|  |  |  |
| --- | --- | --- |
|  |  | **ISO/TR 80002-2** |
|  |  | 第一版  2017年06月 |
|  |  |  |

医疗器械软件——

第2部分：医疗器械质量体系软件的确认

（标准草案）

|  |
| --- |
|  |

目录

[前 言 **错误!未定义书签。**](#_Toc525677618)

[引言 3](#_Toc525677619)

[1 范围 4](#_Toc525677620)

[2 规范性应用文件 4](#_Toc525677621)

[3 术语和定义 4](#_Toc525677622)

[4 软件确认探讨 4](#_Toc525677623)

[4.1 定义 4](#_Toc525677624)

[4.2信心建立活动：工具箱内的工具 4](#_Toc525677625)

[4.3批判性思维 5](#_Toc525677626)

[5软件确认与批判性思维 5](#_Toc525677627)

[5.1 概述 5](#_Toc525677628)

[5.2确定软件是否外在范围内 9](#_Toc525677629)

[5.2.1记录软件过程和使用的高级定义 9](#_Toc525677630)

[5.2.2监管使用评估 9](#_Toc525677631)

[5.2.3与医疗器械监管要求无关的过程和软件 9](#_Toc525677632)

[5.3开发阶段 10](#_Toc525677633)

[5.3.1制定确认计划 10](#_Toc525677634)

[5.3.2定义 10](#_Toc525677635)

[5.3.3实施、测试和部署 14](#_Toc525677636)

[5.4维护阶段 15](#_Toc525677637)

[5.4.1进入维护阶段 15](#_Toc525677638)

[5.4.2维护安排 16](#_Toc525677639)

[5.4.3维护阶段内的维护类型 16](#_Toc525677640)

[5.4.4过程变化：改变风险控制措施 16](#_Toc525677641)

[5.4.5应急变化 17](#_Toc525677642)

[5.4.6维护预期用途 17](#_Toc525677643)

[5.5 退役阶段 18](#_Toc525677644)

[6文档 18](#_Toc525677645)

[7前提过程 19](#_Toc525677646)

[附件A （资料性附件） 工具箱 20](#_Toc525677647)

[附件B （资料性附件） 风险管理和基于风险的方法 26](#_Toc525677648)

[附件C （资料性附件） 示例 30](#_Toc525677649)

# 引言

本文档旨在帮助读者采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所使用的进行过程软件确认的适当活动。

其中包括质量管理体系中所使用的软件、生产和提供服务过程中所使用的软件，以及ISO 13485:2016标准：第4.1.6、7.5.6和第7.6款所要求的用于监测和测量的软件。

本文件汇集了医疗器械行业此类软件确认人员以及可审计文档专职建立人员的经验。文档开发过程中脑子里始终想着在面对医疗器械质量体系所用软件确认过程时我们都会经历的特定问题和疑问，如：必须完成什么工作？需要多少钱？风险分析如何进行？经过深入讨论，得出的结论是：在每个示例中，确定了一组活动（即一个工具箱里的多个工具），对软件按照预期用途运行的能力提供一定的信心，而活动清单由于软件的复杂性、所涉及的伤害风险以及供应商所提供软件的谱系（例如：质量、稳定性）等因素而各不相同。

本文件旨在帮助包括厂家、审核员和监管机构在内的利益相关方理解和运用ISO 13485:2016标准第4.1.6、7.5.6和7.6款中所包含的软件确认要求。

**技术报告 ISO/TR 80002-2:2017(E)**

**医疗器械软件**

第2部分：

**医疗器械质量体系确认软件**

# 1 范围

本文件适用于设备设计、测试、组件验收、制造、标签粘贴、包装、分发以及投诉处理中所使用的任何软件，或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。

适用于：

* 质量管理体系中所使用的软件；
* 生产和服务提供过程中所使用的软件；以及
* 用于要求监测和测量的软件。

不适用于：

* 作为医疗器械组件、部件或附件的软件，或
* 本身为医疗器械的软件。

# 2 规范性应用文件

本文件内没有规范性引用文件。

# 3 术语和定义

就本文件而言，ISO 9000标准和ISO 13485标准中给出的术语和定义适用。ISO和IEC在以下地址维护用于标准化的术语数据库：

* ISO在线浏览平台：<http://www.iso.org/obp/>。

# 4 软件确认探讨

## 4.1 定义

“软件确认”这一术语的解释，即有广泛含义，亦有狭义含义，范围可仅是测试，亦可指包括测试在内的广泛活动。本文件使用的术语——“软件确认”——表示建立信心，令人相信软件适合其预期用途，而且性能可靠、值得信赖的一切活动。选定活动，无论是什么活动，均宜确保软件满足其要求和预期目的。

## 4.2信心建立活动：工具箱内的工具

工具箱内的工具（参见表A.1至表A.5）包括软件生命周期内完成的各项活动，可降低风险并且建立信心。

## 4.3批判性思维

本文件提倡采用批判性思维，确定宜执行哪些活动，充分确认特定软件。批判性思维是分析、评估软件各个方面及其使用环境的过程，从而确定确认过程中要运用的最有意义的信心建立活动。批判性思维防止一刀切，避免在未彻底评估解决方案、确定其是否确实产生了预期结果的情况下，采用适用于一切确认解决方案的方法。批判性思维承认确认解决方案在软件之间可能存在很大差异，允许在类似情况下，将不同确认解决方案应用于同一软件。批判性思维挑战确认解决方案的提案，以确保其满足质量管理体系的预期意图，同时考虑所有关键利益相关方及其需求。批判性思维也在软件特征发生变化、软件预期用途发生变化或新信息可用时，用来重新评估、确认解决方案。

批判性思维得到的确认解决方案可为制造商建立合规性，确保软件使用安全。批判性思维得到的是审核人员认为既合适又充分的书面证据。批判性思维使开展确认工作的个人感觉工作使其增值，其努力代表着达到预期结果的最有效方式。

附件C给出的示例研究，展示了批判性思维在各种情况下医疗器械质量体系所用软件的确认情况，包括不同复杂程度、不同谱系和不同风险等级。

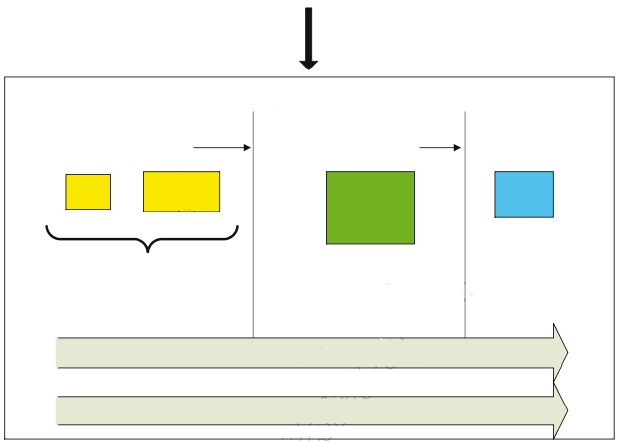
# 5软件确认与批判性思维

## 5.1 概述

在医疗器械质量体系软件的整个生命周期，需要采取适当的控制措施，确保软件按预期运行。融入批判性思维并运用选定活动建立信心，使软件有效状态的建立和维护得到实现。图1为典型活动与控制措施的方案视图。图中的典型活动和控制措施从做出决定那一刻开始，到某个过程实现自动化、直至软件退役或不再是医疗器械质量体系生命周期的一部分为止。虽然图1描绘的是一个序变模型，实际上，此过程在元素定义、风险识别和批判性思维应用过程中都具有迭代性。

在开发用于医疗器械质量体系的软件时，从工具箱中选择一种建立信息的基本活动就是选择软件开发生命周期模型。所选模型宜包含批判性思维活动，以便在各种生命周期活动开展期间，选择其他合适的工具。所采用的分析评估结果是，选出一系列最重要的信心建立活动的推动因素，确保软件按预期运行。本文件并无暗示或规定使用任何特定软件开发模型的意图。然而，为了简便起见，本文件的其余部分解释了批判性思维的概念。各个阶段均采用通用名称，均以瀑布式开发模型为背景。其他软件模型（例如：迭代、螺旋等）只要吸收了批判性思维和合适的工具，均能使用。

**医疗器械质量体系软件**



**开发**

**退役**

**维护**

**迭代风险分析**

**迭代变更管理**

保持确认状态

建立电子记录长期访问通道

最终用途

纠正、适应性、完善性维护

实施/测试/使用

**生命周期控制**

软件确认  
（建立确认状态）

定义

**图1——生命周期控制**

在考虑在某一过程使用软件时，宜通过调查其预期用途，确定所推荐软件是否是某医疗器械质量体系过程的一部分。如果是，则宜确认该软件的预期用途。尽管本文档描述了医疗器械质量体系软件的确认方法，但同样的方法也是软件评估其是否满足规定要求的良好实践。软件确认最关键的部分是开发/购买正确的软件工具，以便能够按照厂家的意图提供过程支持。这意味着需要宜准确确定，从而评估开发/购买的软件是否适于满足预期用途的要求。适用于确认的技术要求和适合于确认的工艺要求同样重要。在考虑将软件应用于某一过程时，该软件能够与其他软件交互，或能够为其他软件提供接口。

在生命周期的开发阶段，执行风险管理和确认计划制定任务，以收集信息，并驱动软件在以下四个方面做出决策：

* 所采用的工作量以及对文件记录和可交付成果的审查；
* 文件记录与可交付成果的内容范围；
* 工具箱工具的选择和工具应用方法；
* 工具应用方面的工作量。

这四个领域决策的主要驱动因素即是过程风险，也是软件风险。但是，其他驱动因素也可能影响决策，包括软件和过程的复杂性、软件类型和软件谱系。

确认计划制定过程包含两个不同的元素。第一个确认计划制定元素涉及确定文档记录的严格程度以及审查由此产生的可交付成果要采用的审查程度。此要素的决策主要由过程风险分析结果驱动。第二个确认计划制定元素驱动的是从工具箱中选择工具，以实施、测试和部署该软件。工具选择主要由软件风险分析驱动。此类计划步骤源于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，很多时候将这些步骤合并为一个活动，其中包括风险分析的不同方面以及由此产生的进行确认的选择。

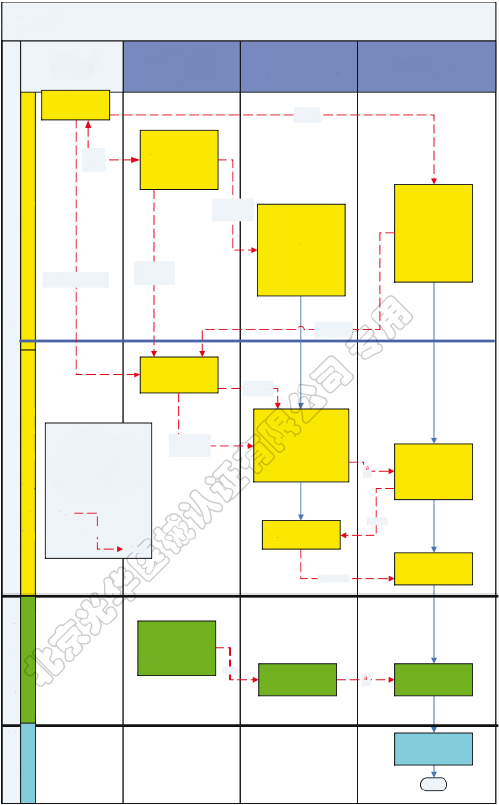
在生命周期的开发阶段，风险管理和确认计划制定任务用于确定应用于软件的适当工作量，并确定建立信心使用哪些工具。这种方法可以完成适当的增值活动和确认任务。这是建立某种经确认状态的基础。这些活动和任务完成后，工具及其相关结果将即刻在确认报告中引用，作为对软件确认结论的支持。

部署完成后，软件将进入软件生命周期的维护阶段。在此期间，软件将按照业务需求或法规要求变化下达的指令进行监控、改善和更新。变更控制活动采用的概念与在生命周期的开发阶段所采用的初始方法的概念相同。然而现在评估变化针对的是变化在初始开发期间对预期用途、失败风险、风险控制措施的影响以及对软件本身任何功能的影响。

退役阶段是通过删除过程或替换过程正在使用的软件来去除软件的行为。

图1中显示的活动反映了软件生命周期主要控制活动。其他工作流包括项目管理、过程开发、供应商管理（若适用）以及可能的其他工作流，具体取决于正在实施的软件。

图2描述了其他工作流活动中的软件生命周期控制活动和批判性思维。关键的思考活动出现在迭代风险分析和确认工作流中。在组织的业务模型中对这些工作流进行清晰、正式的定义非常重要，以确保某个程序从业务和监管两个角度妥善管理软件。



**确认**

**流程**

**迭代风险分析**

**确认计划与报告**

**软件系统**

**开发**

**定义**

**实施、测试、部署**

**维护**

**退役**

定义流程要求

分析流程故障风险

**确认计划**

定义软件预期用途

分析软件故障风险

**确认计划**

软件实施（设计、开发、建立及测试）

确认报告

软件发布

软件维护

维护确认计划

**图例**

a：包括风险控制措施，如代码审查等活动和监视时钟等在设计活动。还包括目标区域的测试方向和要使用的测试类型。

输入

信息

产量

分析对现有风险和/或引入新风险影响的变化

软件退役

结束

流程定义

下游控制

伤害的可能性和严重性

伤害的可能性和严重性

伤害的可能性和严重性

下游控制确认

软件要求

要控制的风险

结果

结果

验收

注意：在使用术语“develop”或“development”时，“develop”或“development”系指开发软件某一经过确认的状态。

**图2——生命周期控制工作流**

图2中描绘的各种颜色对应于图1中整体方法过程图中的生命周期部分。红色虚线表示某一个活动输出的信息，而该信息为另一活动提供输入，或在另一活动中有助于推动决策。该图展示了在完成需要输入的活动之前，获取输入信息的需要如何推动活动排顺。值得注意的是，所有活动都会完成，不受正在实施的软件的大小或复杂程度的影响。但是，对于更大或更复杂的软件，此类活动很可能是分离的；而对于更小或更简单的软件，许多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，用于在各种工作流中识别和包含适当的信心建立活动或工具，以支持软件在发布时得到确认，并且在软件退役之前将保持确认状态的结论。

以下子条款提供了图1中描述的生命周期控件中的每个方块的其他详细信息。子条款使用图2中所示的迭代风险分析、确认和软件活动的工作流描述，提供包含批判性思维的各种决策点以及各种决策驱动因素的透视图。

## 5.2确定软件是否外在范围内

### 5.2.1记录软件过程和使用的高级定义

确定是否认为软件用于医疗器械质量体系的第一步是记录该软件的过程和使用的高级定义。在很容易知道软件处在范围内并且有人已经在着手定义软件详尽的预期用途时，此活动可能看起来价值很小。而在此类假设不太清楚的情况下，记录过程和使用情况可以明确确定该软件是否在范围内。此外，对于已确认超出范围的软件，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软件是否是一款“医疗器械质量体系软件”，因此属于本文件的范围。首先确定适用于软件使用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。

1. 软件的故障或潜在缺陷是否影响医疗器械的安全或质量？
2. 软件是否自动执行或按照法规要求执行某项活动（特别是医疗器械质量管理体系的要求）？示例可包括捕获电子签名和/或记录、维护产品的可追溯性、执行和捕获测试结果、维护CAPA、不合格、投诉、校准等数据日志。

任何问题回答“是”都确定了软件需要确认，并且在本文件的范围内。

有时，确定某一过程以及相应的软件是否是质量体系的一部分可能很困难。有些工具能够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缘软件的周围环境，并宜完全了解软件故障对过程的影响，并最终了解软件故障对任何人造医疗器械安全性和有效性的影响。如果答案不确定，最好的办法是将把该软件视为处在范围内，并采用本文件中所定义的方法。

### 5.2.3与医疗器械监管要求无关的过程和软件

当过程或软件包含超出医疗器械法规要求的功能时，宜进行分析，确定认为该软件的哪些部分处在范围内，哪些部分不在范围内。应根据软件各个组件、模块和数据结构之间的集成程度以及组织的合规性需求，使这些决策合理化。对于用于支持质量体系的软件（例如：复杂企业资源规划（ERP）的大型软件），这种合理化尤其重要。ERP软件能够包括非医疗器械监管过程（如：会计和财务等）的功能。但是，这种功能对于商业运营能够至关重要，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制定确认计划

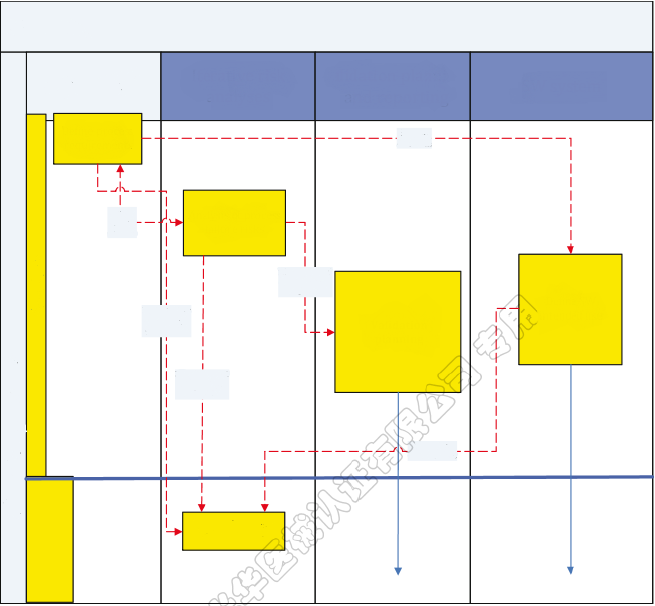
在应用批判性思维时捕获的确认计划制定活动的第一部分，使用过程风险分析的输入（见附件B）确定宜用于文件编制的工作量的基础，并驱使软件从工具箱的“定义”部分选择工具（见表A.1至表A.5）。第二部分使用软件风险分析的输入，驱使软件从工具箱中选择实施、测试和部署工具。执行完毕，软件活动和确认状态随即建立，而且确认的证据在确认报告即刻形成。

