺失；

- 与制成品安全检查有关的质量控制记录缺失；
- 合规数据缺失；
- 医疗器械确认数据缺失；
- 无法控制和报告所制造器械中的软件配置；
- MRP系统未能提供可追溯性结果，导致无法通知潜在用户医疗器械召回。

——对制造人员或环境的伤害：

- 操作人员受伤；
- 释放有毒化学物质。

在分析依存于使生产和质量体系实现自动化的软件的有关风险时，宜考虑所有类别的伤害。

风险估计包括估计可能的伤害严重度以及发生该伤害的可能性。

严重度估计通常通过分类来完成（例如见YY/T 0316-2016，附录D或与接受水平相有关的G.4（见B.5））。

事实证明，伤害的可能性很难估计，特别是在考虑软件故障造成伤害的可能性时。在这种情况下，应该记住，软件故障只是导致伤害的其中一个因素，可能还涉及软件之外的其他几个因素（事件序列）。假设未知事件可能性的最坏的情况以及最终可能造成伤害的最坏情况是有用的。

YY/T 0664—2020采用了类似方法。作为决定过程控制严格性的基础，YY/T 0664—2020假设了软件出现故障的最坏情况可能性，但允许考虑与软件之外的一系列事件相关的较低的伤害可能性（见YY/T 1406.1）。

## B.5 风险评价

一旦估计有风险，就需要对风险进行评价，看其是否可以接受。如果不能接受，制造商宜确定并实施风险控制措施，以将风险降低到可接受水平。

也许风险管理中最困难的活动是确定什么是可接受的风险水平。这种决定在很大程度上取决于潜在伤害的严重度。每个制造商都需要建立准则，以定义和记录风险的可接受性，并以允许评价符合这些准则的格式识别所有风险。一般而言，如果可接受的风险降低到一个人可以在其同行、管理层或审核员面前能够从容应对的水平，则该风险很可能处于适当水平。

推荐可接受性阈值超出了本文件的范围，但关于可接受性阈值设置过程的一些建议是适当的。

- 要明确。“尽可能低”或“与任何其他产品一样安全”等接受准则是无意义的。接受准则宜同测试规范一样，以便可以客观地确定是否已满足可接受性准则。
- 如果难以估计伤害的可能性，则接受准则只能基于严重度。
- 接受准则可与软件过程控制的预定义选择（即表A.1至表A.5中所列工具的选择）联系起来。
- 尽早确定接受准则。一旦识别潜在伤害风险，立即设定目标或规范。在尝试控制风险之前设定可接受性目标很重要。一旦做出控制风险的一次尝试后，对可接受性认知通常会提升到更高风险水平。提前记录可接受性准则可以防止提升。
- 记录确定风险可接受性的理由。此类文档对于将来维护过程以及将思维过程传达给监管调查员非常有用。

## B.6 风险控制

### B.6.1 不可接受的风险

如果风险被评价为不可接受，制造商宜确定并实施风险控制措施，将风险降低到可接受水平。这些风险控制措施可能影响软件或过程的其他部分。

### B.6.2 对软件没有影响的风险控制措施

对软件没有影响的风险控制措施的示例包括程序更改、硬件冗余、备份系统、监控系统、输出确认（下游确认）或供应商检查。

嵌入式生产过程软件通常难以访问，制造商也很少提供细节。常见的示例是嵌入在制造医疗器械的机床内的软件。如果独立确认软件，此类软件的预期用途很难确认。



在这些情况下特别有效的风险控制措施是下游验证软件的输出或软件控制设备的输出。换言之，通过监视软件控制过程的输出是否存在任何和所有潜在有害缺陷，能够直接确定软件是否适合其预期用途。这种方法可以作为替代通过应用生存周期控制方法来推断软件是否适合其预期用途。此方法仅适用于关键操作数量相当少的过程，这些操作能够在每个部件或在统计确定的部件抽样上进行检查。确认工程师宜详细说明替代下游验证的合理性理由，以及用于证明选择抽样验证代替连续验证的任何假设，并测试这些假设。

宜将下游验证形成文件，与任何其他风险控制措施的形成文件的方式相同。特别重要的是，记录确认过程是一种风险控制措施，以便在以后的成本削减措施中就不会消除。此外，还宜记录下游验证结果，因为确认的定义要求为确认“提供客观证据”，而且此验证步骤替代确认的一大部分。随着产品的发展，软件控制过程的预期用途也会发生发展。举个例子，考虑一台机床，最初它只在医疗器械的某个组件上执行一种关键操作。后来，医疗器械设计略有改动，要求软件驱动机床执行两种关键操作。机床的预期用途发生了更改（两种关键安全操作相对于一种关键安全操作），因此，下游验证宜更改为验证两种操作。

下游验证可通过手动操作或其他人工操作完成。示例可能包括对毛刺边缘或机械对准的目视检查，以及对机械公差或电气连续性的手动测量。无论测试性质如何，如果是软件控制过程的下游验证，而且如果被用作该过程的风险控制措施，则宜记录该验证测试。人体测试人员的测试程序宜详细描述，明确规定每个测试参数合格通过和失败的结果范围。测试人员还宜提供书面证据，证明自己已经执行了测试过程输出的程序。

### B.6.3 对软件产生影响的风险控制措施

影响软件的风险控制措施是以下两种措施中的一个：

- 设计更改，或
- 过程控制。

在本文件的范围内，过程控制的选择也就是确认的严格性，意味着选择表A.1至表A.5中定义的工具。

最好是在仅依赖于过程控制之前，宜实施软件之外充分理解的风险控制措施（例如：“下游确认”以及软件设计更改）。但宜采用的过程控制的最小集，特别是在对作为风险控制措施的软件设计更改的正确实施提供信任时。

### B.6.4 风险控制措施的验证与剩余风险的评价

宜验证风险控制措施的实施情况。宜验证过程控制以外的风险控制措施的有效性。在这种情况下，宜评价剩余风险的可接受性。

## 附录 C

### (资料性)

#### 示例

本文件适用于用于质量体系和制造过程自动化部分的软件，包括拟用于监管提交、质量体系、生产和数据处理的数据的生成、测量、评估或管理。其他预期用途可能包括直接或间接从仪器获取数据、操控及控制设备和流程、报告和存储数据等。由于这些活动不同，软件也不相同，可能是包含在可编程逻辑控制器（PLC）或个人计算机（PC）中的软件，也可能是包含在具有多种功能的实验室信息管理系统（LIMS）中的软件。以下是一些预期用途的示例：

- 对产品做出通过/不通过判定的软件；
- 用于质量体系内自定义记录保存的软件；
- 用于提交产品的数据处理和分析的软件；
- 用于向监管机构报告的数据处理和软件；
- 用于监管过程软件的任何软件开发工具或编译器；
- 用于鉴定和验证生命安全相关的关键软件的任何软件工具或附属软件工具；
- 用于质量体系内组件、产品或患者可追溯性的任何软件；
- 用于上述目的的任何“未知来源软件”（即不了解软件的质量和稳健性）。

本附录内所提供的示例表示本文件作者尝试提供医疗产品制造商可能遇到的软件的实际、现实的示例。提供这些示例是体验批判性思维方法并理解软件类型、软件风险和预期用途之间可变性的最佳方式。

请注意以下限定项：

- 此处使用的示例包括本文作者的批判性思维的结果，并且代表可接受的确认工作量和严格程度，这将增加价值并提供软件按预期运行的信任。强烈建议读者从工程角度考虑哪些活动和投入工作量水平是有意义的，并根据医疗器械质量管理体系过程所用软件的关键因素确定所需的严格程度。
- 对确认投入工作量的适当性建立信任的方法不止一种。本文介绍的示例提供了一种基于方法的途径，其基于的是当前的思想和本文作者的经验。
- 强烈反对读者将作者们的工作视为权威和规范。引用示例采用相似格式的原因仅为展示数据之用，包括用于证明批判性思维使用的关键的思维过程。采用这种布局既不预期将其用作确认模板，也不包含实际确认文档所期望要达到的所有深度和细节。
- 所采用的示例假设第 6 章内确定的前提过程真实存在且处于良好工作状态。虽然这些示例不包含对前提过程的广泛引用，但这些过程宜予以采用，以确保软件和所有相关方面（如文档和其他基础设施）受到更改控制。
- 每个示例都从明确定义待控制的过程开始。因此，已经确定了过程以及软件均在范围之内。然后识别并总结批判性思维活动。
- 此处使用的示例旨在提供批判性思维过程中使用的决定以及决定推动因素的信息，但并不一定表示所讨论软件的全面确认。
- 示例中所使用的任何公司名称、团体或个人均为虚构，在此使用仅为了便于讨论。

此处使用的示例通常侧重于将特定系统带入确认状态。虽然为系统建立确认状态非常重要，但在系统的维护阶段保持该确认状态对于确保软件和周围过程的正确运行也至关重要。维护活动需要与初始确认活动相同的控制和批判性思维。

### 示例1：用于设备制造的可编程逻辑控制器（PLC）

#### 背景资料

软管供应公司已与一家大型医疗器械制造商签订合同，为其静脉注射（IV）系统提供软管。该公司已经收到了输液软管的技术规范，包括用于软管专有形状的形成要求。软管供应公司将执行这种特殊管形要求并作为其管段制造过程的一部分。

该软管形成过程是供应商特别关注的，因为软管形成是供应商目前唯一没有使用机器执行的过程。供应商决定开发一具有可编程逻辑控制器的定制设备来执行该任务。根据医疗器械公司的政策要求，宜对该设备及其内包含的PLC的预期用途进行确认。

#### 定义过程

软管供应公司和医疗器械制造商成立了一个团队，定义软管形成过程。会上定义的过程是利用温度和压力在一根塑料管中形成一个形状。包括以下步骤：

- a) 获得材料；
- b) 插入机器；
- c) 通过压力和热使软管变形至合适的直径；
- d) 等待软管冷却；
- e) 从机器取出软管；
- f) 测量软管直径是否合适。

#### 过程风险分析

医疗器械制造商已向软管供应公司通报了风险分析过程产生了以下问题和相关危险。

- 与输液袋缺乏良好的连接会导致泄漏，虽不危险，但可能存在护理人员滑倒的风险。泄漏也可能延误治疗。
- 外观问题可能影响顾客接受度并导致延误治疗。
- 软管成形过程中操作人员可能被灼伤。

在风险降低之前，由于护理人员滑倒、治疗延误和操作人员灼伤的危险，造成产品失效的风险为中等水平。

目前采取了以下风险控制措施：

- 上游操作，如进货检查和生产线清理，确保软管可接受使用；
- 下游验证检查，包括泄漏测试、过程检查和测试配件，以减轻设备错误；
- 安装设置屏蔽罩、独立的温度传感器和冷却液喷雾器，以防止操作人员受伤。

基于此信息，供应商与医疗器械制造商合作得出这样的结论：经此软管形成过程，软管失效的剩余风险较低。

#### 定义软件目的和意图

软管供应公司知道，要确认软件的预期用途，宜首先定义其预期用途。为了就设备的目的达成共识，团队成员向自己提出一系列问题，旨在确定系统目的和意图的简洁且可用的定义。团队成员形成了以下结论。

- 软件控制设备旨在自动化执行已定义工序的第2至第6步骤。该系统拟用于第3生产线上的B设施创建PN 001。该系统将自动插入、成形、移除并测量用于输送常规无害溶液的静脉注射（IV）软管。

#### 确认策划

确认策划的第一步涉及确定可交付成果的严格程度和评审。由于确定剩余过程风险为较低，因此采用了以下方法：

- 文档的严格程度：

- 此项目文档的严谨性为中等，意味着将存在可交付成果组合的情况，并且实施前不会将设计转换为详细的设计规范。

#### ——审查程度：

- 评审和批准将由负责过程开发和实施人员（软管供应公司代表）及独立的质量人员（医疗器械公司代表）进行；
- PLC代码和所有规范/设计将置于正式配置管理下，例如在文档控制系统或配置控制系统中。

#### ——系统定义：

- 将创建过程要求，过程要求包括详细说明设备的功能的系统要求规范，包括设备的预期输入和输出（例如设备的整个功能部件的设计控制元素）；
- 团队将从操作人员的角度创建一本关于系统使用的操作员手册。此外还将创建各种软件需求，而软件需求要包括逻辑功能流程，这也足以涵盖软件的设计。

### 建立对软件的信任和控制

软管供应公司和医疗器械制造商之前都没有使用过这种PLC编程包。软管供应公司没有有助于建立信任，相信软件有能力按要求运行的历史记录。但是，通过评审要求、配置控制以及通过测试协议测试系统功能，可以控制PLC的编程。

### 定义软件与其他系统的边界

PLC包含该设备中唯一的软件。此软件未链接到任何其他系统。

### 软件风险分析

软件可能由于将错误形状的软管放在生产线上而失败，导致泄漏并可能导致护理人员滑倒。软件也能发生故障，导致过热，从而导致操作员灼伤。软件本身不会给产品带来在过程风险分析中任何尚未捕获的新风险。因此，团队确定当前的下游过程宜保持不变，并足以降低与软件失效相关的风险。

### 完成确认计划

鉴于团队成员已对软件及其使用有了更多了解，团队成员宜按照以下步骤完成确认计划。

#### ——实施工具：

- 设备内的一系列可编程参数包括时间、温度和压力。设备内这些参数的预定设置和范围都在软件需求内捕获。因此，软件需求规范足以满足设计目的，无需额外的设计活动或文档。
- 团队将在软件需求及其相关测试之间建立一个可追溯性矩阵，并将进行可追溯性分析，以确保可追溯性是完整的。

#### ——测试工具：

- 软件系统测试将基于操作员手册中的软件需求和程序。
- 若需要，将进行回归测试。

#### ——部署工具：

- 系统操作员和工程师将评审工作说明的清晰性和可用性。
- 使用设备需要操作员认证。
- 在完成确认计划并执行其活动后，团队满意于系统将始终如一地提供所需预定的输出。

### 维护考虑因素

若认为改过程的任何部分发生了更改，或者如果软件的预期用途发生更改，则宜进行分析，以确定当前的任何降低措施会受到影响，或者任何新的风险与更改相关。此分析包括评审与软管成形设备相关的软件风险。