许多开发生命周期模型能够在开发阶段应用。本文件没有提倡或推荐任何一个模型；但是，要求采用某种受控方法。这种受控方法将以实施、测试和部署之前定义要求（包括预期用途）的概念为基础。这对于确定软件预期用途确认至关重要。

### 5.3.2定义

**5.3.2.1定义方块要求**

在定义块内完成的活动包括过程的定义、该过程内软件预期用途的定义，以及基于该过程内所确定的固有风险规划确认工作量。图3描绘了选定瀑布模型示例中开发阶段的这一部分。

****

**确认**

**流程**

**迭代风险分析**

**确认计划与报告**

**软件系统**

**开发**

**定义**

定义流程要求

分析流程故障风险

**制定确认计划**

定义软件预期用途

分析软件故障风险

流程定义

下游控制

伤害的可能性和严重性

伤害的可能性和严重性

下游控制确认

软件要求

**图3——生命周期阶段：定义方块工作流**

**5.3.2.2过程要求**

应用生命周期控制的第一步是确定整个过程的目的和作用，特别是要由软件控制的部分。最好的实现方法是让适当的主题专家参与，并要包括所有相关方面和活动，无论是否全部受软件控制。好处如下：

* 能够清楚地了解监管要求；
* 能够清楚地了解在该过程的范围内特定软件的预期用途；
* 能够通过程序或其他方式清楚地识别和处理不受特定软件控制的过程问题和活动;
* 进出所识别软件的过程活动得到识别，并能在软件故障风险评估过程中和找出软件故障风险控制办法的过程中予以考虑。

过程定义活动为后期在生命周期做出决策奠定了基础，对于把精力中在增值、基于风险的活动上面至关重要。

**5.3.2.3过程故障风险分析**

软件与医疗产品的最终安全性和有效性之间的关系将在风险分析过程考虑。还宜考虑以下内容。

* 对人类有害的风险：这包括对患者和用户的直接伤害以及当软件控制制造或设备质量出现故障时的间接伤害，从而导致设备故障，从而造成伤害。
* 监管风险：如果软件故障可能导致监管机构要求的记录丢失（例如CAPA，投诉，设备主记录或设备历史记录）或偏离质量，则应考虑不遵守监管要求的风险系统和制造程序。
* 环境风险：软件运行环境的风险。物理和虚拟。

其他类型的风险能够纳入本模型，但本文件的范围和为降低风险而讨论的工具并不针对这些风险。本文件的重点是在过程故障的范围内，确定与软件故障相关的人身安全风险、监管风险和环境风险。

风险分析的结果宜清楚地记录，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合理的有价值的决策驱动因素。

**5.3.2.4制定确认计划**

确认程度和客观证据的范围必须确保软件要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计划制定活动以过程故障风险分析输入为唯一根据。

该确认计划制定活动产生确认计划文档的第一次迭代。计划包括选择“工作量”（即决策）以及做出这些选择的理由（即决策驱动因素）。理由宜以某过程故障造成的伤害风险为依据。确认计划制定过程中应提供批判性思维应用于确认计划过程的客观证据。

**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的完整画面及其在过程中的目的。具体来说，预期用途系指描述和解释软件适应整个自动化过程的方式、软件的功能、人们对软件的期望以及人们能够多大程度地依赖软件设计、生产和维护安全医疗器械。预期用途是用于了解与软件使用相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，人物、内容、何时、原因、地点、方式等），
  + 软件的监管使用，以及
  + 过程内软件的界限或与其他软件和/或用户的界限；
* 软件使用要求。一般而言，随着复杂性和风险的增加，该元素增加更详细的有关软件使用的信息（例如：使用案例、用户要求等）；
* 软件要求。随着复杂性和风险增加到宜向软件实施者提供明确方向的程度，该元素提供更具体、详细的有关软件期望的信息。

预期用途宜由对经验丰富的，精通法规、质量体系和受控过程的人员正式控制和批准。

鉴于我们应确认“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。

以下子条款提供了有关软件预期用途元素的更多详细信息。

**5.3.2.5.2软件目的和意图**

包含涵盖三个要素的信息：软件使用、监管使用和边界定义。

1. 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、地点、地点。问题的答案探讨软件正在满足过程要求的方式。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答案宜纳入既定预期用途定义。

**表1——样题和答案**

|  |  |
| --- | --- |
| **问题** | **答案** |
| 软件解决什么问题？ | 为了进行趋势研究，存在有效、准确地汇集产品缺陷数据的问题。 |
| 软件为什么有用？ | 软件能够从全球的地址汇集数据并研究其趋势。 |
| 软件如何解决问题？ | 软件驱动数据收集过程，并自动汇集和计算趋势信息，或者软件并不驱动该过程，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？ | 该软件在正常工作时间内访问全球的地址（即每周一至周五）。 |
| 注意：这些样题并不详尽。 | |

1. 监管使用

* 在评估监管使用时，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，从而纳入得出这些结论的原因。既然经确定，软件处在范围内，则需确定任何对人类（医疗器械用户除外）或对环境的潜在危害。以下所有问题都使用户的考虑方向指向公共健康、安全性和有效性或电子记录和签名真实性等作为法规要求一部分的要素。
  + 软件的故障或潜在缺陷如何影响医疗器械的安全性或质量？
  + 软件如何自动执行或如何执行法规（特别是医疗器械质量管理系统的要求）规定的某项活动？
  + 软件怎么会对人（医疗器械用户除外）或环境造成伤害？

1. 软件边界

* 定义有待通过软件控制的过程的几个部分（该过程的内部边界）以及定义软件接口的位置（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，多个软件产品单独确认相比，多个软件产品作为一组进行确认，通常效率能更高。还应考虑，各种分组策略能够以何种方式影响正在进行的维护阶段活动的效率。
  + 过程的内部边界
  + 识别软件在过程内部的边界，清楚地确定了要纳入预期用途中的方方面面。软件既能自动化某个整体过程，也能自动化多个活动的一部分，还能充当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故障相关的风险。
  + 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连接时，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都宜纳入批判性思维过程。

**5.3.2.5.3软件使用要求**

软件使用要求包括记录良好且可追溯的为软件目的和意图提供额外细节层的元素。从用户的角度或产品需求的角度来看，这些要求提供了对系统使用场景的深入见解。用户的视角能够以用户需求、使用案例的形式或根据其他以用户为中心的需求进行捕获。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，能够将某个参考文献引入特定器械要求内或纳入某个软件可能影响的生产线的简介内。

**5.3.2.5.4软件要求**

由元素定义活动组成。这些元素记录良好，具有可跟踪性，用于指定软件为了满足其目的、意图和使用要求需要执行的操作。软件要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实施、测试和部署

**5.3.3.1所需活动**

在实施、测试和部署方块中完成的活动包括：

a）规划设计中的确认严格程度，

b）开发和配置，

c）构建软件，和

d）测试基于确定风险的软件。

**5.3.3.2软件故障风险分析**

软件故障风险分析的关键点是确定并记录与软件故障相关的租赁风险，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措施）。然后使用该分析得出真现有效的确认方法。

在审查可归因于软件故障的风险时，考虑构成风险控制措施的分析软件之外的任何过程控制。这种风险控制措施能够减少软件故障的影响，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，确保软件的安全运行。纳入这些考虑因素将有助于确保在整个过程的背景下查看软件。

附件B中提出的模型并不代表是一个无所不包的公式。由此产生的分析结果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制定确认计划**

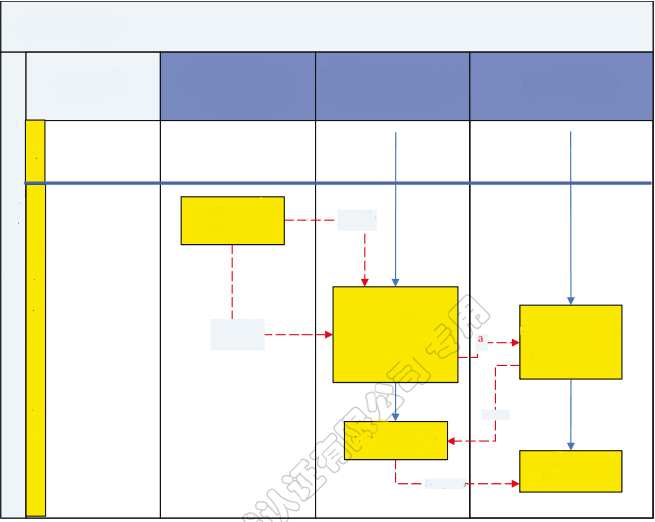
此活动采用预期用途定义和软件风险分析结果作为输入，用于确定风险控制措施以及从工具箱中选择用于确认软件的工具。

工具选择过程由符合条件的个人参与，这一点很重要。符合条件的个人不必是软件专家，但了解故障对过程的影响以及将使该过程自动化的软件的故障的固有风险。来自不同学科（监管、质量、临床等）的个人宜参与任何高度复杂或与其故障相关的高风险软件的规划过程。

制定确认计划生成一个文档化计划，该计划描述选择结果（决策）以及选择原因（决策驱动因素）。制定确认计划提供了信心建立活动选择依据的文档证据，以确保软件以预期的方式运行。

**5.3.3.4软件实施（设计、开发、构建和测试）**

本方块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）。



验收

软件发布

**确认**

**流程**

**迭代风险分析**

**确认计划与报告**

**软件系统**

**开发**

**实施、测试、部署**

分析软件故障风险

**确认计划**

软件实施（设计、开发、建立及测试）

确认报告

伤害的可能性和严重性

要控制的风险

结果

1. 包括代码审查等作为活动的风险控制措施以及监督计时器等设计内容。另外还包括目标区域的测试方向和要采用的测试类型。

**图4——生命周期阶段：实施、测试和部署方块工作流**

**5.3.3.5确认报告**

足够的信心建立活动一旦完成（包括选自工具箱的工具），以确保软件按预期实施，活动和（有可能）活动结果宜在最终确认报告中引用。报告的正式审查和批准为所有已记录的客观证据提供一份参考摘要，而这些证据支持软件已针对其预期用途完成确认的结论。

**5.3.3.6软件发布**

确认结论认为，宜存在一种用于发布软件的正式受控方法。定义控制措施应确保并确认：投入使用的软件与已通过确认报告所引用信心建立活动评估的软件相匹配。否则，理论基础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

阶段入口标准：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各类变化的管理和控制的同时，确保软件保持在经过确认的状态。有些变化类型只涉及软件使用所处过程的更化。

对任何已确认系统的更改宜根据政策和程序以受控方式进行。

理想情况下，建议更改在测试环境下进行，并在将系统升级投入生产使用之前进行确认。如果在测试环境下的变化无法确认，而且变化测试宜在生产环境中进行，则应采取适当的控制措施，最大限度地减少对生产环境或直接对产品产生的不良影响。

确认变更选用工具箱中的哪些工具，宜通过分析对现有风险控制措施的影响或通过引入新风险确定；或由两者共同确定。

由于软件或其配置的实际使用能够随着时间变化（尽管在竭尽全力对其进行控制），使用维护阶段专用工具（例如：定期监控实际使用情况，或实时监控软件配置）或许合适。如果预期用途的变化导致风险级别更高，则该变化能够引起范围比原来执行的确认活动更大的一系列确认活动，即使软件毫无变化，也会如此。

关于开展范围更加广泛的确认活动的决定，宜作为确认计划的一部分记录在案，以提供证据，证明软件保持在确认状态。

### 5.4.2维护安排

应在开始维护阶段之前记录维护计划证据。

理想情况下，维护计划在开发阶段开始安排。宜正确理解变更影响软件确认的方式，检查变更对风险的影响，并规划适当的确认维护活动。

大型复杂的软件或许必须在不影响软件预期执行能力的情况下适应日常维护和性能调整活动。在开发阶段安排维护计划能够在不影响确认的情况下完成哪些操作活动，哪些更改需要确认工作。在软件到达维护阶段之前，宜计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）的变化影响已确认软件的方式。培训软件操作员识别这些边界并识别正常操作活动与需要确认的任何更改之间的差异是有好处的。

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。矩阵反映测试或其他确认活动的要求、反映风险控制措施等等。如果初始实施期间可追溯性分析进行良好，可追溯性分析将通过促进变更影响的识别以及促进适当变更确认活动的识别成为维护期有价值的工具。在简单软件中，此类分析能够包含对实施和确认的单级需求跟踪。但是，复杂软件可能需要多级矩阵，将顶级功能分解为较低级别的需求，然后分解为实施和确认。也可以嵌入其他信息，例如，能够在跟踪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。其中较常见的维护变化类型包括：

* 做出纠正维护变化，以纠正软件中的错误和故障；
* 完善维护更改，以增强性能、可维护性或其他软件属性；
* 为更新软件操作环境而进行的自适应维护（例如，操作系统变化、系统硬件变化或与软件连接的其他应用程序的更化）。

### 5.4.4过程变化：改变风险控制措施

当由软件完全或部分控制的过程发生变化时，宜进行影响分析，重新评估风险控制措施。

软件完全或部分控制的过程能够独立于软件变化。当某一过程发生变化时，了解该变化对软件确认状态的影响非常重要。过程变化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。

过程变化还能影响作为确认基本原理的软件一部分的现用风险控制措施。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，则建议过程变化影响分析更容易进行。对于软件和软件运行过程而言，以建立信心的方式进行维护，是影响分析的关键。

### 5.4.5应急变化

应紧变化宜根据批准程序进行调整。这些程序宜要求提供开发和实施的正当理由，要求说明获得和记录变化部署的授权机制，要求说明确保风险得到适当评估和控制的规定，以及调用应急变化所需的任何活动（如：培训、沟通、产品评审和处理等）。在此情况下，执行规定，适当评估和控制风险，就表示需要在发布之前，满足变化确认监管要求所需开展的最小范围的活动。

紧急情况下，可能需要进行软件变更。通常，如果软件操作系统的完整性或数据的完整性受到损害，或为了减轻潜在有害情况，需要进行此类更改。

此外，可能还需要的在应紧变化发生后开展一系列活动，以充分评估变化带来的所有影响。依据某过程故障带来的总体风险，过程输出（数据或产品）可能需要实施额外控制，直到应变化后的所有活动全部完成为止。

造成进程中断软件问题通常很明显，而找出细微、隐蔽的问题比较困难。定期评估错误日志、定期评估帮助中心的请求、客户反馈和其他缺陷报告，可以指出隐蔽的问题。此类监控技术发现的问题，不足以显示在错误报告中，但能够指出可纠正的软件问题。因此，可能需要开展维护活动，通过对未来发布软件实施更正，处理识别出的问题。此外，已发布软件中归因于此类软件问题的问题能够主动管理。

在利用维护活动纠正未来发布软件的问题后，宜检查已发布软件中已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正确使用，定期评估用户培训效果则是另一种有助于维持确认状态的监控技术。

### 5.4.6维护预期用途

如果软件的预期用途发生变化，则宜确认新的预期用途，或终止新用途。对于后者，风险评估是为了确保在未授权使用期间不会引入任何风险。

预期用途的变化是需要特别注意的类别，因为变化可能很细微，而且难以发现，也可能非常明显。细微变化的变化对象是目的和意图或软件使用要求，并不一定导致详细的软件需求元素发生变化（见5.3.2.5）。这种变化可能是有意为之，也可能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慢慢变化，或者用户可能会以最初未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。

每次要对已确认的软件进行更改，都宜重新审查预期用途，确保预期用途与软件的实际用途一致。

## 5.5 退役阶段

在退役阶段，宜记录软件的退役情况，并建立可以在任何必须的记录保留期内访问任何相关电子记录的方法。

软件退役活动高度取决于要退役软件的类型。有些软件只实施某种活动而不存储任何数据。其他软件则可以像批量可追溯性或文档控制系统一样复杂，其中包含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，宜存在一份用于处理数据的计划。需要考虑的一些问题包括以下内容：

* 是否有软件取代退役软件？
* 数据是否能够转移到新软件里？
* 数据是否宜转移成便携格式，以便长期保留？
* 此类数据的数据保留要求是什么？
* 数据是否会存储在耐用媒体上？
* 如果数据会存储在耐用媒体上，那么，存储指令或存储程序是什么？可以检索包含所有相关数据要求的数据吗？
* 可读耐用媒体和软件的维护程序是什么？
* 是否会存储硬件平台档案，用来使用和检索已退役的应用程序？
* 存储硬件如何维护？
* 是否需要作为投诉或CAPA调查的一部分访问退役软件？
* 是否需要平台和应用程序重新创建软件程序？

# 6文档

文档编制宜确保与软件生命周期控制活动相关的所有信息均得到适当记录和控制。

具有优质高效的文件编制能力可以带来两大好处：

1. 文档记录中清晰明确地表达完整的软件定义，可以充分了解软件的预期用途和预期性能，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。
2. 记录确认计划和记录确认实施情况，为利用批判性思维所做决策提供了书面证据。编制这种文档围绕所执行的评估或分析以及因此发生的针对基于风险的有意义的信心建立活动的工具选择，可以提供关于已执行确认的简洁知识。文档包含验收标准满足情况的概要，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。