### 工具箱使用

使用了工具箱中以下工具。

——开发—定义阶段：

- 过程要求定义；
- 过程失效风险分析；
- 预期用途；
- 确认策划；
- 软件需求定义；
- 确定制造过程内的风险控制措施。

——开发—实现阶段：

- 软件失效分析；
- 可追溯性分析。

——开发—测试阶段：

- 软件系统测试；
- 回归测试。

——开发—部署阶段：

- 用户程序评审；
- 操作员认证。

征求意见稿

## 示例2：自动焊接系统

戴夫是确认新生产线上所有系统的团队成员之一。他的工作是确认箱盖焊机。他是这个项目投入的项目经理。

### 过程描述

戴夫的团队花了很多时间讨论新生产线的各部分由谁来开发和确认。当戴夫拿到这几部分的时候，上面都已经做了标记，所有材料都接受了检查和认证。这些部分在上游经确认的系统上进行测试。

设置焊机需要四个步骤：

- 打开机器；
- 确认待运行部分中存在条形码；
- 从制造执行系统中调取该部分的程序；
- 根据设备主记录，确认正确的程序版本。

箱盖焊接过程本身有10个步骤：

- a) 开门；
- b) 装载各部分；
- c) 关门；
- d) 启动程序；
- e) 将视觉系统指标置于启动点；
- f) 打开激光器；
- g) 确保运动控制器能够移动部件焊缝；
- h) 关闭激光器；
- i) 开门；
- j) 移除部件。

该过程完成后，部件转移至戴夫不负责的系统。他知道下游活动包括焊缝熔深的破坏性测试、罐体尺寸的高度检查以及密封件的泄漏检查。

### 定义预期用途

为了定义软件的预期用途，戴夫收集了信息。他知道视觉、移动、功率和速度的准确性对于过程中保护操作员安全和实现一致的焊接熔深都很重要。

戴夫首先通过陈述软件的目的和意图定义其预期用途，如下：

- 该软件用于焊接机箱盖，同时保护机器操作员免于直接接触操作中的激光器。包括上述过程描述中的步骤 e) 至 h)。

### 风险分析

戴夫希望消除该过程的人为错误。他知道，激光控制器、伺服机构和视觉是该过程的关键组成部分。软件首先检查门是否处于关闭状态。出于安全原因，如果软件检测不到门已关闭，就不会启动此过程。软件以确认激光器处于关闭状态、然后允许门打开而结束。紧急关停或意外开门会切断激光器的电源。戴夫使用来自该过程的信息和设计风险管理活动，这些活动是焊接过程设计的一部分。他参考了FMEA并重点关注了三个方面：关键部件参数、密封和用户界面。戴夫确定了与此过程相关的多个危险。首先，操作员如果暴露于激光下可能会被灼伤。与产品相关的是，该过程可能会不正确焊接，从而导致产品泄漏并伤害最终用户。戴夫判定此过程风险很高。

### 确认策划

对于这个项目，戴夫查看了工具箱中的定义工具，确定自己需要创建软件需求定义和维护文档。他的软件需求宜包括工具、激光时间和功率调整的配置参数。他还需要定义软件和硬件接口。具体而言，戴夫需要定义视觉系统、激光时间、功率范围的精度要求、移动控制精度要求和门传感器保护措施，包括激光器激活时的硬件门锁接口。

戴夫还确定,他需要进行正式的软件需求评审,这将涉及自动化工程师、制造工程师和质量工程师。

这个系统的软件将是一个采购来的软件包,但戴夫知道,自己的公司需要进行定制修改。他需要为工厂制造执行系统(MES)添加一个接口。

### 风险控制措施

戴夫接下来关注的是风险。他认为焊缝深度和其他关键参数的严重度很低,因为他确信下游泄漏检查和定期破坏性测试足以检查焊缝熔深。同样,泄漏检查将确认密封是可接受的。这在用户界面方面留下了风险,特别是在开门时软件可能启动激光器的风险。戴夫知道有对门的封闭进行检查的软件,但是如果软件无法按预期运行,风险的严重度就会很高,因此,他增加了一个冗余硬件互锁装置以防止门打开时激光器激活。

### 确认任务

接着,戴夫转向确认任务。他选择的工具供应商提供了广泛的编程工具。因此,早期创建的软件需求规范和评审(工具)足以满足设计需要,而无需使用工具箱中的其他设计、开发和配置工具。

戴夫从工具箱的测试部分选择的另一项任务是制定测试计划。测试计划将包括软件环境和预期测试结果的详细信息。测试计划需要由自动化工程师、制造工程师和质量工程师以及戴夫评审和批准。测试报告将包括实际测试结果及将其与预期结果进行的比较,提供通过/不通过的提示,也将包括测试识别并且提供任何失效的问题解决和回归测试的文档。对于这份报告,戴夫需要获得自动化工程师、制造工程师、质量工程师和项目发起人的额外批准。

### 部署

对于焊机的部署,戴夫评审了工具箱中的部署工具并决定需要一个制造操作员程序,而且该程序需要经由自动化工程师、制造工程师和质量工程师的评审。为了确保操作员了解如何操作焊机,戴夫创建了一项包括测试在内的操作员培训和认证程序。他知道MES不允许操作员在未经认证的情况下在系统中调出焊接程序,因此他确信操作员受伤的风险已经成功地减轻。

### 维护

戴夫知道他的公司有一个配置检查工具。因此,在本次确认期间没有进行具体的维护策划。

示例3：自动焊接过程控制系统

本示例中的表C.1至表C.14说明图2所示的处理步骤。

表C.1 示例3——过程要求

		过程	迭代风险分析	确认策划和报告	软件系统
开发	定义	<p>过程要求（见 5.3.2.2）</p> <p>器械公司是家 C 类（参见 GHTF/SG1/N77:2012）医疗器械制造商。器械公司已选择实施一套自动焊接过程控制系统。为确保设备外壳焊接适当，器械公司将使用一种使用参数发布决策过程来隔离产品的方法。器械公司还选择使此过程中的信息以支持其器械历史记录。</p> <p>器械公司已指派了一名新的项目经理来确认该自动焊接过程控制系统。该项目经理认识到，该系统需要符合 GB/T XXXXX/ISO 13485 对软件确认的要求。因此也认识到，所提出的焊接过程控制系统需要确认。</p> <p>为了更好地理解焊接系统确认所涉及的需求和风险，项目经理将定义了如下过程：</p> <ol style="list-style-type: none"> <li>a) 操作员将批号输入系统中，作为该批次的开始。</li> <li>b) 操作员将子组件插入机器夹具中。</li> <li>c) 操作员按下循环启动按钮。夹具通过液压装置机械地移动到配合位置。</li> <li>d) 焊接循环连同固定子组件的固定速度旋转一起开始。</li> <li>e) 红外温度计监测焊接过程中材料的温度。温度以及焊接的每个部件的批号和部件序列号记录在文件中。</li> <li>f) 机器在循环结束时打开夹具。</li> <li>g) 操作员移除焊接部件，并根据序列号将该部件放置在批次托盘的相应位置。</li> <li>h) 操作员重复步骤 b) 至 g)，直至装满批次托盘。</li> <li>i) 操作员点击批次结束按钮。</li> <li>j) 机器操作员界面显示焊接温度超出工艺极限的部件序列号。</li> <li>k) 操作员从批次托盘中丢弃相应的部件号。</li> <li>l) 操作员打印拒收部件清单并向下一个工作站发送批次托盘和报告。</li> <li>m) 操作员通过重复步骤 a)，开始新的批次。</li> </ol> <p>项目经理还认识到关键自动化功能如下：</p> <ul style="list-style-type: none"> <li>——存储批次号；</li> <li>——存储每个序列号的焊接温度；</li> <li>——显示焊接过程中超出过程温度极限的部件的序列号；</li> <li>——打印批次拒绝报告。</li> </ul>			

表C.2 示例3——过程失效风险分析

		过程	迭代风险分析	确认策划和报告	软件系统
开发	定义	<p>过程失效风险分析（见 5.3.2.3）</p> <p>然后项目经理考虑了当前过程中可能出现的问题。项目经理意识到，如果过程故障，放行焊接不当的部件可能会使患者接触非无菌设备。由于焊接过程控制系统错误或操作员失误，可能会发生劣质产品意外放行。</p> <p>然后，项目经理考虑有哪些现成的用以降低风险的风险控制措施。项目经理得知，过程组有一个现成的程序，在下一个工艺步骤验证焊接操作员是否已正确地摒弃了部件。此外，项目经理还了解到，焊接系统是一个商业 OTS 系统。</p>			

表C.3 示例3——软件目的和意图

		过程	迭代风险分析	确认策划和报告	软件系统
开发	定义	<p>软件目的和意图（见 5.3.2.5.2）</p> <p>通过对过程的基本了解，项目经理准备编写焊接过程控制系统的目的和意图。</p> <ul style="list-style-type: none"> <li>——焊接过程控制应用程序对焊接件的通过或失败状态做出闭环质量判定。焊接操作员根据判定结果，手工分检出不合格产品。</li> </ul>			



		项目经理评审了目的和意图，以便在过程内适当捕获软件的边界，并决定修订如下方式修改说明：
--	--	---

表 C.3 示例 3 (续)

		过程	迭代风险分析	确认策划和报告	软件系统
开发	定义	<p>——焊接过程控制应用程序对焊接件的通过或失败状态做出闭环质量保证判定。焊接操作员随后根据判定结果，手动排除参数不合格的情况。焊接站是整个设备过程中确保设备密封完整性的唯一控制点。</p> <p>然后，项目经理考虑需要与焊接系统连接的其他系统（若存在）。他确定该软件是一个单独的应用程序，运行在连接着红外温度设备、操作员界面、打印机和机器 PLC 输入/输出设备的个人计算机上。焊接系统未连接至网络。</p>			

表C.4 示例 3——确认策划

		过程	迭代风险分析	确认策划和报告	软件系统
开发	定义	<p>确认策划（见 5.3.2.4）</p> <p>既然项目经理了解了过程并已确定了新系统的预期用途，因此，项目经理可以在高级别上制定确认的计划了。</p> <p>之前，项目经理确定焊接过程存在很高的剩余风险，原因是焊接过程的实施是个不可验证的过程。因此，项目经理确定需要对确认工作进行广泛的评审。项目经理决定关键的批准角色宜由过程工程与质量工程部以及操作过程培训师担任。此外，最终产品接收经理宜批准这些要求。</p> <p>项目经理决定开始编写确认计划，因为质量体系要求对于高风险系统确认计划需在任何其他确认可交付成果或项目可交付成果批准之前批准。</p>			

表C.5 示例 3——软件使用要求与软件需求

		过程	迭代风险分析	确认策划和报告	软件系统
开发	定义	<p>软件使用要求与软件需求（见 5.3.2.5）</p> <p>项目经理认为有必要在此确认工作中提供高级别的详细信息或形式，而且了解定义详细过程和软件需求的重要性。项目经理开始编写软件需求。项目经理决定软件宜在温度确认和不合格判定过程包括冗余量。项目经理还要求系统能够在生产线清理活动发生之前的任何时间重新打印不合格拒绝报告。</p> <p>由于该系统支持参数值，因此项目经理还列出了包括信息安全要求以及系统访问级别可更改数据值的详细列表。</p>			

表C.6 示例 3——软件失效风险分析

		过程	迭代风险分析	确认策划和报告	软件系统
开发	定义	<p>软件失效风险分析（见 5.3.3.2）</p> <p>项目经理现在需要决定采用何种方法建立对焊接系统的完全信任。</p> <p>项目经理指出，焊接机的设计需要业内常用的商用现货（COTS）系统。项目经理发现，制造商已经快速识别并公布了该产品以往的问题或争端。</p> <p>虽然项目经理早已确定焊接过程风险很高，但是项目经理仍然希望正式分析软件失效的风险。为了证实这种直觉，项目经理评审了来自公司风险模型的一些问题。</p> <p>a) 如果软件出现故障，是否存在对产品安全的潜在风险？是</p> <ol style="list-style-type: none"> <li>1) 如何出现？系统基于默认温度限制接收次品部件。电源故障之后限制重置为默认设置。</li> <li>2) 如何控制该风险？要求操作员在每批次运行开始和结束时验证限值。</li> </ol> <p>b) 如果用户出错，是否存在对产品质量的潜在风险（安全风险除外）？是</p> <ol style="list-style-type: none"> <li>1) 如何出现？在手动模式下，若两个部件传感器均触发 3 秒钟，则焊接激光器可能会启动。</li> <li>2) 如何控制该风险？将默认配置更改为仅在自动模式下触发。</li> </ol>			

表C.7 示例 3——确认策划

		过程	迭代风险分析	确认策划和报告	软件系统
开发	实现、测试与部署	<p>确认策划（见 5.3.3.3）</p> <p>了解了软件需求，项目经理获得了足够信息来完成确认。项目经理已决定了实现方法，并分析了软件风险。此时项目经理退后一步，根据了解到的有关该系统的一切信息，提出这个问题：“什么样的确认活动才能让我真正相信焊接系统适合其预期用途。”</p>			

		项目经理考虑第三方将如何开发系统，并担心开发人员是否正确转换了客户化报表的要求。由于该系统将依赖于各种数据字段，因此，项目经理在代码评审中添加了验证步骤活动，以确认开发人员工作的正确性。
--	--	---

表C.8 示例3——软件实现

		过程	迭代风险分析	确认策划和报告	软件系统
开发	实现、测试与部署	软件实现（设计、开发、构建与测试）（见 5.3.3.4） 购买而不是内部开发软件的决定是基于商用现货（COTS）系统的可用性。然而，项目经理仍需向器械公司的质量部门证明该焊接控制软件是在有效软件开发生存周期（SDLC）下开发的，因为其预期使用风险被归类为高风险。 在与 COTS 供应商讨论过此问题后，项目经理得知供应商的 SDLC 过程最近接受了一家独立审核公司的审核。随后项目经理联系了那家独立的审核公司并购买了 COTS 供应商 SDLC 审核报告的副本。最终结果是质量部门采信了 COTS 供应商开发此软件是在有效生存周期模型下进行的。			

表C.9 示例3——确认报告

		过程	迭代风险分析	确认策划和报告	软件系统
开发	实现、测试与部署	确认报告（见 5.3.3.5） 项目经理完成确认报告并获得批准。			

表C.10 示例——软件发布

		过程	迭代风险分析	确认策划和报告	软件系统
开发	实现、测试与部署	软件发布（见 5.3.3.6） 项目经理验证置于正式配置管理系统下的软件与确认报告中引用的软件相匹配。			

表C.11 示例3——更改分析

		过程	迭代风险分析	确认策划和报告	软件系统
维护		更改分析（见 5.4） 按照确认计划项目经理验证了公司有一个正式的变更控制过程，用于管理任何确认后对焊接系统的更改。			