所产生的文档范围与所应用的软件确认工作量直接相关。工作量宜与风险相称。因此，本文件中讨论的软件确认方法，根据过程故障的影响，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，文件的预期范围就越大。此外，较高的伤害风险，宜推动多个跨功能同行或更高层次的管理层或两者一起开展更高级别的文件审查。

如何将生命周期控制信息组织到文档中以决于许多因素，例如，所使用的技术以及软件的大小或复杂性。

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

生命周期控制信息的捕获、记录方式，取决于确认实施各方的偏好和既定政策。有关如何将生命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。从合规性审查角度看，宜建立确认计划和报告文档，提供一大汇编文件，将曾经计划和执行过的所有增值、信心建立活动汇集在一起，以确保软件按预期执行。从本质上讲，此文档是基于输入（决策驱动因素）所做选择（决策）的关键记录。输入体现了确认所采用的批判性思维过程，表明符合法规意图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”用于指被记录的信息主体，不管信息主题是否是记录在要求管理工具等捕获信息工具内的实际文档。

# 7前提过程

本文件中介绍的方法旨在完全在有效质量管理体系内运作，以提高效率。

质量体系能够对批判性思维方法的成功产生最积极的影响。受影响的内容包括：资产与基础设施管理（人和硬件）、变更管理（包括配置管理）和供应商管理。详述介绍这些内容不在本文件的范围内；其中每项内容业内其他标准和文件中都有介绍（参见参考文献）。此外，本文件无意将特定作用或功能（例如：质量保证、管理和制造）与本文件中的活动关联在一起。执行确认活动的满意人选由各公司的理念和人力资源基础设施决定。

# 附件A （资料性附件） 工具箱

**A.l综述**

工具箱提供了一系列信心建立活动，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，但提供了基于当前软件工程知识机构的启动程序集。其中一些活动相互重叠或协同工作，例如，正常的案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。这些活动将充当制定确认计划和执行该计划的基础。

活动的选择和进行宜适合软件所带来的风险。为了支持此选择，工具箱内的活动按照以下体系分类和标记。

* 全范围：无论如何此活动都要进行。
* 量身定制：选择并执行此活动的相应部分。
* 选择：适当的时候选择并开展活动。

工具箱能够量身定制，定义贵组织使用的活动。随着时间的推移而发展，工具箱还能够随着技术的变化和经验教训的积累进化，从而整合新的软件工程最佳实践。如适用，有些活动也会在标准程序中以程序性方式调出。

**A.2工具箱结构**

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

列表中出现的每个命名活动都有简短的定义以及关于该活动价值（即该活动有助于确认工作）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** |
| 过程要求定义  （全范围） | 通过软件、某个制造过程或某个质量体系过程定义正在考虑的部分或全部自动化过程的活动。活动还描述了在过程执行或软件风险分析时能够考虑的过程内的任何确认或预防措施。  此活动的输出能记录在流程示意图或要求说明书中，这些说明书定义了在业务、制造或质量体系流程中实施的活动。 |
| 过程失败风险分析  （全范围） | 确定过程故障对设备安全性和功效、制造人员、环境或质量体系的影响的活动。 |
| 预期用途  （量身定制） | 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档的扩充文档，或许还可包括详细的软件要求。风险也是决定预期用途定义深度的重要因素。  预期用途的要素：   * 软件目的和意图； * 软件使用要求； * 软件要求。 |
| 制定确认计划  （全范围） | 制定确认计划分两个阶段进行   * 在开发—定义阶段，确定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，在实施阶段，根据定义阶段和相关风险分析活动中做出的决策，选择适当的确认活动。   制定确认计划输出的是一个计划，描述了为确保软件始终满足其预期用途要求而进行的信心建立活动。 |
| 软件需求正式审查  （选择） | 利益相关方根据预期用途审核软件要求并就软件要求达成一致的活动（过程、会议等）。 |
| 软件开发生命周期模型选择  （选择） | 定义在软件整个生命周期开发部分时段所使用生命周期方法和控制的活动。通常仅复杂软件或有风险的软件需要此活动。IEC 62304:2006/ AMD1:2015可能特别适合作为某些软件的过程标准。 |
| 风险管理计划  （全范围） | 与规划执行软件风险管理相关方式的活动。风险管理计划的输出是一个计划，用于定义分析软件相关风险关注领域的方法，以及选择风险分析方法（故障模式与影响分析（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  （全范围） | 此活动是识别风险控制措施或危害控制措施的机制（例如：程序控制）。包括持续监控，确保控件到位并正常运行。 |

**表A.2——开发阶段：实施**

|  |  |
| --- | --- |
| **活动** | **定义** |
| 软件故障分析  （风险分析）  （全范围） | 软件故障分析系指确定软件故障相对于过程的影响以及确定软件故障相对于过程故障分析所确定关注领域的影响。 |
| 软件架构文档与审核  （选择） | 软件架构确定软件元素的高级结构及其相互关系，同时记录架构，并审核软件功能实施的正确性、完整性以及软件功能实施能力。 |
| 设计规范  （选择） | 设计规范系指软件要求实现方式的精确声明，通常包括软件或组件的结构、算法、控制逻辑、数据结构、数据集使用信息、输入输出格式、接口描述等等。 |
| 开发设计审核  （选择） | 开发设计评核系指为评估某个或多个配置项的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  （全范围） | 本活动确定控制风险评估过程中发现的风险或危害的控制措施。确定风险控制措施宜允许持续监控并确保控制措施到位且正常运行的迭代过程（如：程序控制、硬件冗余）。 |
| 代码审核或代码确认  （选择） | 代码审核或代码确认包括对软件源代码一次同行评审，旨在查找和排除缺陷，提高整体代码质量。通过建立和遵守一组通用编码标准，能够增强代码审核质量和整体代码质量。 |
| 可追溯性分析  （选择） | 可追溯性分析系指对设计要求、代码要求、测试要求、风险或危害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  （选择） | 供应商审计系指对达到必要级别的软件供应商体系进行评估，使买方确信供应商能够充分提供安全可用的软件。各种各样的供应商审计方法都可能采用。 |

**表A.3——开发阶段：测试**

|  |  |
| --- | --- |
| **活动** | **定义** |
| 测试计划  （选择） | 测试计划宜确定测试活动的整体方法，以帮助建立软件满足其预期用途的信心。但是，软件测试本身可能不足以确定软件适合其预期用途的信心。其他确认技术可能需要与测试相结合，确保确认方法综合、全面。  测试级别宜基于风险驱动因素和风险因素，并宜提供适当的信心，证明软件符合相应测试方法的要求和设计规范。这种测试能够包括开发人员测试、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  （选择） | 进行测试，以确认某个软件元素（如：某个单元或模块）或软件元素集合设计的实现情况。 |
| 数据确认  （选择） | 数据确认系指为确认数据的正确性而完成的活动。数据确认可以作为数据转移、转换或测试工作的一部分或独立完成，数据确认亦能包括（若适当）统计抽样。 |
| 集成测试  （选择） | 集成测试系指测试的有序进展，期间，软件元素、硬件元素或软硬件元素彼此结合并接受测试，以评估其交互作用，直至软件集成完成。 |

**表A.3**（续）

|  |  |
| --- | --- |
| **活动** | **定义** |
| 用例测试  （选择） | 用例测试是一种功能测试形式。用例测试忽略系统或组件的内部机制或结构，侧重于响应选定输入和执行条件生成的输出。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某个序列的预定流能够连接一系列用例。 |
| 接口测试  （选择） | 接口测试系指确认软件应用程序之间的接口，同时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数据100％确认完成。测试活动宜包括确保接口在规范界限内或边界条件下按正常情况和异常情况执行的策略。 |
| 回归测试  （选择） | 重新运行某个程序以往已正确执行过的测试用例，以便检测软件开发和维护期间所做的更改或修正所产生的错误。 |
| 供应商提供的测试套件  （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心。但是，宜评估此类套件是否适合即定预期用途和测试的完整性，包括测试任何已采取的风险控制措施。使用这样的套件可能需要一份契约协议，要求供应商在软件的整个生命周期内维护测试套件。 |
| 软件系统测试  （选择） | 软件系统测试系指测试某一集成硬件和软件系统以确认软件是否满足其指定要求的过程。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，因为软件确认确认的是软件在其预期环境下是否适用以及是否适用于其预期用户。软件系统测试仅确认软件要求已成功实施。  对于由软件控制的生产系统，过程确认测试能够涵盖这些测试的一部分或全部。对于质量体系应用程序，执行软件工作指令所要求的所有步骤能够涵盖软件测试要求。 |
| 用例测试  （选择） | 用例测试系指基于用例执行的测试，包括用例中定义的替代流程和错误条件。 |
| 正常情况测试  （选择） | 正常情况测试系指使用常规输入进行测试。 |
| 稳健性测试  （压力测试）  （选择） | 稳健性测试宜证明：某种软件产品在得到意外无效输入时表现正常。稳健性测试用于评估某个系统或组件是否达到或超出了其规定要求的界限。  各种用于识别足够的此类测试用例集的方法，此类用例包括等价类划分、边界值分析和特殊情况识别（错误猜测）等。 |
| 输出强制测试  （选择） | 输出强制测试系指选择测试输入，以确保体系正确生成选定（或所有）输出。  输出强制涉及制作一组测试用例，用于从体系中生成特定输出。重点是创建所需输出，而非关注之前启动系统响应的输入。 |

**表A.3**（续）

|  |  |
| --- | --- |
| **活动** | **定义** |
| 输入测试组合  （选择） | 输入测试组合是一种测试技术。通过该技术，使用某软件单元或体系在运行期间可能遇到的输入的组合。 |
| Beta测试  （选择） | Beta测试系指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  （选择） | 性能测试根据其所需响应时间、中央处理单元（CPU）使用情况以及其他运行中的量化特征衡量软件系统的执行程度。 |

**表A.4——开发阶段：部署**

|  |  |
| --- | --- |
| **活动** | **定义** |
| 用户程序审核  （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进行审查。这样的审查确保正确定义软件的用途。 |
| 申请的内部培训  （选择） | 内部培训系指针对该软件的文档化的培训活动。 |
| 安装确认  （选择） | 安装确认系指使人相信软件已按照文档化安装说明安装和运行。 |
| 运行合格确认与性能确认（执行过程确认时）  （选择） | 运行合格确认使人们相信制造过程和相关系统能够始终如一地在既定限制和公差范围内运行。  性能确认证明确立过程的有效性和可重复性。 |
| 最终验收测试  （选择） | 最终验收测试系指在最终部署之前应用于系统的测试。又称上线测试。 |
| 操作员认证  （选择） | 操作员认证证明已接受培训的人员显示培训合格证据的过程。 |

**表A.5——维护阶段**

|  |  |
| --- | --- |
| **活动** | **定义** |
| 维护计划  （量身定制） | 维护计划相关方法如下：  预先计划。此方法涵盖软件变更的前瞻性规划和预期，能够在进入维护阶段之前在软件初始实施期间使用，但也能在维护阶段的任何时间使用。  规划待定更改。此方法涵盖的是软件某一更改待定时所做计划。规划通常关注针对待定变更的活动。此计划在软件维护阶段完成。 |
| 已知问题分析  （选择） | 已知问题分析系指评估供应商已知的软件任何问题和所有问题及其对已已安装软件的使用或确认状态影响的过程。 |
| 兼容性测试  （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。 |
| 基础架构兼容分析  （选择） | 基础架构兼容性分析是确定软件基础架构变更如何影响已安装软件的过程。更改可能包括变更硬件或改变系统位置。 |

**表A.5**（续）

|  |  |
| --- | --- |
| **活动** | **定义** |
| 系统监控  （选择） | 系统监控包括用于在软件生命周期的维护阶段评估软件系统一般健康状况的技术。系统监控方法能够包括以下内容：   * 定期评估预期用途是否已发生变化； * 最终用户的实际使用状况； * 培训效果评估； * 缺陷分析； * 数据审计。 |
| 备份和恢复进程  （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  （选择） | 除备份与恢复过程、监控和报告之外，还能通过运行控制，帮助确保软件按预期运行。常用方法包括：   * 安全措施； * 访问权限管理； * 数据库管理； * 归档； * 应急计划。 |
| 回归分析  （选择） | 回归分析包括可追溯性分析或影响分析等任务。进行回归分析的目的是为了确定维护系统确认状态所需的活动。 |

# 附件B （资料性附件） 风险管理和基于风险的方法

**B.1综述**

如本文档核心部分所述，确认内容和确认的严谨性取决于与软件相伴的风险。

有关此概念的详细阐述，参考ISO 14971标准。ISO 14971标准描述了一种应用于医疗器械的风险管理过程，但基本原则及术语可以应用于受ISO 14971约束的软件。

**B.2术语**

下定义或取自ISO 14971标准，或基于ISO 14971标准中的定义。

* 危险：伤害的潜在来源；
* 危险状况：人员、财产或环境暴露于一种或多种危险的情况；
* 风险：伤害发生概率与伤害严重程度的结合；
* 伤害：人身伤害或人身健康受损或财产损害或环境损害；
* 严重程度：衡量危险可能造成的后果；
* 风险控制措施：将风险降低或维护在规定水平的措施。

**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，制造商需要使用软件识别可能的危险情况、估测相关风险，并评估这些风险是否符合验收标准。验收标准（如果没有另外给出）例如，依据规定，由制造商界定。

特别是由于软件本身不会受到损害，软件控制的整个过程以风险管理为依据。

**B.4识别危险状况和估测风险**

遵循ISO 14971标准从预期用途开始的方法，宜确定可能的危险和危险状况，并估测相关风险。但是，要考虑的可能危害与ISO 14971标准考虑的内容有很大不同。

生产和质量体系故障很少对患者或医疗器械用户造成直接伤害，医疗器械的制造或质量由软件控制。在这种情况下，伤害几乎总是间接的。最终会成为患者伤害源或医疗器械使用户伤害源的是对器械的伤害。这并不是说间接伤害无论如何都不那么严重。事实上，在某种意义上，可能认为生产和质量体系故障造成的结果更严重，因为生产体系和质量体系中的一次单一故障可能导致许多设备存在故障，最终在检测到之前伤害许多患者。单个器械的软件故障一次只能伤害一位患者。

直接多重伤害和间接多重伤害都可能产生于生产体系故障或质量体系故障。请注意，以下列表内的危害并不相互排斥。每种危害都有可能对患者或医疗器械使用者造成间接伤害。示例包括但不限于以下内容：

* 对医疗器械的损害：
  + 机器不产生临界公差；
  + 标定系统错误标定某一药物输送装置；
  + 灭菌器控制器失效，致使非灭菌组件产生；
  + 最终测试系统存在故障，检测不到潜在设备缺陷；
* 对制造过程的损害：
  + 某软件控制过程的某个故障由于采用手动解决办法而降低生产率；
  + 软件驱动过程的某个故障导致很大比例的超差部件产生；
* 合规性有损：
  + 投诉处理系统误报故障统计数据，从而使现场报告缺陷无法察觉；
  + 设备服务或维修系统未能突出显示以往可能指向未察觉缺陷的问题趋势；
  + 植入设备的某个数据库出现信息篡改；
  + 与制造物品安全检查有关的质量控制记录丢失；
  + 合规数据丢失；
  + 器械确认数据丢失；
  + 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，导致设备安全召回无法通知潜在用户；
* 制造人员伤害或环境损害：
  + 操作员受伤；
  + 释放有毒化学物质。

在分析相关风险时，宜考虑各类损害，取决于使生产体系和质量体系实现自动化的软件。

风险评估包括估测可能伤害的严重程度以及发生该危害的可能性。

严重程度估测通常通过分类来完成（参见例如ISO 14971：2007、附件D或与验收标准相连的G.4（见B.5）。

事实证明，伤害的可能性很难估测，尤其是在查看软件故障造成伤害的可能性的时候。因此，应该记住，软件故障只是导致危害的其中一个因素，可能还涉及软件之外的其他几个因素（事件序列）。为未知事件的可能性设想一种最坏的情况是很有意义，最后是伤害可能性的最坏状况。

IEC 62304:2006/AMD1:2015采用了类似方法。作为决定过程控制严格性的基础，IEC 62304:2006/AMD1:2015假设了软件出现故障的最坏情况，但允许考虑与软件之外的一系列事件相关的较低的损害可能性（另见IEC/TR 80002-1）。

**B.5风险评估**

一旦估计有风险，就需要对风险进行评估，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，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。这种决定很大程度上取决于潜在伤害的严重程度。每个制造商都需要确定定义和记录风险可接受性的标准，确定识别以某种格式存在的所有风险的标准，以使评估符合这些标准。一般而言，如果可接受的风险降低到人可以轻松保护其同行、管理层或审计师的水平，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，但推荐几个可接受性阈值设置过程是可以的。

* 要明确。“尽可能低”或“与任何其他产品一样安全”等验收标准并无意义。验收标准读起来宜像可测试规范，以便可以客观地确定是否已满足可接受性标准。
* 如果难以估计损害的可能性，则验收标准只能基于严重程度。
* 验收标准能够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。
* 尽早确定验收标准。一旦潜在伤害风险确定，立即设定目标或规格。在尝试控制风险之前设置可接受性目标很重要。一旦尝试控制风险，对可接受性感知通常会迁移到更高风险水平。提前记录可接受性标准可以防止迁移。
* 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**

如果风险评估为不可接受，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。这些风险控制措施能够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包括程序变更、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往往难以访问，制造商也很少提供细节。常见的例子是嵌入医疗器械制造机床内的软件。在软件接受独立确认时，确认此类软件的预期用途可能会很困难。

在这些情况下特别有效的风险控制措施是软件输出的下游确认或软件控制器械输出的下游确认。换句话说，通过监控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方法应用生命周期控制方法推断软件对其预期用途的适用性。此方法仅关键操作数量相当少的过程可用，这样，操作能够在每个部件或在统计确定的部件采样上进行检查。确认工程师宜详细说明替代下游确认的理由，详细说明用于决定用抽样确认代替连续确认的任何假设的合理性，然后这些假设宜接受测试。

下游确认宜记录下来，记录方式与任何其他风险控制措施的记录方式相同。特别重要的是要把确认过程记录为一种风险控制措施，这样，确认过程在以后的成本削减措施中就不会消除。此外，还宜记录下游确认结果，因为定义确认需要为确认“提供客观证据”，而且此确认步骤要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变化。举个例子，想想最初只在医疗器械的某个组件上执行一种关键操作的机床。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变化（两种关键安全操作相对于一种关键安全操作），因此，下游确认宜转而确认两种操作。

下游确认能够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确认，而且如果用作该过程的风险控制措施，则宜记录该确认测试。人体测试仪的测试程序宜详细描述，明确界定每个被测试参数的合格结果和不合格结果的范围。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：

* 设计变更，或
* 过程控制。

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

软件之外充分理解的风险控制措施（例如：“下游确认”以及软件设计变更）还是宜在仅依赖于过程控制之前实施。但所采用的过程控制集宜最小，特别是风险控制措施是为了为软件设计变更的正确实施提供信心的时候。

**B.6.4风险控制措施的确认与剩余风险的评估**

宜确认风险控制措施的实施情况。宜确认过程控制之外的风险控制措施的有效性。在这种情况下，宜评估剩余风险的可接受性。

# 附件C （资料性附件） 示例

本文件适用于用于自动化部分质量体系和制造过程的软件，包括准备用于监管提交、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捕获仪器数据、设备操控以及数据处理、报告和存储。对于那些不同的活动，从包含在可编程逻辑控制器（PLC）或个人计算机（PC）中的软件，到包含在具有多种功能的实验室信息管理系统（LIMS）中的软件，能够各不相同。以下是一些预期用途的示例：

* 对产品做出合格/不合格判定的软件；
* 用于在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软件；
* 用于向监管机构报告的数据处理和分析软件；
* 用于监管过程软件的任何软件开发工具或编译器；
* 负责鉴定和确认生命关键软件的任何软件工具或从属软件工具；
* 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）。

本附件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。体验批判性思维方法和了解软件类型、软件风险和预期用途可变性的最佳方式是提供这些示例。

请注意以下资格。

* 这里使用的示例包括本文作者的批判性思维结果，并且代表确认量和严谨性的可接受水平，这会增加价值，提供软件按预期运行的信心。强烈建议读者从工程角度考虑哪些活动和工作量有意义，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。
* 建立确认工作适当性信心的方法始终不止一种。本文介绍的示例提供了一种基于方法的方法，而方法基于的是当前的思想和本文作者的经验。
* 强烈建议读者将作者们的努力视为权威和规范。引用示例采用相似格式的原因仅为展示数据之用，引用示例包括用于证明批判性思维使用的关键思维过程。采用这种布局的目的不将其用作确认模板，这种布局也不包含期望实际确认文档要达到的所有深度和详细程度。
* 所采用的示例假定第6条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，但宜采用这些过程，确保软件和所有相关方面（如文档和其他基础架构等）接受变更控制。
* 每个示例都从明确定义待控制过程开始。因此，事实已经清楚，该过程以及软件均在范围之内。随后确定并总结批判性思维活动。
* 这里使用的示例旨在提供批判性思维过程中使用的决策以及决策驱动因素的信息，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、团队或个人均为虚构，在此使用仅为了便于讨论。

这里使用的示例一般侧重于将特定系统置于确认状态。虽然为系统建立确认状态非常重要，但在系统维护阶段维护该确认状态对于确保软件和周围过程的正确运行同样至关重要。维护活动需要的控制措施和批判性思维与初始确认活动所需的控制和批判性思维相同。

**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。该公司已经收到了软管的技术规格，包括软管形成专有形状的规格要求。软管供应公司将作为其管段制造过程的一部分执行这种特殊管形要求。

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机器执行的流程。厂家决定开发一大具有可编程逻辑控制器的定制设备来执行该任务。该设备及其内包含的PLC宜根据医疗器械公司的政策要求进行有效用途的确认。

**定义流程**

软管供应公司和器械制造商成立了一个团队，定义软管形成流程。会上定义的流程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：

1. 获得材料；
2. 插入机器；
3. 通过压力和热量使管子变形到合适的直径；
4. 等待软管冷却；
5. 从机器内取出软管；
6. 测量软管直径是否合适。

**分析流程风险**

医疗器械制造商已向软管供应公司通报了风险分析过程出现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。
* 表皮问题可能影响客户接受度并导致延误治疗。
* 软管成型过程中操作人员可能被灼毁。

在风险减轻之前，与造成护理人员滑倒、治疗延误和操作人员灼伤危险相关的产品故障风险程度为中等。

目前采取了以下流程风险控制措施：

* 通过进货检查和生产线清理等上游操作，确保软管可以使用；
* 通过下游确认检查（包括：泄漏测试、过程检查和测试配件），减轻设备错误；
* 屏蔽罩、独立温度传感器和冷却液喷雾器安装到位，防止操作人员受伤。

基于此信息，供应商与医疗设备制造商一起得出这样的结论：经过此软管形成流程，软管的残余风险较低。

**确定软件目的和意图**

软管供应公司知道，要确认软件的预期用途，宜首先确定其预期用途。为了就设备的目的达成共识，团队成员会问自己一系列问题，旨在确定系统目的和意图的简洁但可用的定义。团队成员形成了以下评估报告。

* 软件控制设备旨在自动化已定义工序的第2至6个步骤。该系统预期用途是在B厂第3生产线上用于创建PN 001。该系统将用于静脉注射（IV）软管插入、成形、移除和测量的自动化，从而提供常规非危险解决方案。

**制定确认计划**

确认计划的第一步涉及确定可交付成果的严格性和审核。由于残余过程风险确定为较低，因此采用了以下方法：

* 文档的严谨性：
  + 此项目文档的严谨性为中等，即，将会在实施之前，出现可交付成果合并而设计不会转换为详细的设计规范的情况。
* 审查程度：
  + 流程开发和实施负责人员（软管供应公司代表）和独立的质量人员（医疗器械公司代表）可以提供审批。
  + PLC代码和所有规格/设计将置于正式配置管理下，例如，置于某文档控制系统或配置控制系统之下。
* 系统定义：
  + 将会创建流程要求，流程要求要包括系统要求规范，详细说明设备的功能，包括设备的预期输入和输出（例如，整个功能设备的设计控制元素）。
  + 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。此外还会创建各种软件要求，而软件要求要包括逻辑功能流程，这也足以涵盖软件的设计。

**建立对软件的信心和控制力**

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心，相信软件有能力按要求运行的历史记录。但是，通过根据测试协议审核系统功能要求、配置控制以及测试情况，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**

软件能够由于生产线上放出的软管形状不对，导致泄漏并可能导致护理人员滑倒而失败。软件也能发生故障，导致过热，而过热能导致操作员灼伤。软件本身不会给产品带来在过程风险分析中尚未捕获的任何新风险。因此，该小组判定：当前的下游流程宜亦然并足以降低与软件故障相伴的风险。

**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，团队成员宜完成以下确认计划。

* 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要求内捕获。因此，软件要求规范足满足设计目的，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跟踪性矩阵，并将进行可追溯性分析，确保可追溯性的完整性。
* 测试工具：
  + 软件系统测试将基于操作员手册中的软件要求和程序。
  + 如果需要，将进行回归测试。
* 部署工具：
  + 系统操作员和工程师将审核工作说明的清晰性和可用性。
  + 使用设备需要操作员认证。
  + 在确认计划完成和确认计划活动实施后，团队很自然地认为系统将始终如一地提供所需预定输出。

**维护考虑因素**

如果认为此流程的任何部分发生了更改，或者如果软件的预期用途发生了变化，则应进行分析，确定当前的任何缓解措施是否会受到影响，或者该变化是否伴有任何新的风险。此分析包括审查与软管成形设备相关的软件风险。

**工具箱使用**

从工具箱内取用了以下工具。

* 开发—定义阶段：
  + 流程需求定义；
  + 过程故障风险分析；
  + 预期用途；
  + 制定确认计划；
  + 软件需求定义；
  + 确定制造流程内的风险控制措施。
* 开发—实施阶段：
  + 软件故障分析；
  + 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：
  + 用户程序审核；
  + 运营商认证。

**示例2：自动焊接系统**

戴夫是确认新生产线上所有系统的确认团队的一员。他的工作是确认箱盖焊机。他是这个项目成果的项目经理。

**流程描述**

戴夫的团队花了很多时间讨论由谁开发和确认新生产线的哪些部分。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。各部分在确认系统的上游进行测试。

要设置焊机，需要四个步骤：

* 打开机器；
* 确认要运行部分是否存在条形码；
* 从制造执行系统中拉出该零分的程序；
* 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：

1. 开门；
2. 装载各部分；
3. 关门；
4. 启动程序；
5. 将视觉系统指标置于启动点；
6. 打开激光器；
7. 确保运动控制能够移动零件焊缝；
8. 关闭激光器；
9. 开门；
10. 移除部件。

该流程完成后，零件移至戴夫负责范围之外的系统。他知道下游活动包括焊缝渗透的破坏性测试、罐体尺寸的高度检查以及气密密封的泄漏检查。

**定义预期用途**

为了定义软件的预期用途，戴夫收集了信息。他知道视觉精度、移动精度、功率精度和速度精度对于保护操作员安全和实现焊接穿透一致性的过程都很重要。

戴夫首先通过陈述软件的目的和意图定义其预期用途，如下文所示：

* 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激光器。包括上述过程描述中的步骤e）至h）。

**风险分析**

戴夫想消除该流程的人为错误。他知道，激光控制、伺服机构和视觉是此流程的关键组成部分。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，就不会启动此流程。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。戴夫使用来自该流程的信息以及作为焊接流程一部分的设计风险管理活动发生过的信息。他参考了FMEA（故障模式和效应分析）结果，重点关注了三个方面：关键部件参数、气密密封和用户界面。戴夫确定了与此流程相关的多种危害。首先，如果接触激光，操作员可能会被灼伤。与产品相关的是，该流程可能会不正确焊接，从而导致产品泄漏并伤害最终用户。戴夫判定此流程风险很高。

**制定确认计划**

对于这个项目，戴夫查看了工具箱中的定义工具，断定自己需要创建软件需求定义和维护文档。自己的软件要求宜包括工具配置参数以及激光时间和功率调整。自己还需要将软件定义到硬件接口上。具体而言，戴夫包括视觉系统的精度要求、激光时间与功率范围、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。

戴夫还断定，自己需要进行正式的软件需求审核，这会涉及自动化工程师、制造工程师和质量工程师。

这个系统的软件将是一个购买来的软件包，但戴夫知道，自己的公司将需要进行自定义修改。自己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。在他眼里，焊缝深度和其他关键参数的严重程度很低，因为他确信：下游泄漏检查和检查焊缝熔深的定期破坏性测试就足够的。同样，泄漏检查将会确认：气密密封是令人满意的。这在用户界面方面留下了风险，确切地说，风险是开门时，软件可能启动激光。戴夫知道门的密封有软件检查措施，但是如果软件无法按预期运行，风险的严重程度就会很高，因此，他增加了一个冗余硬件互锁装置，以防在门打开时激光激活。

**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范围广泛的编程工具。因此，前面创建的软件要求说明和评审（工具）足够满足设计需要，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。测试计划应包括软件环境的详细信息和预期测试结果的详细内容。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审核批准。测试报告将包括实际测试结果，并将其与预期结果进行比较，提供合格/不合格提示，列入测试标识，并且提供问题解决和任何故障回归测试的文档。就本报告而言，戴夫希望获得自动化工程师、制造工程师、质量工程师和项目发起人的额外批准。

**部署**

针对焊机的部署，戴夫审查了工具箱中的部署工具，断定需要制造操作员程序，而且该程序需要由自动化工程师、制造工程师和质量工程师审查。为了确保操作员了解如何操作焊机，戴夫创建了包括有某种测试的一种操作员培训与认证程序。他知道，MES在未经认证的情况下，不允许操作员将焊接程序抽离系统，因此他很清楚，操作员受伤的风险已经成功地减轻。

**维护**

戴夫知道，自己的公司具有配置检查工具。因此，确认期间没有执行具体的维护计划。

**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——流程要求**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **开发** | **定义** | **流程** | **迭代风险分析** | **制定确认计划和报告** | **软件系统** |
| **流程要求（见5.3.2.2）**  器械公司是家C类（参见GHTF/SG1/N77:2012标准）医疗器械制造商。器械公司已决定实施一套自动焊接工艺控制体系。为确保设备外壳焊接合格，器械公司将采用的办法是，利用一种参数化发布决策流程隔离产品。器械公司还决定用来自此流程中的信息支持其器械历史记录。  器械公司已指派了一名新的项目经理来确认该自动焊接工艺控制体系。该项目经理意识到，该体系需要满足ISO 13485标准的软件确认要求。因此认识到，所提出的焊接工艺控制体系需要确认。  为了更好地理解焊接系统确认所涉及的要求和风险，项目经理将确定了如下流程：   1. 操作员将批号输入该批次第一部分的系统中。 2. 操作员将子组件插入机器夹具中。 3. 操作员按下循环开始按钮。夹具通过液压机械移动到适配位置。 4. 焊接循环周期连同固定子组件的固定速度旋转一起开始。 5. 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。 6. 机器在循环结束时打开固定装置。 7. 操作员移除焊接部件，并根据顺序编号将该部件放置在批次托盘的相应位置。 8. 操作员重复步骤b）至步骤g），直至批次托盘装满为止。 9. 操作员点击批次结束按钮。 10. 机床操作员界面显示焊接温度超出工艺极限的零件序列号。 11. 操作员从批次托盘中丢弃相应的部件号。 12. 操作员打印废品清单并发送批次托盘和报告到下一个工作站。 13. 操作员通过重复步骤a），开始新的批次。   项目经理还意识到，关键自动化功能还包括如下内容：   * 存储批次号； * 存储每个序列号的焊接温度； * 显示焊接过程中超出过程温度极限的零件的序列号； * 打印批次作废报告。 | | | |

**表C.2——示例3——过程故障风险分析**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **开发** | **定义** | **流程** | **迭代风险分析** | **制定确认计划和报告** | **软件系统** |
| **流程故障风险分析（见5.3.2.3）**  然后项目经理考虑了当前流程可能出现问题的地方。项目经理意识到，如果流程崩溃，焊接不当的部件释放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产生。  然后，项目经理考虑了哪些用以降低风险的风险控制措施已经到位。项目经理得知，流程组有一个现成的程序，可以确认焊接操作员是否已在下个工艺步骤正确地摈弃了零件。此外，项目经理还了解到，焊接系统是商业OTS系统。 | | | |

**表C.3——示例3——软件目的和意图**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **开发** | **定义** | **流程** | **迭代风险分析** | **制定确认计划和报告** | **软件系统** |
| **软件目的和意图（见5.3.2.5.2）**  项目经理对流程具有基本了解，可以随时编写焊接流程控制系统的目的和意图。   * 焊接流程控制应用程序对焊接壳体的合格或不合格状态做出闭环质量判定。焊接操作员根据判定结果，手工分捡出不合格产品。   项目经理审查了在流程内适当捕获软件边界的目的和意图，并决定按如下方式修改说明：   * 焊接流程控制应用程序对焊接壳体的合格或不合格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。焊接站是整个器械流程唯一可确保设备密封完整性的控制点。   然后，项目经理考虑的是，其他系统（若存在）要与焊接系统连接需要什么东西。他断定，该软件是在一套在个人计算机上运行的单个应用程序，计算机上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。焊接系统没有连接在网络上。 | | | |

**表C.4——示例3——制定确认计划**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **开发** | **定义** | **流程** | **迭代风险分析** | **制定确认计划和报告** | **软件系统** |
| **制定确认计划（见5.3.2.4）**  既然项目经理了解流程，并已确定了新系统的预期用途，因此，项目经理可以随时制定高级别的确认计划。  早些时候，项目经理断定，焊接流程存在高残留风险，因为焊接流程的实施是个不可确认的过程。因此，项目经理断定需要对确认工作进行广泛的审查。项目经理决定，关键审批任务宜由流程工程与质量工程部以及运行流程培训师完成。此外，最终产品验收经理宜批准这些要求。  项目经理决定开始编写确认计划，因为质量体系要求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高风险系统的确认计划。 | | | |

**表C.5——示例3——软件使用要求与软件要求**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **开发** | **定义** | **流程** | **迭代风险分析** | **制定确认计划和报告** | **软件系统** |
| **软件使用要求与软件要求（见5.3.2.5）**  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程式，而且了解定义详细流程和软件要求的重要性。项目经理现在编写软件要求。项目经理断定，软件定包括温度确认和不合格判定过程的冗余量。项目经理还规定系统必须能够在线路清理活动发生之前的任何时间重新打印不合格报告。  由于该系统支持参数值，因此，项目经理还列出了严格性要求以及系统访问级别可更改数据值的详细列表。 | | | |