表C.12 示例3——维护确认策划

		过程	迭代风险分析	确认策划和报告	软件系统
维护		维护确认策划（见 5.4.2） 项目经理提前考虑哪些活动为确保系统持续满足其预期用途是适合的。考虑到系统风险较高，项目经理决定宜每季度进行一次校准和认证，确保实际温度测量结果与批次报告中打印的温度值对比精准无误。项目经理在确认计划中包含一个用于记录此结论的章节，并要求制定和实施一个校准和认证程序，以确保在系统投入生产后按季度进行评审。			

表C.13 示例3——软件维护

		过程	迭代风险分析	确认策划和报告	软件系统
维护		软件维护（见 5.4.6） 按照确认计划项目经理验证了公司有一个定期评审过程，能够确保焊接系统及过程不会偏离其预期用途。			

表C.14 示例3——软件退役

		过程	迭代风险分析	确认策划和报告	软件系统
退役		软件退役（见 5.5） 按照确认计划项目经理验证了公司有正式的软件退役过程可管理焊接系统的退役问题。			

## 工具箱选择

设计、开发与配置工具：

- 定义过程要求；
- 正式的软件需求评审；
- 确定制造过程和业务过程内的风险控制措施；
- 过程开发评审；
- 可追溯性矩阵（要求规范中固有的）。

测试工具：

- 测试策划；
- 软件系统测试；
- 软件配置控制。

部署工具：

- 用户程序评审；
- 应用的内部培训；
- 安装鉴定；
- 过程确认。

征求意见稿

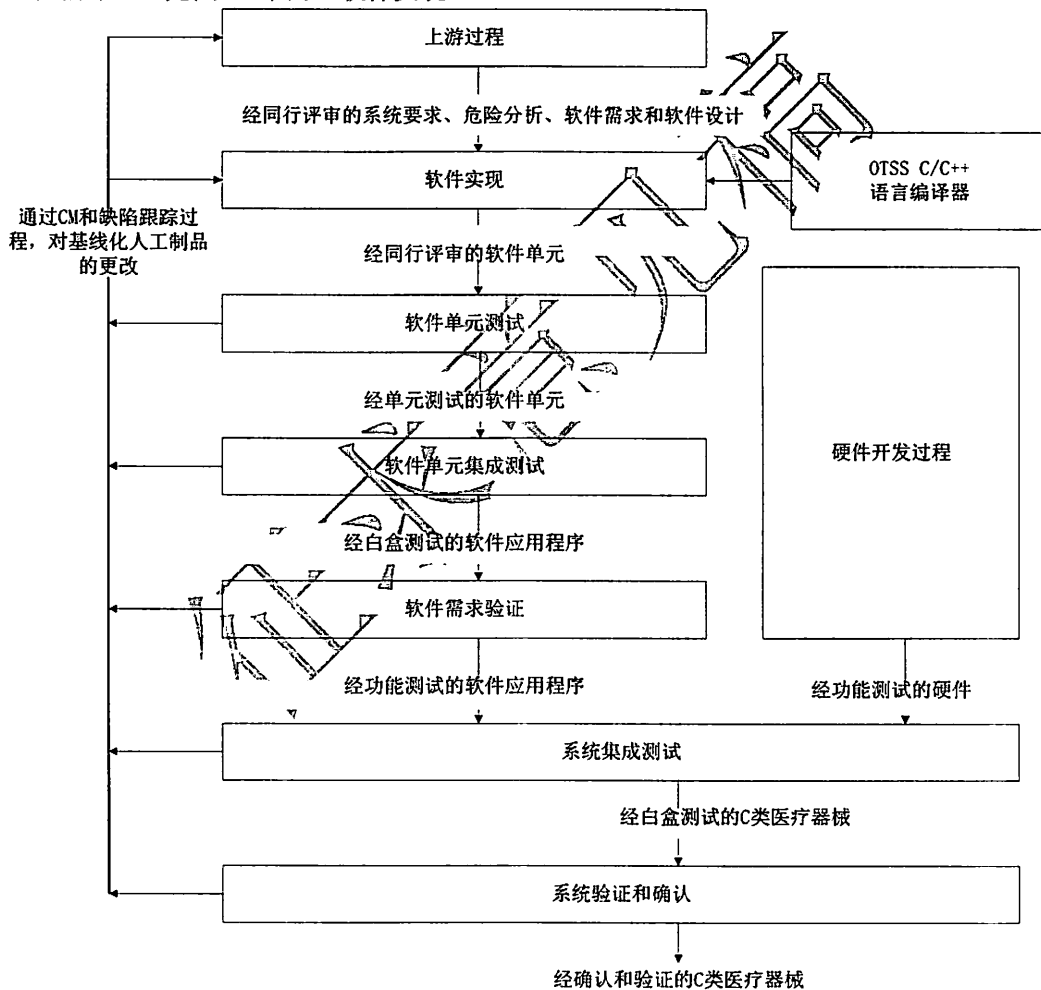
### 示例 4: C/C++语言编译器

#### 背景资料

某C类医疗器械公司需要为一种嵌入式系统确认其现用软件C/C++语言编译器。现已确定，由于此编辑器生成的医疗器械产品软件（软件源代码和可执行软件）已置于医疗器械设计记录中，因此编译器受规范限制。

#### 质量体系过程描述

两个质量体系过程与本案例研究相关。第一个是C类医疗器械软件实现的完整质量体系过程（见图C.1）。第二个是实现软件设计并满足所有软件需求的可执行软件单元的开发过程。软件单元包括OTSS C/C++语言编译器（见图C.1中的“软件实现”）。



图C.1 C类医疗器械的软件实现

#### 上游过程



用于软件实现的过程的上游是用于开发系统级文档（例如要需求、设计、危险分析等），这些文档描述了待开发的医疗器械的特征。然后通过软件需求开发、软件设计和其他软件文件或计划的过程来描述软件中已实现的系统部分。在软件开发的同时，还执行附加过程以开发医疗器械硬件。

### 软件实现过程

使用的正式软件语言为C/C++软件语言。OTSS C/C++语言编译器用于将高级软件语句编译为可执行的机器代码。软件实现过程的输出是基线化的软件单元，由其他技术成员进行同行评审，以确保其完整性和正确性。对于软件单元同行评审，软件单元宜在最高编译器级别无错误编写，任何编译器警告宜在同行评审中予以解释。

### 下游测试过程

软件单元在以下几种测试过程中进行测试或验证：

- 软件单元测试。对每个软件单元的逻辑正确性和边界条件进行测试。这种测试可能发生在开发系统或目标系统（医疗器械硬件）上。在确定某一代码同行评审足以检测单元逻辑错误时，简单的软件单元可以略过该测试；
- 软件单元集成测试。对软件单元进行集成和测试，以确保软件设计正确实现，并且测试与设计相关的边界条件。此测试在目标系统上进行；
- 软件需求验证。根据完整的软件需求集验证完整的软件应用程序。此验证在目标系统上进行；
- 系统集成测试。对医疗器械中的软件和硬件进行测试，以确保系统设计正确实现，并且测试与系统设计有关的边界条件；
- 系统验证和确认。医疗器械在系统要求级别进行验证，此外，对其预期用途进行确认。

### 过程失效风险分析

该项目遵循公司的过程风险评估程序。C类医疗器械软件实现的整个质量体系过程（包括图C.1描述的所有过程）本身具有很高的高风险，原因是由其生成的软件在C类医疗器械内运行。

作为软件实现过程的一部分，OTSS C/C++语言编译器基于以下两个因素被评估为低风险：

- 编译器不会直接导致患者、操作员或旁观者严重损伤或死亡；
- 对工具的输出（软件源代码和可执行软件）实施下游验证（例如软件单元测试、软件单元集成测试、软件需求验证、系统集成测试、系统验证和确认等）。

### 定义预期用途

在上述软件实现过程中OTSS C/C++语言编译器的目的和意图是编写嵌入式系统源代码并执行编译过程，以生成C类医疗器械的可执行软件。

### 软件使用要求

- a) 该工具宜交叉编译C和C++代码，以使用所选供应商操作系统在精简指令集计算机（RISC）处理器上运行。
- b) 编译器宜具备一个源代码调试器。
- c) 编译器宜与美国国家标准协会（ANSI）的C和C++相兼容。
- d) 编译器宜与各种已批准的行业标准集成开发环境集成。
- e) 供应商宜发布可搜索的已知错误（BUG）列表。此清单宜作为参考，以便根据需要进行查阅。
- f) 要求供应商在受监管行业内拥有庞大的用户群。

### 软件失效风险分析

OTSS C/C++语言编译器的风险分析表明，如果出现错误可能发生以下事件：

- 风险1. 供应商未能提供适当的业务过程、开发方法和支持功能。
  - 降低措施1. 见下文的“供应商选择过程”部分。
- 风险2. 编译器生成错误的可执行语句。

- 降低措施2. 见下文的“确认计划”部分。
- 风险 3. 未运用最严格错误检查级别的用户错误地使用了编译器。
- 降低措施3. 改进培训、程序和工作指示。

### 供应商选择过程

项目在选择和批准供应商方面遵循公司的质量体系程序，这些信息记录在项目的设计记录中。该程序包括评审供应商的SDLC方针、程序、任务和活动的现场评估。对供应商提供的OTSS C/C++语言编译器功能是否满足上述定义的软件使用要求进行验证。

### 确认计划

为OTSS C/C++语言编译器选择了下游确认的方法。供应商选择过程已确定该供应商满足所有形成文件的软件使用要求。编译器在供应商处有大量的运行时间，并且在项目调试和测试期间也将有大量的运行时间。编译器的输出在下游过程中接受以下动态测试：

- 软件单元测试；
- 软件单元集成测试；
- 软件需求验证测试；
- 系统集成测试；
- 系统验证和确认。

### 确认报告

确认报告内容如下：

- OTSS 描述；
- 软件使用要求；
  - 硬件要求；
  - 软件需求；
  - 补丁。
- 风险评估和危险分析；
- 供应商选择；
- 安装活动；
- 确认；
  - 软件使用要求测试用例和结果。
- 已知错误列表；
- 配置控制；
  - 培训；
  - 安装位置；
  - 维护；
  - 退役过程。

### 工具箱选择

- 定义阶段：
  - 预期用途；
  - 确认策划；
  - 风险管理计划（风险评估）。
- 实现阶段：
  - 风险控制措施；
  - 供应商审核。
- 部署阶段：
  - 安装鉴定；

- 应用程序的内部培训；
  - 最终验收测试。
- 维护阶段：
- 维护策划；
  - 已知问题分析。

征求意见稿

## 示例5：自动软件测试系统

### 背景资料

本示例中制造商是一家C类医疗器械制造商。该制造商生产的医疗器械由软件控制。软件在架构上分为两大组件：操作员控制台和实时嵌入式控制软件。操作员控制台是系统的主要人机界面。实时嵌入式控制软件是执行机电控制、数据采集、定时等的软件。操作员控制台软件（安装在一台运行行业标准操作系统和数据库的PC/个人计算机中）和实时嵌入式软件（驻留在一块板载嵌入式CPU卡中）采用标准传输控制协议/互联网协议（TCP / IP）硬件和协议接口进行连接。

该项目软件经理已经认为通过引入软件自动化测试改进软件开发和测试过程将是很有价值的。软件经理决定最初仅对操作员控制台软件进行自动化软件测试。自动化软件测试将在集成测试点和软件系统测试点进行。

### 确定软件处于监管状态

因为自动化测试软件将用于执行制造商软件开发程序所需的测试，并且因为它将在集成测试点和系统测试点为所需的回归测试提供证据，所以自动化测试软件被确定为开发过程的自动化部分，并因此确定其遵循GB/T XXXXX/ISO 13485的确认要求。

### 定义过程

为了更好地理解引入操作员控制台的自动化软件测试所涉及的需求和风险，软件经理在软件开发过程中定义了自动测试软件的如下用途。

在器械软件开发期间，安排了在不同时间将各种模块集成到系统软件中。此外，由于缺陷纠正和要求修改，已集成到系统中的模块将会发生更改。计划将自动测试系统用于集成系统软件的回归测试以及系统中特定模块的最终测试。软件项目计划要求每周进行两到三次模块的集成或更新。自动测试将在每个集成点上运行，确保新功能正常工作，并且先前运行的功能不会受到新增代码或特定构建中代码更改的不利影响。对于最终发布给确认和最终客户的候选构建，自动化测试将在软件系统测试级别运行。如果在开发的最后阶段发现缺陷需要进行纠正，也将使用自动测试，以提供一定程度的用于补充手动测试的回归测试。

### 风险分析

如果自动测试软件使用不当，现在软件经理将通过分析过程来确定任何潜在影响。

软件经理首先需要评价的是自动测试过程的失效、自动测试软件的某个失效或任何使用自动测试软件的人所犯的的错误，是否最终会导致医疗器械的缺陷，从而可能对患者、操作人员、旁观者、服务人员或环境造成伤害。

- 软件经理最担心的是，自动化软件测试系统会给出错误的指示，即被测试的操作员控制台软件在实际存在缺陷的情况下工作正常。
- 如果未检测到的缺陷位于软件的关键区域，则其可能导致医疗器械出现故障，从而造成伤害情境。
- 软件经理意识到，这种风险可能源于自动测试软件管理不当、使用或其本身的缺陷。
- 软件经理断定，对于自动化软件测试系统使用时机以及使用目的设置边界条件极其重要，以确保软件开发和测试团队不过度依赖系统。
- 参与自动测试软件配置、编程和操作的人员需要接受岗位培训。
- 软件经理认为，如果这些因素得到控制，潜在的相关风险将降低到可接受水平。

### 定义软件的预期用途

在分析了自动测试软件的潜在用途和相关风险后，软件经理准备制定自动软件测试系统的目的和意图。内容如下：

- 自动测试系统在开发过程中将用于在集成测试点测试软件的构建。



- 自动测试系统在软件系统测试点将用于测试确认和发布候选的构建。
- 自动测试系统将对系统进行回归测试，以确保新引入的软件或软件更改不会对工作过程产生不利影响。
- 自动测试系统的基本作用是为将要进行的手动测试提供补充回归测试。
- 对于复杂性较低的可预测的工作过程，自动测试系统可用作软件正确性的最终决定因素，因为经过确认，前提是已验证特定协议与相应的手动测试一致。
- 自动测试系统将运行于软件系统或医疗器械整体提供保障（风险降低措施）的软件。

### 确认策划

软件经理已清楚地了解要自动化的过程、自动测试系统的特定预期用途以及所涉及的潜在风险。软件经理已经确定需要对软件的使用进行控制，如果按照软件经理规定的适当控制来使用软件，自动测试系统与其使用相关的风险将处于可接受水平。

- 在这种情况下，软件经理确定若自动化软件测试系统使用适当，几乎不存在导致医疗器械缺陷的风险。软件经理定义了使用适当，意指软件开发和测试团队不会过度依赖和使用该系统来确定软件的正确性。鉴于认为风险程度较低，软件经理已经确定系统的确认要求中软件测试系统的测试工作量和严格程度都会偏低。