**表C.6——示例3——软件故障风险分析**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **开发** | **定义** | **流程** | **迭代风险分析** | **制定确认计划和报告** | **软件系统** |
| **软件故障风险分析（见5.3.3.2）**  项目经理现在需要决定采用何种方法建立对焊接系统的充足信心。  项目经理指出，焊工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。  虽然项目经理早已断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故障的风险。为了证实这种直觉，项目经理回顾了公司风险模型的各种问题。  a）如果软件出现故障，产品安全是否存在潜在风险？*是*   1. 怎样出现？*系统基于默认温度限制认为次品合格。电源故障之后限制重置为默认设置。* 2. 宜该采取哪些措施控制此风险？*要求操作员在每批次运行开始和结束时核实限值。*   b）如果用户出错，产品质量是否存在潜在风险（安全风险除外）？*是*   1. 怎样出现？*如果两个部件传感器都在手动模式下起动3秒钟，则焊接激光器可能会触发。* 2. 宜采取哪些措施控制这种风险？*将默认配置更改为仅在自动模式下触发。* | | | |

**表C.7——示例3——制定确认计划**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **开发** | **实施、测试与部署** | **流程** | **迭代风险分析** | **制定确认计划和报告** | **软件系统** |
| **有效性计划（见5.3.3.3）**  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施方法，并已分析了软件风险。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。”  项目经理思考了第三方正在采用的系统开发方式，忧虑的是开发人员是否正确诠释了报告定制要求。由于该系统将依赖于各种数据字段，因此，项目经理在代码审核中添加了确认步骤活动，以确认开发人员工作的正确性。 | | | |

**表C.8——示例3——软件实施**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **开发** | **实施、测试与部署** | **流程** | **迭代风险分析** | **制定确认计划和报告** | **软件系统** |
| **软件实施（设计、开发，构建与测试）（见5.3.3.4）**  购买软件而不内部开发此软件的决定依据是有可供购买的商用现货（COTS）系统。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命周期（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，项目经理了解到，供应商的SDLC流程最近接受了一家独立审计公司的审计。随后项目经理联系了那家独立的审计公司并且购买了COTS供应商SDLC审核报告的副本。最终结果是，质量部门确信COTS供应商开发此软件是在有效生命周期模型下进行的。 | | | |

**表C.9——示例3——确认报告**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **开发** | **实施、测试与部署** | **流程** | **迭代风险分析** | **制定确认计划和报告** | **软件系统** |
| **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3——软件发布**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **开发** | **实施、测试与部署** | **流程** | **迭代风险分析** | **制定确认计划和报告** | **软件系统** |
| **软件发布（见5.3.3.6）**  项目经理证实，位于正式配置管理系统下的软件与确认报告中引用的软件匹配。 | | | |

**表C.11——示例3——变化分析**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **维护** |  | **流程** | **迭代风险分析** | **制定确认计划和报告** | **软件系统** |
| **变更分析（见5.4）**  项目经理证实，根据确认计划，公司有一个正式的变更控制流程，用于管理任何确认后对焊接系统的更改。 | | | |

**表C.12——示例3——制定维护确认计划**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **维护** |  | **流程** | **迭代风险分析** | **制定确认计划和报告** | **软件系统** |
| **制定维护确认计划（见5.4.2）**  在系统始终满足其预期用途方面，项目经理提前考虑了适合用来确保置心度的活动。考虑到系统风险较高，项目经理断定宜每季度进行标定和认证，确保实际温度测量结果与批次报告中打印的温度值相比，准确度和精密度均能达到要求。项目经理在确认计划中包含一个用于记录此结论章节，并要求开发实施一种标定与认证程序，确保在系统投入生产后按季度接受审核。 | | | |

**表C.13——示例3——软件维护**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **维护** |  | **流程** | **迭代风险分析** | **制定确认计划和报告** | **软件系统** |
| **软件维护（见5.4.6）**  项目经理证实，根据确认计划，公司有一个定期审查流程，能够确保焊接系统及流程不会偏离其预期用途。 | | | |

**表C.14——示例3——软件退役**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **退役** |  | **流程** | **迭代风险分析** | **制定确认计划和报告** | **软件系统** |
| **软件退役（见5.5）**  项目经理证实，根据确认计划，公司有正式的软件退役流程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

* 流程要求定义
* 正式软件要求审查
* 确定制造与业务流程内的风险控制措施
* 流程开发审查
* 可追溯性矩阵（要求说明内固有的）

测试工具

* 测试计划
* 软件系统测试
* 软件配置控制

部署工具

* 用户程序审查
* 内部应用培训
* 安装资格
* 流程确认

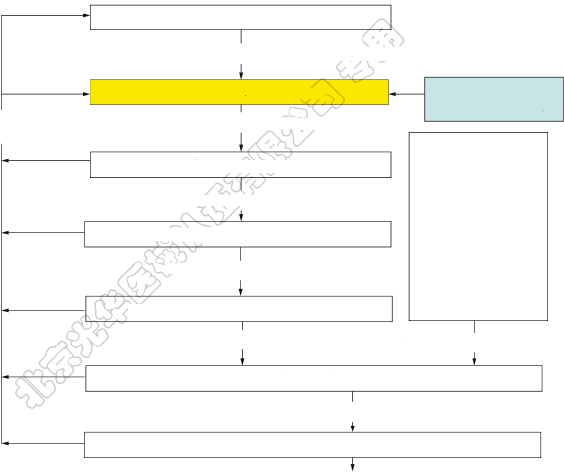
**示例4：C/C++语言编译器**

**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，该编译器是受控制的，因为此编辑器产生置于医疗器械设计记录内的医疗器械产品软件（软件源代码和可执行软件）。

**描述质量体系流程**

两个质量系统流程与本案例研究相关。第一个是C类医疗器械软件的整体质量体系实施流程（见图C.1）。第二个是实现软件设计并满足所有软件要求的可执行软件单元的开发流程。软件单元包括OTSS C/C++语言编译器（见图C.1中的“软件实施”部分）。



上游流程

**同行评审的系统要求、危害分析、软件要求和软件设计**

软件实施

*OTSS C/C++语言编辑器*

**通过CM和缺陷跟踪流程更改基线化人工制品**

**同行评审的软件单元**

软件单元测试

**单元测试的软件单元**

软件单元集成测试

**经过白箱测试的软件应用程序**

软件要求确认

硬件开发流程

**经过功能测试的软件应用程序**

**经过功能测试的硬件**

系统集成测试

**经过白箱测试的III类医疗器械**

系统确认与检验

**经过检验和确认的III类医疗器械**

**图C.1——C类医疗器械软件的实现**

**上游流程**

用于软件实施的流程的上游是用于开发系统级文档（例如，要求、设计、危害分析等）从而表征待开发的医疗器械的流程。然后，凭借软件要求开发流程、软件设计和其他软件文件或计划的流程，表征在软件内实施的系统部分。在软件开发的同时，执行附加流程，从而开发医疗器械硬件。

**软件实施流程**

所使用的正式软件语言是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机器代码。软件实施过程的输出是基线化软件单元，基线软件单元由其他技术成员进行完整性和正确性同行评审。对于软件单元同行评审，软件单元宜在最高编码器级别无错误编写，而且任何编译器警告宜在同行评审中予以解释。

**下游测试流程**

软件单元在以下几种测试流程中进行测试或确认：

* 软件单元测试。个体软件单元测试其逻辑正确性和每个单元的边界条件。这种测试可能在开发系统或目标系统（医疗器械硬件）上出现。在断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放弃此测试。
* 软件单元集成测试。为了确保软件设计正确实施，软件单元接受集成和测试，而且与设计相关的边界条件也要进行测试。此测试在目标系统上进行。
* 软件要求确认。完整的软件应用程序根据整套软件要求进行确认。此确认在目标系统上进行。
* 系统集成测试。对医疗器械中的软硬件进行测试，以确保系统设计正确实施，而且与系统设计相关的边界条件也要进行测试。
* 系统确认与检验。医疗器械在系统要求级别进行检验，此外，还针对其预期用途进行确认。

**分析流程故障风险**

该项目遵循公司的流程风险评估程序。由于C类医疗器械软件的整体质量体系实施过程（包括图C.1描述的所有流程）产生会在C类医疗器械内起作用的软件而存与生俱来就有很高的风险。

作为软件实施流程的一部分，OTSS C/C++语言编译器基于两个因素的评估结果为低风险：

* 编译器不会直接给患者、操作员或旁观者造成严重伤害或死亡；
* 对工具的输出（软件源代码和可执行软件）实施下游确认（例如，软件单元测试、软件单元集成测试、软件要求确认、系统集成测试、系统检验与确认等）。

**预期用途的确定**

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，以生成某C类医疗器械的可执行软件。

**软件使用要求**

1. 该工具宜交叉编译要利用所选供应商操作系统在精简指令集计算机（RISC）处理器上运行的C代码和C++代码。
2. 编译器宜具备一个源代码调试器。
3. 编译器宜符合美国国家标准协会（ANSI）的C标准和C ++标准。
4. 编译器宜与各种已批准的行业标准集成开发环境集成。
5. 供应商宜发布可搜索的已知错误列表。此清单宜作为参考，根据需要进行查阅。
6. 供应商必须在受监管行业内拥有庞大的用户群。

**分析软件故障风险**

OTSS C/C ++语言编译器风险分析表明，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流程开发方法和支持功能。
  + 减轻措施1。请参阅下文的“供应商选择流程”部分。
* 风险2。编译器生成错误的可执行语句。
  + 减轻措施2。请参阅下文的“确认计划”部分。
* 风险3。未运用最严格错误检查级别的用户不正确地使用编译器。
  + 缓解措施3。改进培训、程序和工作指示。

**供应商选择流程**

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设计记录中。此程序包括现场评估、审查供应商的SDLC政策、程序、任务和活动。对供应商提供的OTSS C/C++语言编译器功能是否满足上面定义的软件使用要求进行了确认。

**制定确认计划**

OTSS C/C++语言编译器选择了一种下游确认方法。供应商选择流程已确定：该供应商满足所有已记录的软件使用要求。编译器在供应商处具有大量运行时间，而且在项目调试和测试期间也会有大量运行时间。编译器的输出在下游流各进程接受以下动态测试：

* 软件单元测试；
* 软件单元集成测试；
* 软件要求确认测试；
* 系统集成测试；
* 系统检验与确认。

**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
  + 软件要求
  + 补丁
* 风险评估和危害分析
* 供应商选择
* 安装活动
* 确认
  + 软件使用要求测试案例和结果
* 已知错误列表
* 配置控制
  + 培训
  + 安装位置
  + 维护
  + 退役流程

**工具箱选择**

* 定义阶段：
  + 预期用途；
  + 制定确认计划；
  + 风险管理计划（风险评估）。
* 实施阶段：
  + 风险控制措施；
  + 供应商审计。
* 部署阶段：
  + 安装资格；
  + 内部应用程序培训；
  + 最终验收测试。
* 维护阶段：
  + 维护计划；
  + 已知问题分析。

**示例5：自动软件测试系统**

**背景资料**

在本示例中，制造商是一家C类医疗器械制造厂。此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（驻留在一台台式机上，运行行业标准操作系统和数据库）和实时嵌入式软件（驻留在一块板载嵌入式CPU卡中）接口采用标准传输控制协议/ Internet协议（TCP / JP）硬件和协议。

项目软件经理已经断定，通过引入软件自动化测试改进软件开发和测试流程是很有价值。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。集成测试点和软件系统测试点均会进行自动化软件测试。

**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流程所需的测试，而这将在集成点和系统测试点为所需回归测试提供证据，所以自动化测试软件被确定为开发流程的自动化部分，并因此确定其符合ISO 13485标准的确认要求。

**定义流程**

为了更好地理解引入操作员控制台自动化软件测试所涉及的要求和风险，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，安排了各种模块在不同时间集成到系统软件的时间。此外，已集成到系统中的模块由于缺陷修正和要求修改，将会发生变更。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内某个特定模块的最终测试。软件项目计划要求模块每周集成或更新两到三次。自动化测试将在每个集成点上运行，确保新功能正常工作，而以前运行的功能不会受到新增代码或特定构建代码更改的不利影响。自动化测试将在软件系统测试级别运行，用于测试最终接受确认并最终向客户发布的候选品。如果在开发的最后阶段发现缺陷，需要加以修正，也会进行自动化测试，以便提供一定程度的回归测试，以补充原订手动测试方案。

**风险分析**

如果自动测试软件使用不当，现在软件经理会利用分析流程确定任何潜在影响。

软件经理首先需要评估的是自动化测试过程的某个故障、自动测试软件的某个故障或任何使用自动测试软件的人所犯的错误，最终是否会导致某种可能对患者、操作人员、旁观者、服务人员或环境造成伤害的医疗器械缺陷。

* 软件管理员最关心的问题是，自动化软件测试系统会给出错误显示，在接受测试的操作员控制台软件确实仍然存在缺陷时，显示工作正常。
* 如果未检测到的缺陷位于软件的关键区域，则未检测到的缺陷可能导致医疗器械出现故障，从而造成危害。
* 软件经理意识到，这种风险可能源于自动测试软件使用的管理不当或自动测试软件本身的缺陷。
* 软件经理断定，围绕自动化软件测试系统使用时机以及使用方向设置边界条件，确保软件开发和测试团队不过度依赖系统，极其重要。
* 参与自动测试软件配置、编程和操作的个人需要接受岗位培训。
* 软件经理认为，如果这些因素得到控制，潜在相关风险将降低到可接受水平。

**确定软件的预期用途**

在分析了自动化测试软件的潜在用途和相关风险后，软件经理既可编写自动化软件测试系统目的和意图说明。说明内容如下。

* 自动测试系统将用于在开发过程中在集成测试点测试软件的构建。
* 自动测试系统将用于在软件系统测试点测试确认与候选发布版本。
* 自动测试系统将对系统进行回归测试，确保新引入的软件或软件更改不会对工作流程产生不利影响。
* 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流程，自动测试系统可用作判断软件正确性的最终决定因素，因为经过确认，特定协议与相应的手动测试一致。
* 自动测试系统将运行软件，为软件系统或整个医疗器械提供防护措施（风险减缓措施）。

**制定确认计划**

软件经理现在清楚地知道要自动化的过程，清楚地知道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，软件使用需要进行某些控制，所以，自动软件测试系统（如果使用方式为软件经理指定适当控件的方式）与其使用相关的风险将处于可接受水平。

* 因此，软件经理已断定，如果使用适当，自动化软件测试系统导致医疗器械缺陷的风险很低或没有风险。软件经理已经进行了适当定义，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，软件经理已经断定，就软件测试系统的测试而言，该系统的确认要求在工作量和严格性上都会偏低。

**确认文档：确认报告方法**

软件管理员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，其中要包括与本系统获得必要信心相关的所有活动的摘要。

**批判性思维**

软件经理现在确定了达到必要信心水平的最好方式，认为系统使用适当，并且不会导致医疗器械出现严重缺陷。

他认为，以下因素是决定系统达到必要信心水平的最重要因素：

* 绝对坚持适当的预期用途
  + 确保参与软件开发和测试的所有人员清楚地了解边界条件和系统的适当预期用途。
  + 文档：在确认报告中保留一个章节，用来描述具体的预期用途以及通过项目软件开发计划传达此信息的方式。
* 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，要求其测试系统正在用于相同级别的危险程度测试或正在用来测试危险程度更高的应用程序。
  + 与供应商一起审查系统的预期用途，确定预期用途是否合适。
  + 获取有关供应商在软件向商业市场发布之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。声明将使人相信：供应商对自动化软件测试系统进行了充分测试，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要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，以确保能够提前考虑新版本软件的减缓计划和复审活动。
  + 文档：在确认报告中包含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信息、有关供应商缺陷（错误）列表访问方法的信息，以及关于新版软件预期减缓计划的信息。
* 安装测试
  + 确认软件所处计算环境符合供应商的规格。
  + 确定初始高级测试协议，以确保软件已正确安装。
  + 文档：在确认报告中包含一个章节，描述安装确认活动的结果。
* 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使用。
  + 在将使用自动测试系统的项目的软件开发计划中包含特定的容许边界条件。
  + 分析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：在确认报告中包含一个章节，描述初始风险分析确定的风险，并说明每种风险的减轻方式。
* 软件使用要求
  + 开发自动测试系统功能列表，列出打算使用的自动测试系统的功能。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：“软件使用要求”列表放在在确认报告的某个章节，描述每个软件的使用要求。
* 自动测试系统的确认
  + 使用“软件使用要求”列表确定必要的信以水平。信心水平可以通过采用三个初始自动化测试脚本或协议，同时对照手动运行相同的协议进行并行测试来建立。三个初始测试脚本或协议使用团队将使用的所有功能。
  + 文档：在确认报告中留出一个章节，总结并行测试的结果，列出测试证据，证明结果相当。
* 培训
  + 为所有系统用户制定培训计划，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，培训是确保自动化软件测试系统安全有效使用所需的最重要元素之一。
  + 文档：在确认报告中留出一个章节，描述系统用户所需的必要培训。
* 单个自动化测试协议的确认
  + 如果自动测试系统的测试对象是用来减轻系统、软硬件风险和危险的软件，请确保每个协议都已使用自动测试和手动测试进行了并行测试确认。
  + 如果自动测试系统将用于复杂性较低的可预测工作流程的最终测试，请确保每个协议都已使用自动测试和手动测试进行了并行测试确认。
  + 文档：确保医疗器械的软件确认记录包含证明接受并行测试的测试脚本或协议分类适当的证据。
* 配置管理
  + 确保仅安装和使用经过适当确认的自动测试软件版本。
  + 在供应商可以提供新版自动测试软件时，控制新版本或更改的实施，确保在适当时间引入新版本或更改。
  + 确保每个更新点都考虑了自动测试系统的重新确认，并且确保系统进行了每个重新确认而且每个重新确认都进行了记录。
  + 文档：在确认报告留出一个章节，描述系统的配置管理计划。

**确认报告**

信心建立活动的结果是，软件经理提交确认报告，以供最终审批。此报告传达的是确定待进行增值活动的思维过程，以便软件经理可以断定，使用自动化软件测试系统会使正在开发的相关医疗器械不小心存在缺陷。此报告还包含表明所有既定重要活动都已按计划进行的证据。

确认报告包含以下内容：

* 流程定义；
* 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、重新确认与配置管理。

**确认报告审批**

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审批。

所有审核人员一致认为，软件经理已清楚透彻地考虑了系统的预期用途，而且了解系统使用中涉及的所有相关风险。审核人员认为，达到允许使用系统所要求的系统信心水平需要完成的所有活动已经全部完成。审核人员批准此计划。认定系统已通过确认，并投入使用。

**示例6：简单的电子表格**

**背景资料**

ZYX公司的实验室分析人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数。实验室中的仪器用于接收检查。该仪器测量三个坐标位置，分析人员使用这些坐标位置计算与规格进行比较的角度。实验室最近遇到过三次分析分员角度计算不当的情况（分析师说原因是“键盘操作失误”），分析人员希望杜绝这种错误再次发生。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50种产品的规格全部放入电子表格。他们要输入仪器测量的三个坐标对，从下拉菜单中选择产品名称，并且获取合格/不合格结果。分析人员还考虑仪器利用某一接口将坐标直接传递给电子表格，但由于接口成本较高，增强功能要延迟到明年。

**定义流程**

当前流程包含以下步骤。

1. 让仪器测量零件。
2. 记下三个坐标对。
3. 计算角度。
4. 从文档控制系统中提取零件的规格。
5. 角度值与规格进行对比，确定是否合格。
6. 在零件上放上合格证或不合格证，并将其发送到产品零件库存中。

新流程将包含以下步骤：

1. 从文档控制系统获取电子表格。
2. 让仪器测量零件。
3. 在电子表格中输入三个坐标对。
4. 根据仪器值，目视检查输入的坐标对。
5. 在电子表格中选择部件号。
6. 在电子表格中选择“计算结果”。
7. 目视检查选择的部件号是否正确。
8. 根据结果，在零件上放置合格证或不合格证，并将其发送到产品零件库存中。

**定义预期用途**

分析人员按如下方式定义电子表格的目的和意图：电子表格采用三个已输入的坐标算出一个角度，然后将此角度与所选产品的产品规格进行对比，报告合格/不合格结果。

**风险分析**

分析人员集思广益，列出了与电子表格有关的可能危害。他们认为，结果不正确可能意味着生产中可能混入不符合规格要求的部件。这样的缺陷部件，如果随着医疗器械到达了最终用户的手里，至少还会发生其他两个下游故障，但仍然存在（如果不大可能的话）对最终用户造成伤害的轻微风险。因此，生产不符合规格产品的风险很低。然而，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，问题直到进行第一次子组件检查才会发现。结果必须废弃整个子组件。此外，如果收到的不合格结果有误，那么，可能会丢弃优质零件，这会再次增加废料的成本。因此，将会以电子表格设计、程序控制、文档审查和测试的形式增加严谨性要求，解决业务上所关心的问题。

**制定确认计划**

由于生产不合格产品的风险较低，因此，确认工作的工作量会比较低。分析人员决定将电子表格的要求和确认计划合并到同一文档中。分析人员还决定将设计文档与高级测试计划合并在一起。此类文件，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进行审核。此外，分析人员还计划咨询技术专家，开发一组有代表性的测试数据，建立计算按预期运行的信心。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。分析人员会确定每个项目的风险减缓措施（见表C.15）。

**表C.15——示例6——风险与缓解措施**

|  |  |
| --- | --- |
| **风险** | **减缓措施** |
| 可能输入不正确的值。 | 通过程序控制，确认针对仪器输入的每个值对。新流程里添加了步骤d）来完成这一工作。 |
| 计算可能不正确。 | 确认公式正确，并且该公式按预期提供了准确结果。 |
| 可能选择有问题的产品。 | 通过程序控制确认部件号。新流程内添加了步骤g）来完成这一工作。 |
| 用于显示结果的宏会不正确。 | 确认宏正确无误并按预期运行。 |
| 电子表格中的规格要求可能不正确。 | 根据50个产品的规格表，确认电子表格的规格要求。增加规格表更改流程，以便在规范发生变更时要求更新电子表格。（这种情况从未发生过，但有可能出现。） |
| 计算公式或宏在确认后可能更改。 | 带有配置控件的电子表格经过确认的将被放入文档控制系统，并在每次需要时再次获取。配置控件将包括所有非数据输入单元的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开发方面经验丰富。确认将确认以下项目：

* 计算；
* 宏；
* 单元锁定功能（锁定单元不能更变）；
* 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。所有测试将创建一个测试计划及报告。此报告还将发布使用电子表格，并且确认公司文档控制系统中此电子表格的控制状况。

**部署**

在部署新系统之前，测试已完成，生产经营商已通过新视觉系统的作业认证。

**工具箱中的工具**

* 要求定义（记录在确认计划中）
* 过程故障与风险分析（记录在确认计划中）
* 预期用途（记录在确认计划中）
* 制定确认计划
* 制定测试计划
* 运营商认证
* 制定维护计划（需要回归分析）

**维护**

产品规格每次更改或每次添加新产品，都要对电子表格进行维护。将会开发一个维护测试计划，其中包含完整确认测试用例的代表性子集，以确保新项目不破坏该电子表格。维护计划会调用回归分析，查看是否需要在测试用例子集内额外添加专门针对所做更改的测试用例。此计划还会描述电子表格的更新方式（例如，解锁单元、更改、重新锁定等）。

**示例7：（并非如此）简单的电子表格**

**软件说明**

软件开发团队使用Microsoft Excel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。该器械的原始版本采用美国英语书写。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。其余每列代表一种软件支持的国际语言，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**

电子表格满足瞬态需求，以便

* 直观地组织英语消息及其翻译，
* 创建可以发送给当地代表的电子表格，以便将翻译信息直接收集到电子表格中，或者以手写体的形式收集在电子表格的硬拷贝上，
* 为翻译消息提供翻译数据存储工具。

译文收集完毕并译成器械软件之后，就无需再保留或维护电子表格。

任何计算单元或宏都非此电子表格的一部分。

**确定软件是否处在范围之内**

Excel仅用于编排信息的格式，以便传递、收集器械消息的外文翻译。乍看起来，电子表格似乎是Excel的一个简单应用程序，人们很容易认为此电子表格不需要确认。

第5.2节提出了以下问题：“软件故障或潜在缺陷是否会对医疗器械的安全性或医疗器械的质量产生不利影响？”

这个问题的答案显然是“是”。如果软件或电子表格故障致使存储在此的消息翻译遭到破坏，则该故障可能会影响器械的安全性。虽然团队认为，这种“简单的应用程序”发生故障的可能性很低，但发生故障的可能性仍然处在ISO 13485标准的确认要求范围之内。

**风险评估**

如果器械消息未正确翻译，则可能导致用户混淆或消息错误解释。因此，存在器械间接伤害患者的可能性。软件故障可以检测出来，而且在器械开发和确认过程中存在大量通过交叉检查检测、纠正软件任何故障的机会。

可能对器械软件产生负面影响的预期故障模式如下：

* 待翻译的英文原始消息由于输入文件丢失、个别消息丢失、消息顺序错误而错误百出，从而导致上下文丢失，或由于随机丢失、替换或字符转换导致个别消息错误百出；
* 区域办事处准备和收集的个人翻译信息错误百出。错误百出的原因可能是由于输入文件丢失、个别消息丢失、消息顺序错误，从而导致上下文丢失或者导致个别消息由于随机丢失、替换或字符转换而错误百出。如果Excel中未正确安装某些非英语字体，则还存在该字体错误百出的可能性；
* 收集结果电子表格错误百出。收集结果电子表格显示每个翻译的结果累积。造成错误百出的原因可能是输入文件丢失、个别消息丢失、消息排序错误，从而导致上下文丢失，或者个别消息由于随机丢失、替换或字符转换而错误百出。除电子表格行排序错误之外，还可能发生列排序错误。在不以原生字体和字符集内的字体和字符显示已翻译消息的各列，软件工程师将会曲解原消息的意思，因为他们要把消息转换成代码。

**制定确认计划**

软件开发工程师承认，如果新器械的消息有误，患者可能面临风险。软件某个故障的严重程度可能很高。需要为建立信心做点什么，令人想信电子表格内组织的消息翻译正确。

但是，Excel仅用于组织信息。对Excel进行大量测试发现会导致消息错误百出的任何缺陷，似乎不太可能。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更容易导致错误。

考虑到人为错误，工程师们意识到，没有用于收集翻译或确认无人为错误潜入流程的明确界定的流程。

工程师们创建了一个用于收集和确认翻译信息的书面程序。然后考虑了可能存在哪些会导致其流程崩溃的风险，软件（即Excel电子表格）故障可能以何种方式导致这种崩溃，最后考虑了，能够采取哪些措施确认该流程（包括电子表格）。

**风险控制措施**

在更好地定义翻译收集流程之后，工程师们识别了用于保护该流程的风险控制措施，以免错误避免嵌入消息翻译。

用于保护翻译收集流程的风险控制措施，同样也会防止软件故障，以免软件无法满足其预期用途。

* 从地区办事处购买软件，译文宜以纸质（硬拷贝）格式提供，或以电子格式提供，同时附带相应的硬拷贝。如果区域办事处提供电子版本，则该电子表格内的数据将在转移到主翻译电子表格时依据硬拷贝进行确认（并记录）。此确认将避免传输期间电子表格损坏导致任何结果曲解，或避免在译文提供电脑与译文接收电脑间的字体能力差异导致任何结果曲解。
* 所有译文收集完毕并放入主电子表格后，宜将硬拷贝电子表格发送到每个区域办事处供各区域办事处审核批准。区域核准将避免传输期间电子表格损坏导致任何结果曲解，或避免在译文提供电脑与译文接收电脑间的字体能力差异导致任何结果曲解。
* 主电子表格得到所有区域办事处、开发审批人员和质量保证审批人员认可后，主电子表格的硬拷贝宜输入器械软件开发流程内。此外，主电子表格的硬拷贝宜作为器械软件翻译确认测试任何预期结果的来源。

**确认任务**

除风险控制措施之外，还宜完成其他确认确认任务，确保软件充分满足其瞬时预期用途。其他确认确认任务包括：

* 对于从区域办事处收集的每条译文，圴宜根据单个译文电子表格的硬拷贝逐行确认更新后主电子表格的硬拷贝。强制性地根据硬拷贝进行硬拷贝，从而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。此过程宜具体考虑以下因素：
  + 消息要求（即英语）随着开发过程中器械功能演变的变化情况；
  + 主文件随所提供译文以及区域办事处审查、修改的最新主电子表格变化的情况。
* 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。配置完整很重要，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。杜绝译文意外更改的唯一方法是在使用电子表格时采用相同配置。
* 电子表格的配置（即操作环境和版本控制）需要控制，防止变化混乱无序、缺乏协调性。选派某个单一个人负责决定配置更改时间以及变更历史的记录时间。
* 每个电子表格的版本均宜在其硬拷贝版本中可见。
* 器械软件中的翻译表宜显示硬拷贝主电子表格输入翻译消息软件时的版本信息。
* 单个翻译确认任务宜包括以下内容：
  + 英文消息宜通过逐行对比电子表格的主版本和翻译版本加以确认。这种比较可防止在将文件传输到区域办事处以及文件由区域办事处返回时可能发生的电子表格文件的任何损坏（例如，损坏或丢失的消息）。
  + 译文插入主电子表格后（手动插入或使用Excel的剪切/粘贴功能插入），宜根据翻译电子表格的硬拷贝，确认修订主电子表格硬拷贝输出的逐行对比结果。
  + 测试器械软件的消息实现情况时，测试程序比较已实现消息与预期消息，宜采用最新版本的主电子表格的硬拷贝（并宜参照版本号）。

宜记录并收集所有确认任务的情况，作为流程和Excel电子表格通过确认的客观证据。

这种确认方法可以100％确认软件输出与输入的对比情况。对电子表格没有安排另外的测试工作。尽管传统测试缺失，工程师们仍对其流程充满信心，他们相信自己的确认原理一直很有价值。工程师断定，软件的任何故障都会检测出来，而且他们还有一个恢复路径，可以利用在该流程的适当切入点收集和记录的硬拷贝实现恢复。该硬拷贝以及逐行确认记录提供了活动的书面证据。

**维护**

电子表格旨在满足瞬态需求。翻译消息嵌入代码后，就会退役。因此，没有制定维护计划。

**讨论**

电子表格的预期用途和初始风险分析，在确定电子表格需要进一步确认关注时，曾经是两个至关重要的因素。预期用途发生了变化，一模一样的电子表格可能会轻松地得出电子表格风险很低，其复杂程度肯定也很低的结论。如果预期用途只是跟踪译文收集进程（即实施活动的设计过程，不采用电子表格内的译文），那么，结论可能一直会是：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，甚至不属于监管范围。

该软件正在“自动化”的“流程”是该器械消息翻译数据收集、格式化和存储过程的一部分。本示例在以下几个方面而人寻味：

* 确认（若存在）几乎没有要求通过软件测试确认软件的用途。请务必注意，软件（Excel）和电子表格之前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。该团队深感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，那对器械就是致命的。
* 确认为100％地确认电子表格的输出。硬拷贝版本被视为“黄金标准”。一旦硬拷贝获得批准并在设计历史文件中使用，软件的任何后续故障都都不再重要。批准之前软件的任何故障都会在审批过程中捕获。
* 此“流程”做过修改，使其免受电子表格软件任何故障的影响。
* 工程师们认为，此应用程序出现人为故障的可能性远高于出现软件故障的可能性。用户可能会出印刷错误，可能使用不正确的电子表格版本，或可能会出现类似错误。在这种情况下，“软件确认”也使该流程对人为错误更具免疫力。
* 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。

注意：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发生了人为错误。出乎意料的是，与安装不同Excel的不同电脑链接的字体版本相关的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出了问题。）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，实际上在消息翻译损坏方面成了问题。

**示例8：参数灭菌器**

玛丽的任务是领导一种新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。

**定义流程**

玛丽首先定义并记录了她所了解的她所在工作即将引入的100％的环氧乙烷（ETO）灭菌流程的知识。

* 医疗器械手工放入灭菌器内。此流程包括灭菌周期参数评估，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，手工取出医疗器械，并将其转移到脱气室内。

**分析流程风险**

玛丽非常关注这一流程带来的风险。此流程发生故障，可能会产生严重后果，包括：

1. 医疗器械灭菌不当。这种故障可能会因使用某非无菌产品而导致严重伤害或死亡；
2. 器械历史信息丢失和产品可追溯性丢失；
3. 有毒化学品进入制造设施或环境。这种故障可能导致当地社区的灭菌器操作人员或个人严重受伤或死亡。

因此，玛丽考虑了宜采取哪些风险控制措施减轻这些风险并对其加以确认。玛丽认为，通过使用参数灭菌技术可以使风险得到控制，确保在正确温度和正确相对湿度下，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，判断参数值是否适当，是确认灭菌是否充分的独立过程。最后她认为，需要采用故障安全停机和屏蔽壳结构，控制进入设施的化学品泄漏。

有了这些风险控制措施，除非多个系统故障同时发生，否则，不可能出现非无菌器械。但是，由于多个系统故障同时发生会造成不良影响，所以，玛丽认为剩余流程风险很高。因此，进行严格确认是恰当的。

**定义软件目的和意图**

玛丽希望详细了解软件在该系统内的使用方法。首先，她考虑了软件应该做什么。在本案例中，软件采用一个100％的ETO灭菌容器控制医疗器械的灭菌过程，包括记录有关器械历史记录内容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可容纳的批量比现有系统可容纳的批量更大；而满足当前的产品需求，需要更大批量。灭菌操作人员将和质量保证团队一起使用该系统，确定发布医疗器械的可接受性。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容器以及在某数据库中存储信息实现。玛丽欣慰地获悉，该系统将以实物的形式安放在现场灭菌设施内，而且该系统通常每周关闭一天，接受任何必要维护。

玛丽断定，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，直到从容器内人工取出该器械。

玛丽记录了软件的目的和意图，内容如下：

* 灭菌软件将控制并监控灭菌过程，并将评估决定参数放行的灭菌周期参数。

**制定确认计划**

现在玛丽了解了该软件的预期目的，也就做好了在高层次上制定确认计划的准备。她知道，稍后自己需要添加更多细节，但她现在就想开始确认计划制定工作，这样她才能够以明智的方式识别软件的故障风险，并用识别出的风险完成自己的计划。

由于之前识别出了高残留流程风险，玛丽认为，自己需要提供详细确认内容和确认程序。她希望文档采用高级别的严谨性和详细程度，希望大多数文档以独立文档的形式存在，而不像平常那样，为了减小工作量组合在一起。由于该系统伴有高风险，她决定对待开发工作要采用与她对待开发医疗器械软件同等的严谨程度。所以她决定采用IEC 62304:2006/ AMD1:20l5标准作为生命周期控制方法。有关软件风险管理的指导，她参考了IEC/TR 80002-1标准。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。她还决定正式定义和记录用户业务流程要求和软件要求。任何特别关注的功能都会经过明确的识别鉴定。玛丽还安排了一次正式的软件要求审核。必须得到质量保证团队、灭菌工程师和灭菌经理的批准。由于本系统的重要性和风险，确认报告的最终批准将包括高级管理层。