### 确认文档：确认报告方法

软件经理选择的方法是为自动化软件测试系统编制一份软件确认报告，其中要包括系统中获得必要信任水平相关的所有活动的摘要。

### 批判性思维

软件经理确定达到必要信任水平最好的方式，系统使用适当，并且不会造成医疗器械出现严重缺陷。他认为，在系统中达到必要的信任水平的最重要因素如下：

- 严格遵守适当的预期用途
  - 确保参与软件开发和测试的所有人员清楚地了解系统的边界条件和适当预期用途。
  - 文档：确认报告中包括描述特定的预期用途以及通过项目软件开发计划传达此信息方式的章节。
- 尽职调查
  - 从信誉良好的供应商处购买行业标准自动化软件测试系统，该供应商的测试系统正用于关键性级别相同或更关键的应用程序。
  - 与供应商一起评审系统的预期用途，以确定预期用途是否适宜。
  - 获取有关供应商在软件发布至商业市场前进行软件确认的信息。获得供应商质量部门的声明，证实该商业化软件已通过供应商确认。声明将建立供应商对自动化软件测试系统进行了充分测试的信任，并为软件经理和软件开发测试团队将执行的其他活动奠定初步基础。
  - 与供应商建立关系，以确保软件经理和软件开发测试团队了解将使用的测试软件版本的已知问题和缺陷。
  - 了解供应商未来的软件更新计划，以确保能够预期新版本软件的迁移计划和再确认活动。
  - 文档：确认报告中包括描述供应商尽职调查活动结果的章节，其中包括供应商对自动化软件测试系统的确认、有关供应商缺陷（错误）列表访问方法以及关于新版软件预期迁移计划的信息。
- 安装测试
  - 证实软件所处计算环境符合供应商的规范。
  - 确定初始高级测试协议，以确保软件已正确安装。
  - 文档：确认报告中包括描述安装确认活动结果的章节。
- 风险管理
  - 确保系统仅按照软件经理所定义的软件目的和意图来使用。

- 在将使用自动测试系统的项目的软件开发计划中包括特定可允许的边界条件。
  - 进行分析以确定系统测试的确切覆盖区域，以确保手动测试能够解决自动化软件测试系统未覆盖的区域。
  - 文档：确认报告中包括描述初始风险分析确定的风险的章节，并说明每种风险的减轻方式。
- 软件使用要求
- 开发预期使用的自动测试系统功能列表。该列表由软件经理和软件开发与测试团队开发，称为“软件使用要求”，代表将使用的功能。
  - 文档：确认报告包括“软件使用要求”列表章节，并描述每一个软件使用要求。
- 自动测试系统的确认
- 使用“软件使用要求”列表以确定必要的信任水平。信任水平可以通过采用三个初始自动化测试脚本或协议并对照手动运行相同的协议进行并行测试来确定。这三个初始测试脚本或协议执行团队将使用的所有功能。
  - 文档：确认报告包括总结并行测试的结果的章节，并包括以表明结果是等效的测试证据。
- 培训
- 为所有系统用户建立培训计划，以确保所有系统用户完全了解系统的使用方法，并且具有使用资格。软件经理认为，培训是确保安全有效地使用自动化软件测试系统所需的最重要因素之一。
  - 文档：确认报告中包括描述系统用户所需的必要培训的章节。
- 单个自动化测试协议的确认
- 如果自动测试系统将用于测试旨在降低系统、软件或硬件风险和危险的软件，宜确保使用自动测试和手动测试的并行测试验证每个协议。
  - 如果自动测试系统将用于低复杂度、可预测工作过程的最终测试，宜确保使用自动测试和手动测试的并行测试验证每个协议。
  - 文档：确保医疗器械的软件确认记录包括符合分类的测试脚本或协议的并行测试的证据。
- 配置管理
- 确保仅安装并使用了适宜的且经确认版本的自动测试软件。
  - 当从供应商可获得自动测试软件的新颁布时，宜控制新版本或更改的实施，以确保在适当时间引入新版本或更改。
  - 确保在每个更新点都考虑了自动测试系统的再确认，并确保系统的每次再确认得以执行并形成文件。
  - 文档：确认报告包括描述系统的配置管理计划的章节。

## 确认报告

作为建立信任活动的结果，软件经理提交确认报告以供最终评审和批准。此报告传达了确定将进行的增值活动的思维过程，以便软件经理可以得出结论，使用自动化软件测试系统可能导致正在开发的相关医疗器械不经意间存在缺陷的情境。报告还包含表明所有确定为重要的活动都已按计划执行的证据。

确认报告包含以下内容：

- 定义过程；
- 风险分析；
- 风险管理；
- 预期用途；
- 供应商尽职调查；
- 培训；
- 安装测试；
- 自动测试系统的预期用途确认；
- 维护、再确认与配置管理。

### 确认报告的评审和批准

软件经理将确认报告发送给项目经理、项目软件质量保证经理和软件测试经理以供评审和批准。

所有评审人员一致认为软件经理已清楚地考虑了系统的预期用途，并理解系统使用中涉及的所有相关风险。审核人员认为，为达到允许系统使用所需的系统信任水平有必要完成的所有活动已全部完成。评审人员批准该计划。认定系统已通过确认并投入使用。

征求意见稿

## 示例6: 简单的电子表格

### 背景资料

ZYX公司的实验室分析员厌倦了分析每个产品都要从文档控制系统中提取不同的规格表, 然后手动计算需要与规格进行比较的角度值。实验室中的仪器用于接收检查。该仪器测量三个坐标位置, 分析员使用这些坐标位置计算出与规格进行比较的角度。实验室最近遇到过三次分析员错误计算角度的情况(分析员说原因是“键盘操作失误”), 分析员希望防止这种错误再次发生。他们决定创建一个电子表格来执行角度计算, 并将他们分析的所有50种产品的规格合并于该电子表格中。他们将输入仪器测量的三个坐标对, 从下拉菜单中选择产品名称并且获得通过/不通过的结果。分析员还考虑了仪器的一个接口, 将坐标直接传递给电子表格, 但由于接口成本较高, 该增强功能要延迟到明年。

### 定义过程

当前过程包含以下步骤。

- a) 让仪器测量零件。
- b) 记录下三个坐标对。
- c) 计算角度。
- d) 从文件控制系统中提取部件的规格。
- e) 对比角度值和规格, 确定是否合格。
- f) 将合格单或不合格单放在部件上, 并将其发送到产品零部件库存中。

新过程将包含以下步骤:

- a) 从文件控制系统获取电子表格。
- b) 让仪器测量零件。
- c) 在电子表格中输入三个坐标对。
- d) 对照仪器值目视检查输入的坐标对。
- e) 在电子表格中选择部件编号。
- f) 在电子表格中选择“计算结果”。
- g) 目视检查选择的部件编号是否正确。
- h) 根据结果, 将合格单或不合格单放在部件上, 并将其发送到产品零部件库存中。

### 定义预期用途

分析员将电子表格的目的和意图定义为: 电子表格将获取三个输入的坐标对、计算出角度, 然后将该角度与所选产品的产品规格进行比较, 报告通过/不通过结果。

### 风险分析

分析人员集思广益列出了与电子表格有关可能的危险。他们认为不正确的结果可能意味着不符合规格要求的部件会用于生产。这样的缺陷部件, 要将其提供给医疗器械的最终用户, 至少将出现其他两个下游失效, 即使没有出现, 对最终用户造成伤害的风险仍然很轻微。因此, 生产不符合规格产品的风险很低。但是, 制造成本增加的风险较大, 因为如果生产中使用了不合格的部件, 问题在进行第一次组件装配检查前都不会被发现。结果不得不将整个组件报废。此外, 如果收到错误的失败结果, 那么可能会丢弃优质部件, 这会再次增加报废成本。因此, 将增加电子表格设计、程序控制、文件评审和测试等形式的严格性, 以解决业务上所关心的问题。

### 确认策划

由于生产不合格产品的风险较低, 因此, 确认工作的工作量比较低。分析员决定将电子表格的要求和确认计划合并到同一文件中。分析人员还决定将设计文档与高级别测试计划合并在一起。分析人员计划由整个分析师团队(四人)以及一位质量保证代表对此类文件进行评审。此外, 分析员还计划咨询技术专家, 开发一组有代表性的测试数据, 为按预期运行计算建立信任。技术专家也将批准此文件。

## 风险控制措施

分析人员查看电子表格中可能引入错误并导致错误结果的每个项目。分析人员针对每个项目确定如何降低风险（见表C.15）

表C.15 示例 6——风险与降低措施

风险	降低措施
有可能输入不正确的值。	通过程序控制，证实对照仪器输入的每个数值对。新过程中增加了步骤 d) 以完成这一工作。
计算有可能不正确。	证实公式正确并且按预期提供结果准确。
有可能选择错误的产品。	通过程序控制证实部件编号。新过程中增加了步骤 g) 以完成这一工作。
用于显示结果的宏有可能不正确。	证实宏正确并且按预期运行。
电子表格中的规格要求有可能不正确。	根据 50 个产品的规格表证实电子表格的规格要求。增加规格表更改过程，以便在规格变更时要求更新电子表格。（这种情况未发生但有可能出现。）
计算公式或宏在确认后有可能被更改。	带有配置控制并经确认的电子表格将被放入文件控制系统，并在每次需要时再次获取。对于所有非数据输入单元，配置控件将包括密码保护和锁定单元。

## 确认任务

了解所用的公式，开发人员在电子表格宏的开发方面经验丰富。确认包括以下项目：

- 计算；
- 宏；
- 单元锁定功能（锁定单元不得更改）；
- 数据输入检查（允许范围内的值、适当的产品选择、信息性错误消息）。

因为电子表格一次只能生成一个结果，所以不需要进行压力或性能测试。将为所有测试创建一个测试计划及报告。此报告还将发布电子表格投入使用，并且将在公司文件控制系统中确认该电子表格的控制状况。

## 部署

在部署新系统之前，测试已完成，生产运营商已通过新视觉系统操作的认证。

## 工具箱中的工具

- 需求定义（记录在确认计划中）
- 过程失效与风险分析（记录在确认计划中）
- 预期用途（记录在确认计划中）
- 确认策划
- 测试策划
- 操作员认证
- 维护策划（需要回归分析）

## 维护

每次更改产品规格或添加新产品时，都需要对电子表格进行维护。维护测试计划将与完整确认测试用例的代表性子集一起制定，以确保新项目不破坏该电子表格。维护计划需要进行回归分析，以确定是否需要在测试用例集内额外添加专门针对所做更改的测试用例。该计划还将描述电子表格的更新方式（例如解锁单元格、更改、重新锁定）。



## 示例 7: (并非) 简单的电子表格

### 软件说明

软件开发团队使用Microsoft Excel<sup>1)</sup> 电子表格作为开发辅助工具。电子表格将记录C类或D类器械中使用的器械消息译文。该器械的初始版本采用美式英语编写。后续版本将支持七种语言。电子表格包括七列。最左边一列是器械中每条消息的英语器械消息。其余每列代表将支持的一种国际语言，一列中的每一行代表该行最左侧一列中特定英语消息从英语到国际语言的译文。

### 预期用途

电子表格满足了以下短时需求:

- 直观地组织英语消息及其译文,
- 创建可以发送给当地代表的电子表格, 以便将翻译的消息直接收集到电子表格中或者以手写形式收集在电子表格的硬拷贝上,
- 为消息译文提供短时的数据存储工具。

译文收集完毕并翻译成器械软件后, 就不再需要保留或维护电子表格。

该电子表格中没有计算单元或宏。

### 确定软件是否处在范围之内

Excel仅用于格式化信息, 以便传递、收集器械消息的外文翻译。乍一看, 电子表格似乎是Excel的一个简单应用, 以致于人们很容易认为此电子表格不需要确认。

5.2中提出了以下问题: “软件失效或潜在缺陷是否会对医疗器械的安全或医疗器械的质量产生不利影响?”

这个问题的答案显然是肯定的。如果软件或电子表格失效致使存储在此的消息译文遭到破坏, 则该失效可能会影响器械的安全。虽然团队认为, 这种“简单的应用”失效发生的可能性很低, 但这种可能性仍然在GB/T XXXXX/ISO 13485确认要求的范围内。

### 风险评估

如果未正确翻译器械消息, 则可能导致用户混淆或消息被误解。因此, 存在对使用该器械的患者造成间接伤害的可能性。软件失效可检测的, 而且在器械开发和确认过程中存在大量通过交叉检查的机会以检测并纠正任何软件失效。

可能对器械软件产生不良影响的预期失效模式如下:

- 待翻译的英文原始消息的损坏, 由于输入文件的缺失、个别消息的缺失、消息顺序错误进而导致上下文的缺失导致, 或个别消息的损坏, 由于随机缺失、字符的替换或换位导致;
- 从地区代表处准备和收集的个别消息译文的损坏。损坏可能是由于输入文件的缺失、个别消息缺失、消息顺序错误, 进而导致上下文的缺失导致, 或者个别消息的损坏, 由于随机缺失、字符的替换或换位导致。如果没有在Excel中正确安装所需的非英语字体, 则任何语言的字体都有可能损坏;
- 集合结果的电子表格的损坏, 该电子表格显示每个译文结果的累积。损坏可能是由于输入文件缺失、个别消息缺失、消息排序错误, 进而导致上下文的缺失导致, 或者个别消息的损坏, 由于随机缺失、字符的替换或换位导致。除了电子表格行的排序错误外, 还可能发生列的排序错误。如果各列不以内置字体和字符集显示消息译文, 则软件工程师在将消息转换成代码时会误解这些列。

### 确认策划

<sup>1)</sup> Microsoft Excel 是一个合适的商业可获得产品的实例。提供本信息是为了方便本文件使用者, 并不构成 ISO 对该产品的认可。

软件开发工程师认识到如果新器械的消息有误，患者可能面临风险。软件失效的严重度可能很高。需要采取一些措施建立对电子表格中消息译文正确的信任。

但是，Excel仅用于组织消息。似乎不太可能通过对Excel进行大量测试而发现会导致消息损坏的任何缺陷。在考虑这个问题时，工程师们抱怨说，人为错误比简单地应用Excel而导致错误的的可能性高得多。

在考虑人为错误时，工程师们意识到没有明确定义过程以收集译文或验证过程中无人错误。

工程师们创建了一个用于收集并验证消息译文的书面程序。然后考虑了可能存在导致其过程崩溃的风险，软件（即Excel电子表格）失效如何导致这种崩溃，最后采取哪些措施确认该过程包括电子表格。

### 风险控制措施

在更好地定义翻译收集过程之后，工程师们确定了风险控制措施，以防止该过程在消息译文嵌入中嵌入错误。

保护翻译收集过程的风险控制措施也将防止软件失效以满足其预期用途。

- 如果来自地区代表处，译文宜以纸质（硬拷贝）或电子格式提供，并附带相应的硬拷贝。如果地区代表处提供电子版本，则该电子表格内的数据在转移到主翻译电子表格时将对照硬拷贝进行验证（并形成文件）。该验证将避免电子表格在传输期间损坏或在译文提供电脑与译文接收电脑间的字体能力差异而导致对结果的任何误解。
- 将所有译文收集完毕并放入主电子表格后，宜将电子表格硬拷贝发送到每个地区代表处以供评审批准。地区批准将避免电子表格在传输期间损坏或在译文提供电脑与译文接收电脑间的字体能力差异导致对结果的任何误解。
- 一旦主电子表格被所有地区代表处、开发审批人员和质量保证审批人员接受，主电子表格的硬拷贝宜作为器械软件开发过程的输入。此外，主电子表格的硬拷贝宜作为器械软件中译文验证测试的任何预期结果的来源。

### 确认任务

除了那些风险控制措施之外，还宜完成其他验证和确认任务，以确保软件充分满足其短时预期用途。其他确认任务包括：

- 对于从地区代表处收集的每条译文，均宜根据单个译文电子表格的硬拷贝逐行验证更新后主电子表格的硬拷贝。对照硬拷贝验证硬拷贝是强制性地，以排除由计算机平台或打印机之间的字体差异引起的任何错误翻译。
- 宜详细记录版本控制过程。此过程宜具体说明以下内容：
  - 消息要求（即英语）的更改，如开发过程中器械功能的更改；
  - 主文件中的更改，如所提供的译文以及地区代表处评审并修改的最新主电子表格的更改。
- 虽然电子表格非常简单，但一些非常真实的版本控制风险与其使用有关。
- 电子表格的配置宜包括电子表格本身的版本号、所用 Excel 的版本、计算机平台配置以及用于创建电子表格硬拷贝的打印机的打印机配置。完整配置很重要，因为安装不同的 Windows 或 Office 以及打印机固件版本都可能存在字体差异。确保译文不会意外更改的唯一方法是在使用电子表格时采用相同配置。
- 需要控制电子表格的配置（即操作环境和版本控制），以防止混乱、不协调的更改。指派一个人负责决定配置更改的时间以及记录更改历史的时间。
- 每个电子表格的版本均宜在其硬拷贝版本中可见。
- 器械软件中的译文表宜显示用作消息译文软件输入的硬拷贝主电子表格的版本信息。
- 单个翻译确认任务宜包括以下内容：
  - 英文消息宜通过逐行予以验证，通过对比电子表格的主版本和译文版本的方式。这种比较可防止电子表格文件在传输到地区代表处以及由区域办事处返回文件时可能发生的任何损坏（例如毁坏或丢失的消息）。
  - 译文插入主电子表格（手动或使用Excel的剪切/粘贴功能插入）后，宜对照翻译电子表格的硬拷贝验证修订后的主电子表格硬拷贝的逐行对比输出。

- 对器械软件进行消息实施测试时，测试程序宜采用最新版本的主电子表格硬拷贝（并宜参照版本号）以便已实施消息与预期消息的比较。

宜记录并收集所有确认任务的情况，作为过程和Excel电子表格已经确认的客观证据。

这种确认方法可以100%验证软件输入对应的输出。没有计划对电子表格进行进一步的测试。尽管缺少传统测试，工程师们仍对其过程充满信任，并相信自己的确认原理是一项有价值的实践。工程师断定，软件的任何失效都会检测出来，而且他们具有恢复路径，可以利用在该过程的适当节点收集和记录的硬拷贝可实现恢复。硬拷贝和形成文件的逐行验证提供了活动的书面证据。

## 维护

电子表格旨在满足短时需求。消息译文嵌入代码后，就会退役。因此，没有创建维护计划。

## 讨论

电子表格的预期用途和初始风险分析对作出电子表格需要进一步确认的决定，至关重要。在其他预期用途情境中，同样的电子表格可能会轻松地得出电子表格风险很低并且其复杂程度也很低的结论。如果预期用途只是跟踪译文收集进程（即电子表格中的译文不会用于实施活动的设计中），那么，结论可能一直是器械的整体性几乎不存在风险，而且事实上，电子表格是一种商业管理工具，甚至不属于受制范围。

该软件“自动化”的“过程”是该器械消息译文数据收集、格式化和存储过程的一部分。本示例在以下几个方面看起来很有趣：

- 确认（若存在）几乎不要求软件测试来确认软件的用途。需要注意的是，软件（Excel）和电子表格已针对此特定用途进行了确认，但未针对任何用途进行过通用确认。该团队认为测试不太可能发现软件的任何缺陷，但如果软件的确以某种不可预测的方式出现了故障，那对器械就是一个弱点。
- 确认包括对电子表格输出的100%地验证。硬拷贝版本被视为“黄金标准”。一旦硬拷贝获得批准并用在设计历史文档中，软件的任何后续失效都无关紧要。批准之前软件的任何失效都会在审批过程中捕获。
- 修改“过程”，使其免受电子表格软件任何失效的影响。
- 工程师们认为，此应用程序出现人为失效的可能性远高于出现软件失效的可能性。用户可能会出印刷错误，可能使用错误版本的电子表格，或可能会出现类似错误。在这种情况下，“软件确认”也使该过程避免了人为错误。
- 本示例强调了配置管理的重要性，即使对于日常办公生产工具也非常重要。

注：本示例基于一个控制不充分的真实案例。实际情况是，电子表格的版本控制出现了人为错误。出乎意料的是，与在不同PC上安装不同Excel有关的字体版本问题给出了不同的硬拷贝结果。（打印机字体在不同打印机上也成了问题。）看似简单的电子表格，一个几乎认为不需要确认的电子表格，实际上成了消息译文损坏的问题。



### 示例8：参数灭菌器

玛丽的任务是领导一种新的自动灭菌系统的确认工作，该系统将由她所在公司（恒安医疗器械公司）定制开发。

#### 定义过程

玛丽首先定义了她所了解的100%的环氧乙烷（ETO）灭菌过程并将其形成文件，她所在工厂即将引入该过程。

- 医疗器械由人工放入灭菌器内。此过程包括灭菌周期参数评价，以支持参数放行。
- 自动灭菌器系统软件控制灭菌循环活动。
- 循环完成后，医疗器械由人工取出，并将其转移至换气室。

#### 过程风险分析

玛丽非常关注这一过程带来的风险。此过程失效可能会产生严重后果，包括：

- a) 医疗器械灭菌不当。这种失效可能会因使用非无菌产品导致感染进而导致严重伤害或死亡；
- b) 器械历史信息和产品可追溯性缺失；
- c) 有毒化学品释放进入制造设施或环境。这种失效可能导致灭菌器操作人员或当地社区居民严重伤害或死亡。

因此，玛丽考虑宜采取哪些风险控制措施并加以验证以降低这些风险。玛丽认为可通过使用参数灭菌技术来控制风险，确保在正确温度和相对湿度下，在合适的时间段使用适量的气体。此外，人工检查灭菌器数据，适当的参数值将独立确认灭菌充分。最后她认为，需要采用失效安全停机 and 屏蔽壳结构，以控制化学品泄漏进入设施。

有了这些风险控制措施，除非同时发生多个系统失效，否则，不可能导致非无菌器械。但是，鉴于这种失效造成的影响，玛丽认为剩余过程风险很高，因此，严格确认是适当的。

#### 定义软件目的和意图

玛丽想详细了解该系统中软件如何使用。首先，她考虑软件应该做什么。在这种情况下，软件通过采用一个100%的ETO灭菌容器来控制医疗器械的灭菌过程，包括记录下器械历史记录中包含的信息以及分析用以支持参数放行的灭菌值。购买新型灭菌器的原因是新型灭菌器比现有系统可容纳的批量更大；为满足当前的产品需求，需要大批量灭菌。灭菌操作人员将和质量保证团队一起使用该系统，确定医疗器械放行的可接受性。玛丽知道，这项工作将在灭菌周期内通过实时控制和监控灭菌器以及在某数据库中存储信息实现。玛丽很高兴地获悉，该系统将实际安装于现场灭菌设施内，而且该系统通常每周关闭一天，以便进行任何必要的维护。

玛丽确定该软件将使灭菌周期实现全方位自动化，从人工将器械放入容器内，直到从容器中人工取出该器械。

玛丽将如下软件的目的和意图形成文件：

- 灭菌软件将控制并监视灭菌过程，并将评价决定参数放行的灭菌周期参数。

#### 确认策划

现在玛丽了解了该软件的预期用途，也就可以着手在高层次上制定确认计划了。她知道后续需要增加更多细节，但她现在就想开始进行确认策划，以便能够在知情的情况下识别软件失效的风险，并用已识别的风险完成计划。

由于之前她识别出有很高的剩余过程风险，玛丽认为她需要提供详细的确认内容和确认程序。她希望文档中采用高级别的严谨程度和详细程度，希望大多数文档以独立文档的形式存在，而不像平常为了减小工作量将其组合在一起。由于与该系统关联的高风险，她决定以开发医疗器械软件使用同等的严格程度对待开发。因此她决定采用YY/T 0664—2020/IEC 62304:2006/ AMD1:2015标准作为生存周期控制方法。有关软件风险管理的指导，她参考了YY/T 1406.1。此外，为了确保不遗漏任何潜在伤害来源，玛丽决定将软件故障树分析应用于开发工作。她还决定正式定义用户业务过程要求和软件需求并将其形

成文件。将明确识别任何特别关注的功能。玛丽还安排了一次正式的软件需求评审。必须得到质量保证团队、灭菌工程师和灭菌经理的批准，由于本系统的关键性和风险，确认报告的最终批准将包括高级管理人员。

### 定义软件需求

玛丽现在编写软件需求定义。她认为软件需求宜涉及警报、错误处理和消息、参数设置的确认、与器械历史记录系统的接口、传感器控制和监视、运动控制和监视。

### 建立对软件的信任和控制

玛丽使用恒安内部的开发控制程序作为推动程序，在整个开发生存周期中使用内部控制。因为全部工作均在内部完成，所以无需进行供应商活动。

### 定义与其他系统的软件边界

然后玛丽考虑了新的灭菌器需要与哪些其他系统相连接。她确定唯一的接口将是与恒安现有数据库系统的接口，该系统将存储灭菌周期中生成的数据。

### 软件失效风险分析

虽然玛丽已经断定即将实现自动化的业务过程风险很高，但她仍然需要分析软件失效的风险。参考本文件，玛丽为此活动选择了一个定量风险模型。她对新系统划分等级如下：

- “严重度”的风险很高（10分），因为系统失效可能导致死亡或严重伤害。
- “可能性”风险也很高（10分），因为软件失效本身可能导致伤害，因为软件正在确定灭菌结果的可接受性。

玛丽计算出风险评分为20，也就是高风险分类。高风险分类意味着宜采用严格的确认方法。采用将灭菌器本身作为医疗器械所要求的严格程度和全面程度相同。

由于采取了降低措施，该自动化系统的剩余风险已尽可能地降低。但由于系统可能带来伤害的严重度，灭菌本质上是一个高风险的过程。还执行了与风险（引自IEC/TR 80002-1）有关的其他活动。

### 完成确认计划

因为玛丽现在已经完成了软件需求的定义，确定了实施方法并分析了软件风险，她有充足的信息来完成详细的确认计划。

在撰写确认计划的初稿时，玛丽已决定采取严格的风险管理方法。她已经计划以非常正式的方式对待确认工作。

因此，她描述了自己计划使用的风险管理工具（在IEC/TR 80002-1中识别的），见下文：

- 风险管理工具：
  - 软件故障树分析；
  - 风险管理计划；
  - 确定制造过程或业务过程内的风险控制措施；
  - 软件失效分析（风险分析）。

然后，玛丽考虑了自己该如何在软件设计开发和配置阶段建立信任。她已经决定采用YY/T 0664—2020/IEC 62304:2006/AMD1:2015标准进行生存周期控制。现在她确定了将在设计开发和配置阶段确保软件正确开发的其他特定工具。

- 设计开发和配置工具：
  - YY/T 0664—2020/IEC 62304:2006/AMD1:2015架构文档与和评审；
  - 设计规范；
  - 软件详细设计和评审；
  - 软件编码标准；
  - 可追溯性矩阵；
  - 识别软件系统设计中的风险控制措施；

- 代码评审与代码验证；
- 开发和设计评审。

玛丽毫确信需要对这个新系统进行全面测试。她首先确定需要进行正式的测试计划活动以及通常的单元测试、集成测试和接口测试活动。但是，由于该系统将实时放行器械成品，因此，她决定通过压力测试、性能测试和更全面的输入测试组合模拟系统极限，以尽可能多地模拟操作条件。

#### ——测试工具

- 测试策划；
- 单元测试；
- 集成测试；
- 接口测试；
- 回归测试（必要时）；
- 软件系统测试；
- 稳健性（负荷）测试；
- 输入测试组合；
- 性能测试。

最后，因为知道系统在生产环境中完全实现之前是不完整的，所以玛丽更关注她希望在部署阶段看到的确认活动。她希望确保系统形成充分的文件，而且用户接受过正确使用该系统的良好培训。她还希望确保系统确实已按预期安装。因此，玛丽为部署阶段制定的确认计划现在包括以下项目：

#### ——部署工具：

- 用户程序评审；
- 内部培训；
- 安装鉴定；
- 运行鉴定与性能鉴定；
- 操作员认证。

#### 维护策划

由于存在高剩余风险，玛丽很关心软件的维护。她计划多项维护活动，在系统部署完毕后可确保软件的质量，包括评价用户培训的有效性、系统监控技术，系统输出和缺陷报告的正确性检查。除软件维护活动外，她还确认了校准和其他硬件维护活动也在进行。

#### 退役活动

由于该系统生成的数据需要存为器械历史记录，而且老格式与新格式不兼容，玛丽极力要求原有系统退役。新系统采用通用数据格式，以便在退役时可以灵活地将存储的数据转移到新系统内。