**定义软件要求**

玛丽现在编写软件要求定义。她认为软件要求宜涉及警报错误处理与参数设置的消息确认、器械历史记录系统的接口、传感器控制监控、运动控制监控。

**建立对软件的信心和控制**

玛丽利用恒安的内部开发控制程序作为驱动程序，在整个开发生命周期使用内部控制。因为工作全部在内部完成，所以不需要发生供应商活动。

**定义与其他系统的软件边界**

然后玛丽考虑了新型灭菌器需要与其他哪些系统连接。她断定，唯一的接口将是与恒安现有数据库系统的接口，而恒安现有数据库系统将存储灭菌周期生成的数据。

**分析软件故障风险**

虽然玛丽已经断定即将实现自动化的业务流程风险很高，但她仍然需要分析软件故障的风险。参与此文档，玛丽为此活动选择了一个定量风险模型。她按如下方式为新系统划分等级：

* “严重性”风险很高（10分），因为系统故障可能导致死亡或严重伤害。
* “可能性”风险也很高（10分），因为软件故障本身可能导致伤害，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，这意味着风险分类高。风险分类高意味着宜采用严格的确认方法。所采用方法的严谨性和全面程度与以往灭菌器本身作为医疗器械所要求的严谨性和全面程度相同。

由于采取了缓解措施，该自动化系统的剩余风险已降低到最低程度。但由于系统可能带来伤害，杀菌过程本质上是一个高风险的过程。还执行了与（引自IEC/TR 80002-1标准）风险有关的其他活动。

**完成确认计划**

因为玛丽现在已经完成了软件要求的定义，已经决定了实施方法，也分析了软件风险，她有足够的信息完成详细的确认计划。

在撰写确认计划的第一稿时，玛丽已经断定宜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标准中识别鉴定），见下文：

* 风险管理工具：
* 软件故障树分析；
* 风险管理计划；
* 识别鉴定制造流程或业务流程内的风险控制措施；
* 分析软件故障（风险分析）。

然后，玛丽考虑了自己该如何获得对软件设计、开发和配置阶段的信心。她已经决定采用IEC 62304:2006/AMD1:2015标准进行生命周期控制。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：
  + IEC 62304:2006/AMD1:2015架构文档与和审查；
  + 设计规范；
  + 软件详细设计和审查；
  + 软件编码标准；
  + 可追溯性矩阵；
  + 确定软件系统设计中的风险控制措施；
  + 代码审查与代码确认；
  + 开发与设计评审。

玛丽毫不怀疑自己需要全面测试这个新系统。她首先断定，需要进行正式的测试计划活动以及例行的单元测试、集成测试和接口测试活动。但是，由于该系统将实时发布成品设备，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进行系统极限模拟，尽可能多地模拟操作条件。

* 测试工具
  + 测试计划；
  + 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，因为知道系统直到在生产环境中完全实现之后才能完整，所以玛丽将注意力转向她希望在部署阶段看到的确认活动。她希望确定系统有充分的文档记录，而且用户接受过有关系统正确使用的良好培训。她还想确定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：
  + 使用程序审查；
  + 内部培训；
  + 安装资格；
  + 操作资格与性能资格；
  + 操作人员认证。

**制定维护计划**

由于存在高残留风险，玛丽很关心软件的维护问题。她计划在部署系统完毕后进行多项维护活动确保软件的质量，包括评估用户培训的有效性、评估系统监控技术，评估正确性检查系统输出和缺陷报告。她还确信，除软件维护活动外，标定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，玛丽拼命要求原有系统退役。新系统采用通用数据格式，以便在退役时可以灵活地将剩余数据转移到新系统内。

**示例9：不合格材料报告系统——系统总体升级**

“高级医学专业公司”正在升级其不合格材料报告系统（NCMRS）软件。不合格材料报告系统（NCMRS）软件是一个商业软件包。在上一次主要版本升级后，“高级医学”决定不再升级，所以正在运行的系统落后两个主要版本。（“高级医学”目前正在运行的是第2版本，而最新版本是第4版本。）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软件的第4版本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—服务器应用程序平台变成了基于网络的应用程序平台。新软件还包括重要的新特点和新功能。弗兰克、业务流程所有者和“高级医学”的项目经理对弗兰克的现有软件和流程没有新的要求，不过，弗兰克的确希望利用新的软件功能。

弗兰克咨询了监管团队，认定ERP系统和NCMRS之间的当前界面能够原封不动地保留，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，这个扩展界面宜受到彻底质疑。弗兰克和同事们以及制造质量工程师和监管团队开始着手确定确认工作的范围。本示例的其余部分，该小组称为“团队”。

**定义流程**

弗兰克首先分析了当前的手动流程，以确定新软件将使工作流程的哪些元素实现自动化。新软件将会更改以下功能：

1. 识别潜在不合格材料或产品（超出范围）；
2. 输入与该物质有关的信息以及证明其周围情况的信息（范围内）；
3. 传递信息，以便正确识别、评估、调查和处置材料（范围内）；
4. 向重要利益相关方以及必须正确处理财务、采购、计划和调度交易的其他计算机系统分发信息（范围）；
5. 材料的实际处置，尽管有关处置的相关数据将记录在系统中（超出范围）。

**分析流程风险**

弗兰克意识到，流程和支持软件存在风险。流程故障可能产生严重后果，包括：

* 因疏忽大意使不合格材料流入制造过程；
* 因疏忽大意使不合格产品流入商业销售；
* 由于废料返工等原因导致成本增加或制造增加。

弗兰克和监管团队考虑了采取哪些风险控制措施缓解这些风险，包括：

* 通过程序控制检测隔离控制并纠正不合格材料；
* 对统计过程控制数据以及其他措施进行管理和质量审查，以确定能在流程未得到适当控制时发出信号的发展趋势；
* 对操作员进行持续培训，确保与程序保持一致；
* 财务报告，以帮助识别材料使用情况，这会揭示制造过程存在的异常问题。

有了这些风险控制措施，只有多个系统故障同时发生，才能出现不合格材料或产品无法适当控制。但是，由于此类故障可能对质量、监管和财务产生影响，弗兰克确定，残余流程风险需要严格的信任建立活动，以帮助确保软件正常运行并满足预期用途。

**定义软件目的和意图**

弗兰克希望详细了解软件升级对其用户和组织的影响。弗兰克得出结论，该软件本质上是一个自动化问题跟踪管理工具。使用标准工具、设备和其他仪器的生产人员负责识别和隔离有可能不合格的材料和产品。一旦某个问题得到确认，将有关情况的详细信息就会输入软件。然后，软件操纵工作流程、作业要求和通知解决该问题，并且记录材料和产品处置所需的各种活动。软件升级宜简化流程，从而提高效率，而且流程宜为质量保证团队提供更强大的分析工具和趋势数据，并使团队更好地了解质量问题。弗兰克了解，升级所需的流程变更主要针对的是工作流程和信息分发。软件本身既不做最终决定，也不独立确定任何结果，但实实在在地保存并记录人机交互做出的决策。

弗兰克确定，软件将使不合格处理工作流程的方方面面实现自动化。以下是监管团队撰写的关于软件目的和意图的陈述：

* NCMRS软件旨在支持不合格材料及产品的加工。该系统用于记录SOP定义的流程步骤，并且记录所执行的流程步骤、步骤执行时间、步骤执行人以及每个步骤的结果。该系统使数据随时可供质量监控和改进活动使用。

**定义与其他系统之间的软件边界**

NCMRS软件有两个界面，包括一个带有ERP系统的主界面和一个带有公司人力资源（HR）系统的二级界面。主界面设计为每日预定两次批处理的过程，以便使用成品数据、在制品数据、物料清单数据和操作清单数据进行系统更新。该界面反过来还将向ERP提供不合格材料报告（NCMR）数据，其中包含有关质量保留、材料处置和其他交易信息的数据。辅助接口是单向的，仅接受来自HR系统的数据，旨在更新NCMR员工数据，用以进行调度和分配。

**制定初步确认计划**

弗兰克还深信，对NCMR流程和软件理解充分，记录正确。他现在做好了制定高级确认计划的准备。在计规过程中将补充更多细节。

监管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“要求”，但不是典型的用户要求本身。相反，这些文件将是一系列关于软件运作方式的详细描述。因此，自动测试和人工测试分析从审查和结果的角度看将更加定性。自动测试和人工测试分析将从整体上看待输出结果以及系统是否按照预期运行，确定是否已满足特定用户要求，而非着眼于单个测试。

该文件集将包括以下内容。

1. 工作流程与业务规则文档。软件的这个区域是可配置的，因此团队将准备软件自己想要制作的所需配置集，并将开发进行操作描述的详细工艺流程和逻辑图。
2. 接口文档。这些文件将描述哪些数据元素从ERP和HR系统转移入NCMRS、哪些数据元素从NCMRS移入ERP系统以及何时转移这些元素。
3. 数据迁移文档。这组文件将描述将哪些历史数据移入升级后的系统。

确认计划将包括每个文件的审批结果。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。由于该系统危险程度很高，并且存在风险，高级管理层的所有成员都宜参与确认报告最终验批。

**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。

**建立对软件的信心和控制力**

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信需要完成的五项主要工作：

1. 根据“高级医学”的内部政策和程序，供应商处在获公司批准状态。以往审计表明，该供应商拥有合格的质量体系和SDLC。供应商生产的商用软件，在受监管行业已确立了历史地位，可满足与“高级医学”类似的预期用途。该供应商将定期接受审核，以保持该批准状态。
2. “高级医学”将采用供应商提供的自动化测试工具，确认软件是否已正确安装、是否正在测试套件的范围内运行。该工具可在几个小时内处理8000多种不同的业务。但是，该工具不会测试公司计划包含的某些特定配置选项。
3. 该团队将制定一项附属测试计划，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。处理结果将接受审查，确保结果准确、数据完整性并合乎程序规定。
4. 该团队将采用抽样技术，确认现有系统记录的数据转换和迁移，确保历史记录保持其完整性。记录计数将用于确认100％转换。
5. 数据接口将用某种采样技术加以确认，以衡量数据传输的完整性和准确性。

**分析软件故障风险**

弗兰克使用此文件作为参考，确定将要求达到的确认严谨程度。软件故障可能导致电子记录丢失或处理不当。风险的降低通过供应商的内部质量系统、软件安装认证预审（自动化测试工具）以及附加用例测试及确认控制。由于存在下游流程控制，该系统的剩余风险程度被认为合理可行。

**制定最终确认计划**

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：他们已经针对方法实施、针对自己已经分析过的软件风险以及已经获得的用于进行详细确认计划的足够信息充分定义了系统要求。

要进行的大部分测试将利用某种自动化测试套件完成。此自动化测试大件经团审核，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附助用例测试。测试目的是：a）确认流程是否按预期工作，b）加速用户验收与培训；c）确认配置变更是否未对软件产生不利影响。附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确认。自动化测试成功完成将证实软件已正确安装，且在功能上可以接受。

团队从本文档中选择了以下工具执行其余安装、配置、测试、确认和确认工作。

* 设计、开发和配置工具：
  + 架构文档与审查；
  + 确定软件系统设计中的风险控制措施；
  + 配置设计评审；
  + 审查供应商的“已知问题”清单；
  + 审查供应商的基本系统确认文档；
  + 审查“开箱即用”的软件工作流程图；
  + 审查“开箱即用”的标准报告库；
  + 对标准工作流程和业务规则进行配置更改差距分析。
* 测试工具：
  + 测试计划；
  + 供应商提供的安装、确认和资格认证自动化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；
  + 确认操作界面的系统检查。
* 部署工具：
  + 使用程序审查；
  + 内部培训；
  + 操作人员认证。

**制定维护计划**

弗兰克计划在部署系统完毕后，进行多项维护活动，以持续保持软件质量，包括评估用户培训系统监控技术的有效性、定期审核内部系统输出和缺陷报告以及供应商系统输出和缺陷报告。弗兰克已与供应商建立了联络点，以便错误维护版本通知和其他信息，能够引起“高级医学”内部相关软件软件维护负责人员的注意。

**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，以此作为比较吞吐量和结果的机会，从中可以编译性能度量。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：用于安排不合格材料报告（NCMR）审查委员会会议的软件**

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审核活动。受命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根据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审查委员会的会议。

**定义流程**

项目组聚在一起，讨论NCMR审查委员会会议的安排流程，审查公司的NCMR处理程序。讨论产生了以下规定流程：

1. 一旦识别出某个不合格现象，相关材料上要贴上标签、立即隔离，并载入经过确认的NCMR数据库的正式记录。
2. 每周举行一次会议，审查与不合格和推荐处置活动有关的所有调查结果。
3. 每次会议都要确定准备接受审查的NCMR清单，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。
4. NCMR审查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。邀请里包括要讨论的不合格材料报告（NCMR）的列表。

**流程风险分析**

通过头脑风暴活动，项目组成员评估该流程可能出现的故障可能造成的伤害：

* 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；
* NCMR列表内所确定的审查对象不正确。

项目组指出，已发布的NCMR处理程序要求指定一个人担任NCMR处理经理。此人负责确保所有NCMR报告及时处理，并负责根据经过确认的NCMR数据库中的数据，公布NCMR报告处理指标。在项目组确定的所有用例中，会议安排软件造成的损害都是NCMR审查委员会会议效率中断。这种中断给NCMR处理经理的时间带来了额外负担。因此，从监管风险、环境风险和人类伤害风险的角度看，流程失败风险分析确定为风险较低。

**定义预期用途**

项目组定义了软件使用的目的和意图、管制用途和边界，见下文：

* 软件使用
  + 由谁使用？软件主要由NCMR处理经理使用。
  + 干什么？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地，所以软件只需要在局域网（LAN）上使用。
  + 如何使用？软件检索出一份处于开放状态的需要NCMR评审委员会审查的NCMR列表。NCMR处理经理确定将在下次会议上审查的NCMR。然后，软件用NCMR处理经理设置的表格，识别需要参加特定会议的个人。会议日期由NCMR处理经理确定。软件提前一天向合适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审查委员会周会的时效。
* 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。

在创建并审查目的与意图说明后，项目组断定，所提议的软件既不会自动执行法规要求的某一活动，也不会创建法规要求的质量记录。虽然软件所推进的会议是某项受监管活动（NCMR流程）的一部分，但软件本身并不会使某个受监管活动实现自动化。因此，项目组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。但项目组还意识到，维护阶段用法发生微小变化都可能会严重影响项目组的原始确认决策。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审查的参会人员列表，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，更新后的质量体系程序包括定期评估预期用途或定期评估由于相关流程引起的变化。

**工具箱使用**

使用了以下工具：

* 开发—定义阶段：
  + 流程要求定义；
  + 过程故障风险分析；
  + 预期用途定义。
* 维护阶段：
  + 制定维护计划。

**讨论**

通过识别该软件自动化的特定用途和活动边界，项目组能够适如其分地声明：该软件不符合医疗器械质量管理系统流程的软件定义，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实际用途。另外，即使没有软件的改变，预期用途也可以在生命周期的维护阶段轻易改变，认识到这一点也很重要。因此，制定维护计划是确保公司所用软件得到合同控制的重要一环。

**示例11：已认可的供应商列表系统**

颠峰公司是一家B类医疗器械制造商。公司一直使用手动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准备提供特定部件。

* 颠峰公司新AVL系统项目经理杰克断定，AVL流程是与ISO 13485标准中的购买控制相关的医疗器械质量体统流程。

因此，所提出的AVL系统属于软件确认要求的范畴。

**定义流程**

为了更好地了解开发AVL系统所涉及的要求与风险，杰克确定了以下相关业务流程：

1. 当工程小组希望报批新的供应商时，该供应商零件的样品将提交给质量组进行资格预审。
2. 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，批准他们将供应商名称和核准的部件号以及部件描述输入已批准的供应商列表（AVL）中。此列表在采购组保留在纸质文件上。接收检查组可以访问AVL列表。
3. 采购组通过人工检查证明供应商的名称已正确添加到AVL列表内。
4. 在订购零件时，采购组会参考AVL列表，确保供应商已获得批准，有权提供所需零件。
5. 如果供应商是获得批准的供应商，则采购组在订购申请书上签字，表明采购组已经检查了AVL列表。

**分析过程风险**

然后杰克考虑当前流程可能出现什么问题。如果流程瘫痪，零件可能会从未经批准的供应商处订购，原因可能是未经批准的供应商混入了AVL列表，也可能是因为采购组在订购零件之前未检查AVL列表。

然后杰克考虑为了减轻这些风险采取了哪些风险控制措施。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仅限于授权员工。此外，杰克还发现，当前的采购程序规定，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。确保订单发给已批准的供应商，其控制权属于接收检查部门，接收检查部门在收到零件时要再次检查AVL列表。

杰克根据这些风险控制措施断定，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，杰克了解了要实现自动化的业务流程，他为所推荐的新AVL系统撰写了目的和意图说明：

* AVL系统将根据电子AVL列表自动检查供应商和部件，确保仅向授权供应商订购部件。新系统将使用AVL数据库，该数据库将链接到现有采购订单系统，在供应商资格认证过程由颠峰公司总部的质量小组使用，在采购订单生成过程由采购代理使用。

杰克还考虑了AVL系统将与之连接的其他系统和流程，并在自己的陈述中添加了一些文字，说明新系统的界限。

* 购买流程将与AVL系统自动化流程相关联。其界面将包含一个AVL数据库询问子系统，用来查询采购订单上指定供应商的状态。采购流程不确认AVL列表数据的准确性，也不与供应商评估流程相关联。

**制定确认计划**

现在杰克了解了有待自动化的业务流程，也确定了新系统的目的和意图，他已做好了在高层次上制定有效计划的准备。他知道，自己将在稍后更详细地充实这个计划，但现在就想开始着手制定确认计划，这样他才能够确定所需确认工作的级别。