### 示例9：不合格物料报告系统——全面系统升级

“高级医学专业公司”正在升级其商业软件包不合格物料报告系统（NCMRS）软件。“高级医学”选择在上一次主要版本时不升级，所以正在运行的系统落后两个主要版本。（“高级医学”目前运行的是版本2，而最新版本是版本4。）为了维护当前的软件维护协议，“高级医学”需要升级软件。NCMRS-Pro软件的版本4与以前的版本相比有很大改变。除此之外，该产品已从典型的客户端—服务器应用程序重新构建为基于web的应用程序平台。新软件还包括重要的新特征和新功能。弗兰克是“高级医学”的业务过程负责人和项目经理，他对现有软件和过程没有新的要求，但他确实希望利用新的软件功能。

弗兰克咨询了法规团队，认为ERP系统和NCMRS之间的当前接口能够完整保留，无需修改。然而，弗兰克认识到新版本能够将数据写回ERP系统，而且在确认过程中这个扩展接口宜受到彻底质疑。弗兰克和同事们以及制造质量工程师和法规团队开始着手确定确认工作的范围。本示例的后文中，该小组被称为“团队”。

#### 定义过程

弗兰克首先分析了当前的手动过程，以确定新软件将自动化工作流程的哪些元素。新软件将更改以下功能：

- a) 识别潜在不合格物料或产品（超出范围）；
- b) 输入与该物料有关的信息及其发现时周围环境情况（范围内）；
- c) 传递信息，以便正确识别、评价、调查和处置物料（范围内）；
- d) 将信息发送至重要利益相关方以及为正确处理财务、采购、计划和调度事宜所需的其他计算机系统（范围内）；
- e) 物料的实际处置，尽管处置的相关数据将记录在系统中（超出范围）。

#### 过程风险分析

弗兰克意识到过程和支持软件存在风险。过程失效可能产生严重后果，包括：

- 因疏忽大意使不合格物料流入制造车间；
- 因疏忽大意使不合格产品流入商业销售；
- 由于报废、返工等原因导致成本增加或制造增加。

弗兰克和法规团队考虑了采取哪些风险控制措施以降低这些风险，包括：

- 检测、隔离、控制并纠正不合格物料的程序控制；
- 对统计过程控制数据以及其他措施进行管理和质量评审，以确定在过程未得到适当控制时可能发出信号的发展趋势；
- 对操作员进行持续培训，以确保符合程序；
- 财务报告，以帮助识别物料使用情况，这会表明制造过程中的异常问题。

这些风险控制措施实施后，只有多个系统失效同时发生，才能出现不合格材料或产品无法适当控制。但是由于此类失效可能对质量、法规和财务产生的影响，弗兰克确定对于剩余过程风险严格的信任建立活动是必要的，以帮助确保软件正确运行并满足预期用途。

#### 定义软件目的和意图

弗兰克想详细了解软件升级将如何影响他的用户和组织。弗兰克得出结，该软件本质上是一个自动化问题跟踪和管理工具。使用标准工具、设备和其他仪器的制造人员负责识别并隔离有可能不合格的物料和产品。一旦发现问题，有关情况的详细信息将会输入软件。然后，软件管理工作流程、分配并通知以解决该问题，并且记录物料和产品处置所需的各种活动。软件升级宜简化过程，从而提高效率，而且过程宜为质量保证团队提供更强大的分析工具和趋势数据，并使团队更清楚地了解质量问题。弗兰克清楚，升级所需的过程更改主要针对的是工作过程和信息分发。软件本身既不做最终决定，也不独立确定任何结果，但软件会保存并记录与系统交互的人所做的决定。

弗兰克确定，软件将使不合格处理工作流程的各方面实现自动化。以下是法规团队撰写的关于软件目的和意图的说明：

——NCMRS 软件旨在支持不合格物料及产品的处理。该系统用于记录 SOP 定义的过程步骤，并记录执行的过程步骤、执行时间、执行人员以及每个步骤的结果。该系统使数据随时可用于质量监视和改进活动。

### 定义与其他系统之间的软件边界

NCMRS 软件有两个界面，包括一个与 ERP 系统的主接口和一个与公司人力资源 (HR) 系统的辅助接口。主接口设计为一个预设每日两次的批处理的过程，以便使用成品、在制品、物料清单和操作清单的数据进行系统更新。该接口反向还将向 ERP 提供不合格物料报告 (NCMR) 数据，其中包含有关质量保留、物料处置和其他事务信息的数据。辅助接口是仅接受来自 HR 系统的单向接口，以更新 NCMR 员工数据，用以进行调度和分配。

### 初步确认策划

弗兰克对 NCMR 过程和软件理解充分并适当地形成文件，从而获得更多信任。他准备制定高级别的确认计划。在其他细节将在策划过程中制定。

法规团队确定了将会增加最大价值并将充分说明软件预期用途的文件。这些文件将通称为“需求”，但本身并不是典型的用户需求。相反，这些文件将是一系列关于软件预期如何运行的详细描述。因此，自动测试和人工测试分析从评审和结果的角度看将更为定性。它将从整体上看待输出结果以及系统是否按照预期运行，而非着眼于单个测试，以确定是否已满足特定用户需求。

该文件集将包括以下内容：

- a) 工作流程与业务规则文档。软件的这一部分是可配置的，因此团队将准备软件所需配置集，并开发描述操作的详细过程流程图和逻辑图。
- b) 接口文档。这些文件将描述哪些数据元素从 ERP 和 HR 系统传递至 NCMRS、哪些数据元素从 NCMRS 传递至 ERP 系统以及何时传递这些数据元素。
- c) 数据迁移文档。这组文件将描述将哪些历史数据迁移至升级后的系统。

确认计划将包括每个文件的评审和批准结果。每个文件都必须获得质量保证团队、制造工程部门和信息系统部门小组的批准。由于该系统关键性和风险，高级管理层的所有成员都宜参与确认报告的最终批准。

### 定义软件使用要求

弗兰克和团队成员参考供应商提供的系统文档和现有接口以前的文档参与了收集上述文档集的事宜。

### 建立对软件的信任和控制

弗兰克对软件和供应商有过积极的体验。弗兰克现在确定了团队对软件建立对软件的信任需要完成的五项主要工作：

根据“高级医学”的内部方针和程序，供应商处在已获公司批准状态。以往审核表明，该供应商拥有充分的质量体系和 SDLC。供应商生产的商用软件用于受监管行业有着悠久历史，并与“高级医学”的预期用途类似。该供应商将定期接受审核，以保持该批准状态。

“高级医学”将采用供应商提供的自动化测试工具，以验证软件是否已正确安装并在测试套件的范围内运行。该工具可在几个小时内处理 8000 多种不同的事务。但是，该工具并不测试公司计划包含的某些配置选项。

该团队将制定一项附加测试计划，其中包括并行处理纸质实际不合格报告里的有效统计抽样。处理结果将接受评审，以确保结果准确、数据完整并符合程序。

该团队将采用抽样技术验证现有系统记录的数据转换和迁移，以确保历史记录保持其完整性。记录计数将用于验证是否 100% 转换。

将使用抽样技术验证数据接口，以衡量数据迁移的完整性和准确性。

### 软件失效风险分析

弗兰克使用本文件作为参考，确定所要求达到的确认严格程度。软件失效可能导致电子记录缺失、损坏或处理不当。风险的降低通过供应商的内部质量体系、软件安装鉴定（自动化测试工具）以及附加测试用例及验证来控制。由于下游过程的控制，该系统的剩余风险程度被认为合理可行。

### 最终确认策划

这一决定意味着将采用相当严格的确认方法。采用这种方法可以在合理的范围内确保软件按预期运行。团队成员得出结论：他们已经充分定义了系统要求，确定了实施方法、分析了软件风险，并已经获得的用于进行详细确认计划的足够信息。

要进行的大部分测试将使用自动化测试套件来完成，该套件经团队评审并确定其对预期用途有效。另外，还将使用来自生产车间的实际业务案例进行其他的辅助用例测试。这些测试的目的是：a) 验证过程按预期工作；b) 加速用户接受和培训；c) 验证配置更改未对软件产生不利影响。辅助测试无意取代供应商的内部系统测试。内部系统测试之前已经过审核予以验证。自动化测试成功完成将证实软件已正确安装且在功能上可接受。

团队从本文件中选择了以下工具来执行其余安装、配置、测试、验证和确认工作：

#### ——设计、开发和配置工具：

- 架构文档与评审；
- 确定软件系统设计中的风险控制措施；
- 配置设计评审；
- 评审供应商的“已知问题”清单；
- 评审供应商的基本系统确认文档；
- 评审“开箱即用”软件工作流过程图；
- 评审“开箱即用”标准报告库；
- 对标准工作流过程和业务规则进行配置更改差距分析。

#### ——测试工具：

- 测试策划；
- 供应商提供的用于安装验证和鉴定的自动测试工具的描述和结果；
- 安装与性能测试（自动测试套件的一部分）；
- 采用用例测试覆盖配置更改，以实际不合格记录作为输入，而非人工构建测试用例；
- 确认迁移数据的抽样计划；
- 确认操作接口的系统检查。

#### ——部署工具：

- 使用程序评审；
- 内部培训；
- 操作员认证。

### 维护策划

弗兰克计划在系统部署后，进行多项维护活动以确保持续软件质量，包括评用户培训有效性的评价、系统监视技术、定期内部体系及供应商体系审核输出和缺陷报告。弗兰克已与供应商建立了联络点，以便在“高级医学”负责软件维护的适当人员注意到缺陷通知、维护版本和其他沟通。

### 退役活动

弗兰克计划在切换发生后保持当前系统的可用性，以此作为比较生产量和结果的机会，从中可以编制性能指标。在新升级版本成功投入运行6个月后，弗兰克将完全停用以前的系统。

### 示例10：用于安排不合格物料报告（NCMR）评审委员会会议的软件

一家拥有1000名员工的公司决定尝试新的软件解决方案，帮助公司以电子方式安排会议，进行必要的NCMR评审活动。实施自动化的项目组听说刚刚发布了一种商业软件程序，供应商声称该软件可以使用通过其他计算机化系统接口接收的数据安排会议。项目组认为，如果该软件能够从公司确认过的NCMR数据库系统上收集NCMR数据，就可以很好地安排公司NCMR评审委员会的会议。

#### 定义过程

项目组聚在一起讨论NCMR评审委员会会议的安排过程，评审公司的NCMR处理程序。讨论产生了以下定义的过程：

- a) 一旦识别出某个不合格现象，相关物料上要贴上标签、立即隔离，并录入经确认的 NCMR 数据库。
- b) 每周举行一次会议，评审与不合格和推荐处置活动有关的所有调查结果。
- c) 每次会议都要确定一份 NCMR 清单以供评审，以及需要参加会议、陈述结果以及参与处置行动和批复的人员。
- d) NCMR 评审委员会会议召开的前一天，向需要参加会议的人员发送与会邀请。邀请里包括待讨论的 NCMR 的列表。

#### 过程风险分析

通过头脑风暴活动，团队成员评价该过程可能出现的失效造成潜在伤害如下：

- 未发送会议邀请；
- 会议邀请发送时间不正确；
- 邀请与会的人员不正确；
- 确定的 NCMR 列表内评审对象不正确。

项目组注意到已发布的NCMR处理程序要求指定一个人担任NCMR处理经理。此人负责确保及时处理所有NCMR，并根据经确认的NCMR数据库中的数据发布NCMR处理指标。在团队确定的所有情况下，会议安排软件造成的伤害是NCMR评审委员会会议效率中断。这种中断增加了NCMR处理经理的时间负担。因此，过程失效风险分析确定在监管风险、环境风险和人类伤害风险方面风险较低。

#### 定义预期用途

团队定义了以下软件使用、监管使用和边界的目的和意图：

##### ——软件使用

- 由谁使用？软件主要由NCMR处理经理使用。
- 用作什么？软件将自动向确定需要参加本周会议的个人发送电子会议邀请。
- 什么时候使用？在需要安排NCMR会议时，使用此软件。
- 在哪里使用？因为所有与会者都在本部，所以软件只需要在局域网（LAN）上使用。
- 如何使用？软件检索开放状态的、需要NCMR评审委员会评审的NCMR列表。NCMR处理经理确定将在下次会议上评审的NCMR。然后，软件使用NCMR处理经理设置的表格来确定需要参加特定会议的人员。会议日期由NCMR处理经理确定。软件提前一天向适当的参会人员发出电子会议邀请。
- 为什么使用？软件将用于改善通知适当人员参加NCMR评审委员会周会的时效。

##### ——边界

- 软件的边界位于NCMR数据库和图形用户界面的接口处。

##### ——监管使用

- 软件不存储任何用于证明任何监管要求符合性的信息。与NCMR或NCMR处理相关的所有器械历史记录信息都记录在纸上或记录在经过确认的NCMR数据库中。

在创建并评审目的与预期说明后，项目组确定拟定的软件既不会自动执行法规要求活动，也不会创建法规要求的质量记录。虽然软件所执行的会议是受监管活动（NCMR过程）的一部分，但软件本身并不

会使受监管活动实现自动化。因此，团队记录了之前列出的预期用途，并清楚地表明不需要进行正式确认。但团队也意识到，维护阶段使用方面的微小变化都可能会显著影响团队最初的确认决定。例如，如果软件用于存储会议记录或用于生成供监管调查员评审的参会人员列表，则最初的“超出范围”判断都将受到影响。因此，项目组更新了其质量体系程序，以包括定期或由于相关过程个更改对预期用途进行评价。

### 工具箱使用

使用了以下工具：

——开发—定义阶段：

- 定义过程要求；
- 过程失效风险分析；
- 定义预期用途。

——维护阶段：

- 维护策划。

### 讨论

通过识别该软件的具体用途和自动化活动的边界，团队能够适当地声明：该软件不符合医疗器械质量管理体系过程软件的定义，因此，该软件不需要确认。在识别此类软件时宜格外小心，以确保预期用途定义完全涵盖了该软件的的实际用途。另外，即使软件没有更改，预期用途也可能在生存周期的维护阶段轻易改变，认识到这一点也很重要。因此，维护策划是确保公司使用软件得以适当控制的重要部分。



### 示例11：经批准的供应商列表系统

颠峰公司是一家B类医疗器械制造商。公司一直使用人工程序维护已批准的供应商列表（AVL）。颠峰公司想开发一个AVL系统，自动检查供应商是否已经获批准提供特定部件。

——颠峰公司新AVL系统项目经理杰克认为，AVL过程是与GB/T XXXXX/ISO 13485中的采购控制有关的医疗器械质量体系过程。

因此，拟定的AVL系统要符合软件确认要求。

#### 定义过程

为了更好地理解开发AVL系统所涉及的要求和风险，杰克确定了以下相关业务过程：

- 当工程小组希望新的供应商获得批准时，该供应商部件的样品将提交给质量组进行鉴定。
- 在证明供应商的部件合格后，质量组会向采购组发送电子邮件，授权将供应商名称和批准的部件号以及部件说明输入已批准的供应商列表（AVL）。此列表在采购组已纸质文件保留。接收检查组可以访问AVL。
- 采购组通过人工检查以验证供应商的名称已正确添加到AVL中。
- 在订购部件时，采购组会参考AVL列表，确保供应商已获得批准，有权提供所需部件。
- 如果供应商获得批准，则采购组在请购单上签字，表明他们已经检查了AVL列表。

#### 过程风险分析

然后杰克考虑当前过程可能出现什么问题。如果过程故障，可能会从未经批准的供应商处订购零部件，原因可能是在AVL列表中增加了未经批准的供应商，也可能是因为采购组在订购零件之前未检查AVL列表。

然后杰克考虑采取了哪些风险控制措施以降低这些风险。杰克发现，采购组有一个人工检查供应商名称是否已正确加入AVL列表的程序，而且该列表的访问权限限于授权员工。此外，杰克还发现，当前的采购程序要求采购组必须在签发采购订单之前签字保证该供应商在AVL列表上。确保向已批准的供应商下订单的控制归属于接收检验部门，在收到零部件时要再次检查AVL列表。

杰克根据这些风险控制措施确定剩余过程风险较低。因此他怀疑新的AVL系统可能是一个低风险系统。

#### 确定预期用途

现在，杰克理解了要自动化的业务过程，他为拟定的新AVL系统制定了以下目的和意图说明：

——AVL系统将对照电子AVL列表自动检查供应商和部件，以确保仅向授权供应商订购部件。新系统将使用AVL数据库，该数据库与现有采购订单系统相链接，并且颠峰公司总部的质量小组在供应商资格鉴定过程中将使用该数据库，采购代理在采购订单生成过程将使用该数据库。

杰克还考虑了AVL系统将与之接口的其他系统和过程，并在说明中添加了一些语言来澄清新系统的界限。

——采购过程将与AVL系统自动化过程相接口。该接口将包含一个AVL数据库查询系统，用来查询采购订单上指定供应商的状态。采购过程不确认AVL列表数据的准确性，也不与供应商评估过程相接口。

#### 确认策划

现在杰克已经了解了有要自动化的业务过程，并且已经确定了新系统的目的和意图，他准备在高层次上制定确认计划。他知道将在后续更详细地充实这个计划，但现在就想开始进行确认策划，以便确定所需确认工作量的级别。

杰克之前已确定，现有AVL过程存在低的剩余过程风险。因此他认为不需要在确认工作中提供太多的细节或形式。他知道为新系统定义用户业务过程要求和软件需求非常重要，但由于该系统风险较低，杰克不需要独立文件，每个文件都有单独签名。因此，他决定以表格的形式将用户业务过程要求、软件需求和测试计划合并到一个文件里。

此外，由于该系统风险很低，杰克认为不需要对确认工作进行广泛的管理评审。他认为由供应商开发经理和质量保证代表批准就足够了。但他也认为，为了确保用户需求正确，还应该加上采购组代表的评审。

杰克根据自己的决定开始起草确认计划。颠峰公司有对所有确认计划均宜采用的标准格式。确认计划的某些部分没有定义，但杰克会在初始系统设计获得批准后更新该计划。

### 定义软件需求

杰克现在编写软件需求。他认为软件需求宜包括“做什么”（AVL过程或系统所要求的行措施）、AVL系统如何与采购系统接口的接口规范、数据字典以及新系统宜能够处理的有效查询示例。

### 确定软件与其他系统的边界

杰克然后考虑了新AVL系统需要与之接口的其他系统。他确定唯一的接口将是颠峰现有采购系统。该系统可以通过简单结构化查询语言（SQL）查询AVL数据库。

### 建立对软件的信任和控制

杰克现在需要决定使用什么方法、采用什么技术来构建新系统。因为业务要求相当简单，所以交易量会很低。由于AVL系统是一个低风险系统，杰克决定使用Microsoft Access<sup>1</sup>开发这个系统。Microsoft Access是一种广泛普及且易于使用的数据库系统。

由于Microsoft是外部软件开发公司，杰克需要决定自己应该通过哪些类型的活动对Microsoft Access建立信任。杰克注意到Microsoft Access的使用很普遍，而且过去该产品的任何问题都会快速识别并在Internet留言板上及时公布。结合AVL系统风险较低这一事实，杰克决定不需要作为数据库开发人员对Microsoft进行供应商审核。

由于新系统中包含电子记录，杰克决定围绕Microsoft Access实施第三方“打包”软件，提供所需控制以确保记录的有效性。

### 软件失效风险分析

虽然杰克已经确定自动化的业务过程风险很低，但他仍然需要分析软件失效的风险。他决定对此活动使用定量风险模型（等级从1到10）。他对新系统分级如下：

- 杰克将“严重度”列为中等（6），因为软件失效只会间接造成伤害。他的分级是根据过程中存在下游控制。
- 杰克将“可能性”排在低位（1），因为数据库设计非常简单，使得在测试期间不太可能发现关键错误。
- 排名的组合转化为低风险分类。

因此，杰克将执行适合低风险级别的确认任务。

### 完成确认计划

现在杰克已经定义了软件需求，决定了实施方法并分析了软件风险，他有足够的信息来完成确认计划。此时，他退后一步，根据对该系统、实施方法和软件风险的所有了解，向自己提出了这样的问题：“哪些确认活动才能真正让我相信这个系统适合其预期用途？”。

由于该系统是采购的数据库工具，风险相对较低，因此杰克认为自己精心安排的确认活动已经足够，但他还需要满足环境要求，以确保对操作系统和Access版本的更改进行很好的控制。他更新了确认计划要求进行正式的软件配置控制。

<sup>1</sup> Microsoft Access 是一个合适的商业可获得产品的实例。提供本信息是为了方便本文件使用者，并不构成 ISO 对该产品的认可。

由于该系统是第三方开发的，杰克需要确定开发人员正确地转换了自定义要求、输入、接口、数据存储和输出的要求。因为该系统将依赖于现有系统的输入，杰克在确认计划中添加了接口测试和集成系统测试，作为确认开发人员工作正确性的重要活动。

最后，杰克希望确保开发人员在开发过程中保持适当的版本控制，因此他将软件版本控制作为一项必要活动添加至确认计划中。

因此，杰克的批判性思维引导使他为其余的开发和确认工作准备了以下工具：

——设计、开发与配置工具：

- 软件架构文档与评审；
- 可追溯性矩阵（要求规范中固有的）；
- 风险控制措施（记录在用户规范中）。

——测试工具：

- 集成测试（记录在要求规范中）；
- 接口测试（记录在要求规范中）；
- 软件系统测试（记录在要求规范中）。

——部署工具

- 用户程序评审；
- 软件应用的内部培训；
- 安装鉴定。

#### 维护策划

杰克提前考虑了系统部署后哪些活动对确保软件质量可能是适合的。鉴于系统剩余风险较低，杰克认为宜每季度对数据库中AVL数据的准确性进行评审。杰克在其确认计划中包含一个章节来记录季度评审，并要求开发和实施一个程序，以确保在系统上线后进行季度评审。

## 示例12：校准管理软件

XYZ医疗公司发展迅速。XYZ收购了欧洲和亚洲购的公司。该公司的成长意味着XYZ的校准管理需求也在增长。目前，XYZ校准经理将所有校准信息保存在一本书中，每周检查校准设备库存，以确定是否有任何项目需要重新校准。随着公司的发展壮大，公司库存越来越庞大而且分散在全球各地，一个人很难靠纸质系统进行管理。是时候建立一套计算机化的系统了。

### 定义过程

XYZ医疗公司有一个标准的操作程序（SOP），对于质量体系的一部分进行自动化的计算机系统要求对其预期用途进行确认。XYZ首先定义了校准管理过程，了解过程中存在哪些固有风险，同时确定软件解决方案是否会自动化当前过程的全部或一部分。XYZ经理评审了有关校准管理的标准操作程序（SOP），该程序包含有关以下步骤的详细信息：

- a) 采购新设备；
- b) 赋予新设备唯一的标识（10）编码；
- c) 校准新设备；
- d) 在设备上记录校准状态；
- e) 维护校准记录，包括校准要求、状态和失效日期；
- f) 搜索校准记录，以便报告并开展校准管理活动。

### 过程风险分析

- 无论是纸质系统还是电子系统，校准管理过程都会带来一些固有的风险。与该过程相关的风险如下：
- 若一台设备在校准期满后仍在使用，该设备所记录的测量值就会不正确。这个问题可能会带来诸多后果，具体取决于该设备及其使用阶段。
  - 一台设备上打上不正确的标签，将表明该设备接受校准的时候超出了实际校准期限。此错误也会产生诸多后果，具体取决于该设备及其使用阶段。
  - 校准记录可能丢失，而且校准过期的设备可能积压。这个问题能够造成工作延误。
  - 如果校准状态记录不正确，则能导致过期设备投入使用。
  - 如果两个设备收到的识别编号相同，则记录将不唯一。

鉴于校准不正确存在诸多潜在不良结果，公司认为该过程风险很高。最坏的情况下，校准过期的设备可能用于最终验收的医疗器械的测量，而且该设备可能在不宜通过验收的情况下认定合格。

为了减轻该问题，XYZ经理宜更新SOP的信息，在其中补充针对设备每个使用者的指导。每个使用者在使用前都应检查设备上的校准期限标签。在校准设备使用期间，每个使用者均宜记录设备识别编号和所用设备的校准期限的日期。用户还要在如何识别需校准项目方面接受培训，懂得不要使用任何带有过期标签或没有标签的设备。

实施这些措施使系统的剩余风险降至中等水平。XYZ经理认为，使用者指导是一种适宜的措施，但效果不足以将剩余风险降至更低。

### 定义软件预期用途

该软件系统不执行校准活动；该软件系统将只是一个包含有关设备及其校准历史和状态的校准信息和数据的数据库。该软件系统将控制校准过程中步骤b)、f)和g)。

XYZ经理同意XYZ将确认系统的以下目的和意图：

- 校准管理系统用于为需要校准的设备提供识别编号、打印校准设备标签、存储校准结果数据、报告设备的校准状态。校准管理系统自动化了GB/T XXXXX/ISO 13485 部分要求，涵盖了检查、测量和测试设备。

### 确认策划

为了设置确认活动的阶段，XYZ经理开始进行确认策划，通过对可交付成果和过程中跨职能部门的参与设定期望值的方式。他们记录了以下几个步骤：

- 确定所选工具的文档严格程度。
  - 系统的文档严格程度将为中等。因此，关键可交付成果将单独创建和批准。
- 确定所选工具的审查级别（管理层和跨职能参与并评审）。
  - 鉴于该系统将在全球范围内用于校准管理，全球信息技术管理与运营管理在该系统确认过程中宜以批准系统确认计划和确认报告形式具有可见性。此外，新的现场设备经理将参与所有文件的评审和批准。
- 在工具箱中选择“定义”工具：
  - 用户和业务过程要求；
  - 软件需求；
  - 正式的软件需求评审。

### 定义软件需求

软件需求将包含以下要素：

- 功能性工作流程；
- 电子记录和电子签名要求；
- 数据逻辑要求；
- 报告要求；
- 设备标签打印的具体要求；
- 用户信息安全与用户资料；
- 性能要求；
- 容量定义。

### 建立对软件的信任和控制

XYZ经理对三家此类产品的供应商进行调查，确定其中一家供应商的产品最符合XYZ所计划的预期用途。虽然该版本产品相对较新，但该系统的供应商被医疗器械行业使用广泛。从以往几个版本的跟踪记录中能够获得一些信任，但还将根据当前报告的问题进行已知缺陷分析，测试开发组还将对区别于先前发布版本的新功能测试进行特别审查。

### 定义与其他系统的软件边界

该软件与其他软件系统没有接口。

### 软件风险分析

确认团队坐下来与全球的校准管理人员一起采用[表C.16](#)中的问卷确定该软件的风险。他们首先识别风险，然后针对那些风险确定了风险控制措施；最后，评估了剩余风险的可接受性（见[表C.17](#)）。

表C.16 示例 12——风险分析

	风险识别问题	表明“是”或“否”。如果是“是”，则指定一个风险标识符（风险 1、风险 2、…… 风险 n）
1.1 产品安全（伤害）	<p>如果软件出现故障，是否产品安全存在潜在风险？</p> <p>——是，始终存在。非校准的设备可能会被软件错误识别为校准设备。</p> <p>——患者伤害——是。如果测量中使用了非校准的设备，则患者可能会使用规格不合格的产品。</p> <p>——操作人员伤害——是。如果温度或力的测量错误，操作人员可能被挤压或受伤。</p> <p>——旁观者伤害——是。这种伤害取决于设备。</p> <p>——维修人员伤害——是。如果温度或力测量错误，维修人员可能被挤压或受伤。</p> <p>——环境危害——是。如果压力测量错误且容器中含有对环境有害的物质，则容器可能泄漏。</p>	风险 1 使用的是非校准的设备
1.2 产品安全（伤害）	<p>如果软件用户出错，产品安全是否存在潜在风险？</p> <p>是，始终存在，如果用户输入不正确的设备校准数据（见 1.1）。</p> <p>——患者伤害——是。</p> <p>——操作人员伤害——是。</p> <p>——旁观者伤害——是。</p> <p>——维修人员伤害——是。</p> <p>——环境危害——是。</p>	见风险 1。
2.1 产品质量	<p>如果软件出现故障，产品质量是否存在潜在风险（安全风险除外）？</p> <p>是。产品可能会不符合规格，因为非校准设备可能被软件误认为校准合格设备。虽然错误识别并非安全问题，但可能引起顾客不满。</p>	见风险 1。
2.2 产品质量	<p>如果用户犯错，产品质量是否存在潜在风险（安全风险除外）？</p> <p>是。如果用户为该设备输入错误校准数据，而且该设备用于测量产品，则该产品可能不符合规格。虽然规格不正确并非安全问题，但可能引起顾客不满。</p>	
3.1 记录完整性	<p>在作为记录存储库的系统中，记录的完整性是否存在潜在风险？</p> <p>记录缺失——是。校准记录可能丢失。</p> <p>记录损坏——是。校准记录可能被损坏。</p>	<p>风险 2—— 校准记录丢失并导致合规性问题。</p> <p>风险 3—— 校准记录被损坏并导致合规性问题。</p>
4.1 证明符合 ISO 标准	<p>合规性证明能力是否存在潜在风险？</p> <p>记录缺失——是。校准数据可能丢失。</p> <p>记录损坏——是。校准数据可能被损坏。</p>	见风险 2 和风险 3。

表C.17 示例 12——风险评价与控制

风险识别符	描述	严重度	控制措施	剩余风险
风险 1	非校准设备用于测量产品或用于测量压力或力。（风险是由于软件错误识别设备或由于用户错误输入设备校准数据而产生的。）	高	系统设计用于打印包含设备 ID 号、序列号、校准状态和到期日期的标签。按照程序，员工在使用设备之前接受培训以便验证此信息。另一个过程要求在输入的数据提交给校准记录前由第二人对其进行验证。	可接受
风险 2	记录缺失，校准管理活动无法得到保护。	中	所有校准数据都以纸质记录保存在校准室中。	可接受

表C.17 示例12——风险评价与控制（续）

风险识别符	描述	严重度	控制措施	剩余风险
风险 3	记录损坏，校准管理活动无法得到保护。	中	所有校准数据都以纸质记录保存在校准室中的。	可接受

风险分析完成后，XYZ经理确信降低后的剩余风险是可以接受的。

#### 完成确认计划

为完成确认计划，经理修改了计划以包含了以下工具选项：

##### ——实施工具：

- 可追溯性矩阵；
- 系统配置评审。

##### ——测试工具：

- 尺寸分析；
- 测试策划；
- 供应商提供的测试套件，对计划的配置以及区别先前版本软件的新功能进行附加测试。

##### ——部署工具

- 软件应用的内部培训；
- （服务器和工作站的）安装鉴定。

#### 维护策划

除了制定系统的确认计划外，XYZ经理还认为进行系统维护策划是有益的，因为某一时刻系统肯定需要维护。

系统监视技术将到位，以评审所有缺陷、使用问题及预期用途的更改。

将制定一项计划，对系统（即硬件、升级、补丁、信息安全问题）的更改进行分类，以便更有效地实施更改。



### 示例 13: 自动视觉系统

加里所在公司的工程师们工作很在行。他们了解在加里自动化区域生产的产品（1.27cm到3.81cm不同长度的金属棒），因此他们发现这种金属棒有两种应用：一种使用长度尺寸不大于2.54cm的金属棒；另一种使用长度尺寸为3.18cm（±0.64cm）的金属棒。棒材的宽度均为0.32cm。两种应用都适用于医疗器械，要求金属棒具有指定长度。加里是自动化领域的工程师，负责进行确认零件分拣的自动视觉系统。该系统将取代手动测量/分拣过程。该过程无其他更改，因此这就是确认的全部内容。

#### 过程描述

两种应用的棒材厚度的规格相同，该尺寸根据棒材切割机所使用的原材料得以确定。所有接收准则均在上游确认，除了但由加里的自动视觉系统进行测量的金属棒长度。

机器的工艺很简单。将棒装入一个容器中，在容器内用漏斗将金属棒逐个放在传送带上。每根金属棒都被传递到一个停靠点，在那里，摄像头对准金属棒并测量其长度。根据结果，把金属棒输送到两个容器中：一个用于长度不大于2.54cm的金属棒，另一个用于更长的金属棒。

下游没有金属棒的长度的其他检查。如果使用错误尺寸的金属棒对患者造成伤害的风险则会增加，因为错误尺寸的金属棒可能导致正在制造的设备泄漏。尚未设计出测试下游风险增加的方法。如果金属棒的尺寸恰在指定尺寸范围之内，则设备不会出现泄漏。设备已经制造多年，该风险众所周知。自动视觉系统正在取代手动测量过程。

#### 定义预期用途

因为加里了解正在实现自动化的过程，所以他从定义目的和意图开始。  
该软件旨在确认单根金属棒处于传送带上并测量其长度。

#### 风险分析

加里使用本地风险分析过程确定系统失效风险很高，因为除了产品失效或破坏性测试外，无法检测何时使用了错误尺寸的金属棒。失效可能导致患者伤害。该过程的关键参数是金属棒的精确长度尺寸。自动化既不会增加也不会降低该风险。

#### 确认策划

在确认策划的第一次迭代时，加里计划采用严格的确认过程（来自风险分析的高风险评级的结果）。在评审了潜在确认工具的工具箱后，他计划了一份正式的要求定义文件，并且安排了一次软件需求评审。评审将由制造工程师、另一位自动化工程师和质量工程师参加。

该系统的软件将在内部开发，但基于以往的系统自动化开发相对简单。

#### 风险控制措施

确定了两个重点风险领域。

需要确认一根金属棒处于可测量的位置。机器沿着狭窄的通道（宽度0.64cm，高度0.48cm）传送金属棒。因此，金属棒只能纵向进入通道，如果两根叠在一起，因为开口不够大，则不会进入。但是，两个部件在输送机可能前后紧贴在一起。

为了降低此风险，软件将在检查长度之前检查每根金属棒的宽度。如果棒宽大于0.32厘米（±0.08cm，先前检查的规格），则会拒收该金属棒，因为输送机上有两根金属棒。因此，在机器设计中加上第三个容器（废料箱）。

金属棒之间可能过于靠近，以致无法判断一根的d结束和另一的根开始。软件会将把任何无法确认其长度小于或等于3.81cm的金属棒传送至废料箱。

#### 确认任务

接下来，加里投入确认任务。他确定需要一份正式设计文件，并计划由进行要求评审的同一组人员对每个部分进行正式检查。此外，一旦代码生成，代码将由其他自动化工程师和制造工程师对照设计进行评审，这些工程师都有软件开发经验。没有选择供应商管理活动，因为该软件正在进行内部开发。要求自动化工程师、制造工程师和质量工程师评审软件和设计对应要求的可追溯性。测试后，他们还将重复相同的步骤，以确保所有要求都经过全面测试。

加里从工具箱的测试部分中选择的工具包括测试策划，测试策划将包括软件环境的详细信息和预期测试结果。他在开发的不同阶段设计了多种类型的测试，包括单元测试、集成测试和系统测试。将采用正常和错误测试用例以及与传送带速度相关的性能测试。除加里外，测试策划还需要由其他自动化工程师、制造工程师和质量工程师评审和批准。测试报告包括实际测试结果、与预期结果相比、通过/不通过的指示、测试标识、对于任何失效的问题解决以及回归测试文档。该测试报告，加里需要得到同组人员的批准。

#### 实施、测试与部署

对于该自动视觉系统的部署，加里评审了工具箱中的部署工具，认为需要进行安装鉴定。此外，他还认为应创建一个用户程序，系统的使用者需要须通过操作员认证。

#### 维护

加里所在部门共同为制造场地上的所有系统制定了维护计划。该领域无需特别策划或措施。

### 示例14: 拣选和放置系统

“高品质医疗公司”是一家B级医疗器械生产制造商。“高品质”希望将部分完成的部件从某一工位自动放入该公司生产医疗器械的盒子中。

吉尔是这个新型拣选和放置系统（简称P&P系统）的项目经理。根据GB/T XXXXX/ISO 13485标准，吉尔确定P&P过程是个医疗器械质量体系过程，因为该过程是医疗器械制造的一部分。因此，拟定的P&P系统将符合软件确认要求。

#### 定义当前过程

为了更好地理解开发P&P系统所涉及的需求和风险，吉尔定义了以下相关业务过程：

- a) 制造过程中，将工位 11 的部件放入工位 12 的盒子中（每个盒子放入 20 个部件）。目前，此操作由一名操作员手动完成。
- b) 然后操作员手动将该盒子放在工位 12 的进线轨道上。
- c) 操作员手动检查盒子，以确认部件放置正确。【步骤 b) 和 c) 每装一个盒子需要 3min。】
- d) 盒子继续进入其他组装步骤，其中包括目视检查，以确认该过程所有先前的步骤没有发生变形。

#### 过程风险分析

接下来吉尔考虑当前过程可能出现的问题。分析表明可能发生以下事件：

- a) 操作员可能使部分完成的部件变形。这种畸形将在下游检测工位检测出来。
- b) 操作员可能错误将部件放入盒子中或者错过盒中的某个插槽。在手动检查期间，放错盒或错过插槽的检测目前在工位 12 完成。

根据这些风险控制措施，吉尔认定剩余过程风险较低。因此，她预期新型P&P系统也将是个低风险系统。

#### 定义新过程

吉尔在评估了过程风险并运用自己对P&P系统的理解后，对新过程进行以下定义：

- a) P&P 系统将装载盒子。
- b) P&P 系统将从工位 11 拾取部件，并将其插入盒子（每盒放入 20 个部件）。
- c) P&P 系统将目视检查盒子，以确保所有部件放置正确，而且盒内所有插槽都已填满。任何不合格的房子将被自动拒收。
- d) P&P 系统将可接收的盒子放在工位 12 上。【从步骤 b) 到 d) 现在需要 1min。】
- e) 盒子将继续进入其他组装步骤，包括目视检查将确认过程中所有之前的步骤没有畸形出现。

#### 定义软件预期用途

现在吉尔了解自动化的过程并准备为拟采用的新型P&P系统编写目的与意图说明。

——P&P 系统将拣选来自工位 11 的部件并将其放入盒子中。该系统将确认所有盒内插槽已正确填满，拒收任何不正确的盒子，然后在传输线上将盒子以每分钟一个的速率传递到工位 12。

然后，吉尔考虑P&P系统是否会与其他系统接口。结论是没有其他互动。她确定存在用户接口但无软件接口。

#### 确认策划

分析了要实现自动化的业务过程并且确定了新型系统的目的和意图后，吉尔准备在高层次上制定确认计划。后续她还需要添加更多的细节，但如果现在开始进行确认策划，她就能够确定所需确认工作量的级别。

之前吉尔已确定目前的过程剩余风险较低。因此，她感觉确认工作不需要过多细节或形式。吉尔知道为新型系统定义用户业务过程要求和软件需求非常重要。但她指出，这是一个低风险系统，并不认为

需要单独的文件，而且每个文件需要单独签字。因此，吉尔决定将用户业务过程要求、软件需求和测试计划合并成一个文件。

由于新系统的风险很低，吉尔认为不需要对确认工作进行广泛的管理评审，并且由制造经理和质量保证代表批准就已足够了。但是，为了确保用户需求正确，她添加了由过程中的代表性操作员进行的评审。

吉尔用“高品质”公司的确认计划标准格式开始起草确认计划。确认计划的某些部分仍然留空；空白部分将在初始系统设计获得批准后完成。

### 定义系统与软件需求

然后吉尔转向系统与软件需求。她决定软件需求将包括P&P过程或系统步骤以及关P&P系统如何与工位11和工位12连接的接口规范。系统要求包括P&P系统移动的速度和准确性。为了降低伤害的风险，吉尔增加了一项安全要求，在操作员和P&P系统之间提供了一道物理屏障。

### 建立对软件的信任和控制

吉尔宜现在决定她会使用的方法和技术以购买新系统。考虑到业务要求非常简单，交易额会很低。由于新系统风险较低，吉尔决定购买一套第三方P&P系统。出于价格和质量原因，吉尔决定向P&P系统的行业领导者Controlsys公司购买该系统。

Controlsys是一家外部系统供应商。因此，吉尔宜现在决定她将采取哪类活动建立对Controlsys的信任。她评估了自己手里有关Controlsys的信息。吉尔指出，Controlsys的P&P产品使用范围广泛并记录良好。在过去，产品的问题均被快速识别并及时公布在互联网留言板上。对此信息的评审表明，仅存在少数较小的已知问题，并且吉尔确定这些问题与其软件的预期用途无关。此外，Controlsys还提供自动化安装鉴定/运行鉴定/性能鉴定（IQ/OQ/PQ）的测试套件。鉴于该公司的历史同时考虑了P&P系统风险较低的事实，吉尔决定不需要对Controlsys进行现场供应商审核。她批准Controlsys作为供应商。

### 软件失效风险分析

吉尔已确定自动化的业务过程风险较低，但仍需要分析软件失效的风险。吉尔决定使用定量风险模型，对新系统的评分如下：

- 吉尔将“严重度”按1到10分打分为较低的（3），因为软件失效将在下游活动中检测出来。
  - 她将“可能性”打分为很低的（1），因为系统设计非常简单所以测试期间不太可能捕获所有关键错误。
  - 她计算出风险评分为4，对应的风险分类为低风险。
- 因此，吉尔决定执行适合低风险级别的确认任务。

### 完成确认计划

吉尔现在已经定义了软件需求、选择了实施方法并分析了软件风险。因此，她有足够的信息完成确认计划。

由于拟定的系统具有较低剩余风险，因此吉尔为剩余的开发和确认工作选择以下工具：

- 设计、开发和配置工具：
  - 软件架构文档与评审；
  - 可追溯性矩阵（集成并入要求规范中）；
  - 风险控制措施将记录在用户规范中。
- 测试工具：
  - 集成测试（记录在要求规范中）；
  - 接口测试（记录在要求规范中）；
  - 软件系统测试（记录在要求规范中）。
- 部署工具：
  - 用户程序评审；
  - 软件应用的内部培训；

- 供应商提供的测试套件（来自Controlsys）。

#### 维护策划

吉尔现在考虑系统部署完成后哪些活动将适合于确保系统质量。由于系统剩余风险较低，因此她在将运动机构校准加入校准计划时，采纳了制造商的建议。吉尔将系统设置为该公司最长的确认评审周期（3年）。

#### 批判性思维评审

最后，吉尔问自己是否已经考虑了所有必要因素，以确保对自己的确认方法具有正确的信任。结论是，选择并已完成的确认活动提供的信任水平可以接受，即软件将按预期执行。

征求意见稿

参 考 文 献

- [1] GB/T 8566, 信息技术 软件生存周期过程
- [2] GB/T 19000, 质量管理体系 基础和术语
- [3] GB/T XXXXX-20XX/ISO 13485:2016, 医疗器械 质量管理体系 用于法规的要求
- [4] GB/T 20002.4, 标准中特定内容的起草 第4部分: 标准中涉及安全的内容
- [5] YY/T 0316—2016, 医疗器械 风险管理对医疗器械的应用
- [6] YY/T 1406.1-2016 医疗器械软件 第1部分: YY/T 0316 应用于医疗器械软件的指南
- [7] YY/T 0664-2020/IEC 62304:2006/AMD1:2015 medical device software—software life cycle process—Amendment 1
- [8] National Institute of Standards Technology (NIST) Special Publication 500-234, *Reference Information for the software Verification and Validation Process*, Dolores R. Wallace, Laura M. Ippolito, Barbara Cuthill, March 19, 1996.
- [9] Software Engineering Institute, *Capability Maturity Model Integration (CMMI)*
- [10] Pressman R. *Software Engineering, A Practitioner's Approach*, McGraw-Hill, Inc. Third Edition, 1992
- [11] *Principles of Medical Devices Classification*, GHTF/SG1/N77, 2012