早些时候，杰克断定，现有AVL流程存在低残留流程风险。因此他认为，他不需要在确认工作中提供非常详细的内容或程式。他知道，为新系统定义用户业务流程要求和软件要求非常重要，但由于该系统风险较低，杰克不需要要求独立文件的每个文件档都单独签署。因此，他决定以表格的格式将用户业务流程要求、软件需求和自己制定的测试计划合并到一个简单的文件里。

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全面管理审查。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，为了确保用户要求正确，他还应该加上采购组代表评审这个环节。

杰克根据自己的判断开始了着手起草确认计划。颠峰公司有所有确认计划宜采用的标准格式。确认计划的某些部分没有做规定，但杰克会在初始系统设计获得批准后更新此计划。

**确定软件要求**

杰克现在编写了软件要求。他认为，软件要求宜包括：“行动内容”（AVL流程或系统要求采取的行动）、有关AVL系统与采购系统关联方式的接口规范、数据字典，以及新系统应该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系统需要与之关联的其他系统。他断定，唯一的界面将是与巅峰现有采购系统的关联界面。该界面可以通过结构化的简单查询语言（SQL）查询系统，查询AVL数据库。

**建立对软件的信心和控制力**

杰克现在需要决定应该使用什么方法、采用什么技术构建新系统。因为业务需求相当简单，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[[1]](#footnote-1)开发这个系统。Microsoft Access是一种广泛普及，而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建立对Microsoft Access的信心。杰克发现，Microsoft Access的使用很普遍，而且过去没有在Internet留言板上快速识别和公布该产品的任何问题。再加上AVL系统风险较低这一事实，杰克认为，自己作为数据库开发人员，不需要对Microsoft进行供应商审计。

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

**分析软件故障风险**

虽然杰克已经确定自动化的业务流程风险很低，但他仍然需要分析软件故障的风险。他决定对此活动使用定量风险模型（比例为1到10）。他按如下方式对新系统进行排名。

* 杰克将“严重性”列为中等（6），因为软件故障只会间接造成伤害。他的排名基于流程中下游控制的存在。
* 杰克将“可能性”排在低位（1），因为数据库设计非常简单，使得在测试期间不会发现关键错误的可能性降低。
* 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，已经决定了要实施方案并且分析了软件的风险，他有足够的信息完成确认计划。在这一点上，他抽身出来，根据自己对该系统、对实施方法和软件风险的了解，向自己提出了这样的问题：“哪些确认活动会真正让我相信这个系统适合其预期用途？”。

由于该系统是买来的数据库工具，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，确保操作系统变更和Access版本更改得到很好的控制。他在确认计划内补充了用来调用正式软件配置控制的内容。

由于该系统是第三方开发的系统，杰克需要确信开发人员正确地转换了自定义要求、输入、接口、数据存储和输出要求。因为该系统将依赖于现有系统提供的信息，杰克作为一项重要活动在确认计划中添加了接口测试和集成系统测试，用以确认开发人员工作的正确性。

最后，杰克希望确信开发人员在开发期间保护着正确的版本控制，因此他在其确认计划内作为一项必要活动，添加了软件版本控制。

因此，杰克的批判性思维使他为其余开发确认工作准备了以下工具：

* 设计、开发与配置工具：
  + 软件架构文档与审查；
  + 可追溯性矩阵（要求说明内固有的）；
  + 风险控制措施（记录在用户规范中）。
* 测试工具：
  + 集成测试（记录在要求说明中）；
  + 接口测试（记录在要求说明中）；
  + 软件系统测试（记录在要求说明中）。
* 部署工具
  + 用户程序审查；
  + 应用程序内部培训；
  + 安装资格。

**制定维护计划**

杰克提前考虑了系统部署完毕后要确定软件质量可能适合进行哪些活动。鉴于系统剩余风险较低，杰克认为宜每季度审查数据库中AVL数据的准确性。杰克在其确认计划中专门留出了一个章节，记录季度审核情况，并要求开发并实施一个程序，确保在系统实时运行状态下季度审核正常进行。

**示例12：标定管理软件**

XYZ医疗公司正在迅速发展。XYZ已经在欧洲和亚洲购买了公司。该公司的成长意味着XYZ的标定管理需求也在增长。目前，XYZ标定经理保存着一本包含所有标定信息的记录本，每周检查标定设备库存，确定是否任何项目需要重新标定。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，靠纸质系统一个人很难进行管理。是时候建立一套计算机化的系统了。

**定义流程**

XYZ医疗公司有一套标准的操作程序（SOP），要求使部分质量体系自动化的计算机化系统必须接受预期用途确认。XYZ首先定义了标定管理流程，弄清流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经理审查了有关标定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：

1. 采购新设备；
2. 赋予新设备唯一的标识（10）号码；
3. 标定新设备；
4. 标定状态记录在设备上；
5. 维护标定记录，包括标定要求、状态和失效日期；
6. 搜索标定记录，以便上报和开展标定管理活动。

**流程风险分析**

无论是纸质系统还是电子系统，标定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。

* 如果某台设备在标定期满后仍在使用，该设备所记录的测量值就会不正确。这个问题可能会带来储多后果，具体取决于设备以及该设备所处使用流程处在哪个阶段。
* 一台设备上打上不正确的标签，将表明该设备接受标定的时候超出了实际标定期限。此错误也会产生诸多后果，具体取决于该设备以及该设备所处使用流程处在哪个阶段。
* 标定记录能够丢失，而且能够出现标定过期的积压设备。这个问题能够造成工作延误。
* 如果标定状态记录不准确，则能导致过期设备投入使用。
* 如果两个套设备收到的识别号相同，则记录就会失去唯一性。

鉴于标定不正确存在诸多潜在结果，公司认为该流程风险很高。情况最糟糕的时候，标定过期的设备可能用于测量接受最终验收的医疗器械，而且该设备可能在不宜通过验收的状态下被认定合格。

为了减轻此问题，XYZ经理宜更新SOP的信息，在其中补充针对设备每个用户的指南。每个用户在使用前都应检查设备上的标定期限标签。在标定设备使用协议执行期间，每个用户均宜记录设备识别号和所用设备的标定到期日期。用户还要在如何识别需标定物品方面接受培训，懂得不要使用任何带有过期标签或没有标签的设备。

这些措施的实施使系统的剩余风险降至中等水平。XYZ经理认为，用户指南是一种适当的措施，但效果不足以将剩余风险降低。

**定义软件预期用途**

该软件系统不会执行标定活动；该软件系统将只是一个包含有关设备及其标定历史和标定状态的标定信息和数据的数据库。该软件系统将控制标定过程的b) 步骤、f) 步骤和g) 步骤。

XYZ经理认同XYZ将根据以下目的和意图进行系统确认：

标定管理系统用于为需要标定的设备提供识别号、打印标定设备使用的标签、存储标定结果数据、报告设备的标定状态。标定管理系统自动化ISO 13485要求内涵盖检查、测量和测试设备的部分。

**制定确认计划**

为了给确认活动创造条件，XYZ管理人员开始制定确认计划，他们首先为可交付成果设定了期望内容，为跨职能部门参与确认流程设定了期望值。他们记录了以下几个步骤：

* 为所选工具确定文档严谨程度。
  + 系统文档的严谨程度将为适中。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（管理层参与与审查和跨职能参与与审查）。
  + 鉴于该系统将在全球范围内用于标定管理，全球信息技术管理与运营管理在该系统确认过程以批准确认计划和系统确认报告形式受到关注是正常的。此外，新的现场设备管理人员将参与所有文件的审批。
* 在工具箱中选择“定义”工具：
  + 用户和业务流程要求；
  + 软件要求；
  + 软件需求正式审查。

**定义软件要求**

软件要求将包含以下元素：

* 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；
* 设备标签印刷的特定要求；
* 用户安全性与用户资料；
* 性能要求；
* 容量定义。

**建立对软件的信心和控制力**

XYZ经理对三家此类产品的供应商做了调查，确定其中一家供应商的产品最符合XYZ计划的预期用途。虽然该版本产品相对较新，但该系统的供应商在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心，但还会根据当前报告的问题进行已知缺陷分析，测试开发组还会对以往发布的几个版本进行新功能测试。

**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标定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（危害） | 如果软件出现故障，是否产品安全存在潜在风险？  是，始终存在。标定不合格的设备可能会被软件错误识别为标定设备。   * 患者伤害——是。如果测量中采用了标定不合格的设备，则患者可能会使用规格不合格的产品。 * 操作人员伤害——是。如果温度测量或用力不当，操作员可能被夹痛或受伤。 * 旁观者伤害——是。这种伤害是由设备造成的。 * 服务人员伤害——是。如果测量温度或力道不对，服务人员可能被夹痛或受伤。 * 环境危害——是。如果压力测量不当而且容器中含有对环境有害的物质，容器可能泄漏。 | 风险1  使用的是超出标定不合格的设备 |
| 1.2 产品安全（危害） | 如果软件用户出错，产品安全是否存在潜在风险？  是，任何情况下，如果用户输入不正确的设备标定数据（见1.1）。   * 患者伤害——是。 * 操作员伤害——是。 * 旁观者伤害——是。 * 服务人员伤害——是。 * 环境危害——是。 | 见风险1。 |
| 2.1 产品质量 | 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）？  是。产品可能会不合规格，因为标定有误的设备可能被软件误认为标定合格设备。虽然错误识别并非安全问题，但可能引起客户不满。 | 见风险1。 |

**Table C.16** *(continued)*

**表C.6**（续）

|  |  |  |
| --- | --- | --- |
|  | **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）** |
| 2.2 产品质量 | 如果用户犯错，产品质量是否存在潜在风险（安全风险除外）？  是。如果用户为该设备输入错误标定数据，而且该设备又用来测量某一产品，则该产品可能不符合规格。虽然规格不正确并非安全问题，但可能引起客户不满。 |  |
| 3.1记录整合 | 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？  记录丢失——是。标定记录可能丢失。  记录污染——是。可能引起标定记录损坏。 | 风险2——  标定记录丢失并导致合规性问题。  风险3——  标定记录已损坏并导致合规性问题 |
| 4.1证明符合ISO标准 | 合规性证明能力是否存在潜在风险？  记录丢失——是。标定数据可能丢失。  记录污染——是。可能引起标定数据损坏。 | 见风险2和风险3。 |

**表C.17——示例12——风险评估与控制**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **风险识别符** | **描述** | **严重程度** | **控制** | **剩余风险** |
| 风险1 | 不定不合格设备用于产品测量或用于测压或测力。（风险由于软件错误识别设备或由于用户输入错误设备标定数据而产生。） | 高 | 系统的设计目的是用于打印包含设备识别号、序列号以及标定状态和截止日期的标签。在程序上，员工在使用设备之前接受培训，以便确认此信息。另一个过程要求由第二人确认在提交输入的数据之前对其进行确认。 | 可接受 |
| 风险2 | 记录丢失，标定管理活动无法得到保护。 | 中 | 所有标定数据都保存在标定室的纸质记录中。 | 可接受 |
| 风险3 | 记录受污染，标定管理活动无法得到保护。 | 中 | 所有标定数据都保存在标定室中的纸质记录上。 | 可接受 |

风险分析完成后，XYZ管理人员确信，缓解后的剩余风险是可以接受的。

**完成确认计划**

为了完成确认计划的制定，管理人员修改了计划，使其包含了以下工具的选项。

* 实施工具：
  + 可追溯性矩阵；
  + 审查系统配置。
* 测试工具：
  + 尺寸分析；
  + 测试规划；
  + 供应商提供的测试套件，并对计划配置以及以往版本或软件的新功能进行额外测试。
* 部署工具
  + 应用软件内部培训；
  + （服务器和工作站）安装资格。

**制定维护计划**

除了制定系统确认计划之外，XYZ管理人员还认为制定系统维护计划很有益，因为某一时刻系统肯定需要维护。

系统监测技术将到位，以检查所有缺陷、使用问题以及预期用途的变化。

制定一项计划，对系统的变化（即硬件、升级、补丁、安全问题）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。他们了解加里自动化区域生产的产品（长1.27厘米到3,81厘米不等的金属棒），所以他们为这种金属棒找到了两个应用程序：其中一个程序棒长不超过2.54厘米；另一个的棒长为3.18厘米（±0.64厘米）。棒材的宽度均为0.32厘米。两种应用程序都适用于医疗器械，要求金属棒为某一指定长度。加里是自动化领域的工程师，任务是进行确认零件分捡的自动视觉系统。该系统即将取代某个手动测量/分拣过程。该过程再无其他变化，因此这就是确认的全部内容。

**过程描述**

两种应用程序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确认。所有验收标准均在上游确认，但棒长除外，棒长使唤用加里的自动视觉系统进行测量。

机器的工艺很简单。将棒装入一个容器中，在容器内用漏斗将棒放在传送带上，一次一根。每根金属棒都被传送到一个停靠点，在那里，摄像头检查金属棒，测量其长度。然后根据检查测量结果，把棒输送到两个容器中：其中一个用于接受长度不大于2.54厘米的金属棒，另一个用于接受更长的金属棒。

下游没有再一次检查棒的长度。如果棒的尺寸用错，对患者造成伤害的风险则会增加，因为棒的尺寸不对，就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。设备已经生产多年，对该风险的了解非常深刻。自动视觉系统正在取代手动测量过程。

**定义预期用途**

因为了解此过程正在实现自动化，所以加里从定义目的和意图开始。

* 该软件用于确认某单根金属棒位于传送带上，并测量其长度。

**风险分析**

加里用本地风险分析过程确定的系统故障风险很高，因为在棒的尺寸用错的时候，除了产品出故障或进行破坏性测试之外，再无其他办法进行检测。此故障可能导致患者受伤。该过程的关键参数是棒的精确长度尺寸。自动化既不会增加此风险，也不会降低此风险。

**制定确认计划**

在首次迭代确认计划时，加里计划采用严格的确认过程（这是由于其风险分析得出的风险评级很高的缘故造成的）。在审查了工具箱中的潜在确认工具之后，他计划了一份正式的需求定义文档，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，但开发基于以往的系统自动化过程，因此开发比较简单。

**风险控制措施**

确定了两个值得重点关注的风险领域。

1. 需要确认有一根金属棒已经到位准备进行测量。机器沿着狭窄的通道传送金属棒，测量宽度为0.64厘米，高度0.48厘米。因此，金属棒只能竖着进入通道，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以前后紧贴在一起。

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（以往检查的规格为±0.08厘米），则会拒绝接受该金属棒，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）。

1. 金属棒有可能因为彼此贴在一起在一根结束另一根开始时识别不出来。软件会将把任何无法确认其等于或小于3.81厘米的棒形物送入废料箱。

**确认任务**

接下来，加里转向确认任务。他确定需要一份正式设计文件，并计划由进行要求审查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根据设计进行审查。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。自动化工程师、制造工程师和质量工程师都将按要求审查软件的可追溯性以及反馈给要求的设计方案。测试后，他们还会重复相同的步骤，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计划，而测试计划将包括软件环境的详细信息和预期测试结果。他在开发的各个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。将采用正常测试案例和错误测试案例以及与传送带速度相关的性能测试。除加里之外，测试计划还需要由其他自动化工程师、制造工程师和质量工程师审核批准。测试报告包括实际测试结果，与预期结果，得出的合格/不合格指示、一份测试辨识和问题解决文档，以及任何故障的回归测试情况。该测试报告，加里需要得到同一组人员的批准。

**实施、测试与部署**

针对该自动视觉系统的部署，加里检查了工具箱中的部署工具，认为需要进行安装资格认证。此外，他还认为应创建一个用户程序，系统用户还必须通过运营商认证。

**保养**

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。该领域不需要特别规划或专门采取行动。

**示例14：挑选和安放系统**

“高品质医疗公司”是一家乙级医疗器械生产厂家。“高品质”希望将部分成品零件从某一一位放入盒（公司生产医疗器械的一部分）内的人工放置过程变成自动放置过程。

吉尔是这个新型挑选放置（简称双P）系统的项目经理。根据ISO 13485标准，吉尔断定双P过程是个医疗器械质量体系过程，因为此过程是某个医疗器械制造过程的一个部分。因此，拟议的双P系统将不符合软件确认要求。

**定义当前流程**

为了更好地理解开发双P系统所涉及的要求和风险，吉尔定义了以下相关业务流程：

1. 制造过程第11工位的部件放入第12工位的盒子中（每个盒子的速率是20个部件）。目前，此操作由一名操作员手动完成。
2. 然后操作员手动将该盒子放在第12工位的进入轨道上。
3. 操作员手动检查盒子，以确认零件放置正确。【步骤b）和步骤c）每装一个盒子需要3分钟。】
4. 盒子继续进入其他组装步骤，其他组装步骤包括目视检查，以确认该过程所有已通过的步骤没有畸形出现。

**分析过程风险**

接下来吉尔考虑当前流程可能出现什么问题。分析表明，可能发生以下事件：

1. 操作员可能使半成品变形。这种畸形将在下游检测工位检测出来。
2. 操作员可能误把零件放入盒中，或者可能错过盒中的某个插槽。放错盒或错过插槽的检测目前在12工位手工完成。

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

**定义新流程**

在评估了流程风险之后，吉尔根据自己对双P系统的理解，为新流程定义了以下流程：

1. 双P系统将装有盒子。
2. 双P系统将从第11工位拾取零件，并将其插入盒子（以每盒20个部件的速率插入）。
3. 双P系统将目视检查盒子，确保部件全部放置正确，而且盒内所有插槽都已填满。任何不合格的盒子都会自动拒收。
4. 双P系统将合格盒子放在第12工位12上。【现在从步骤b）到步骤d）需要1分钟。）
5. 盒子将继续进入其他组装步骤，其他组装步骤包括目视检查，以确认该过程所有已通过的步骤没有畸形出现。

**定义软件预期用途**

现在吉尔了解要实现自动化的过程，做好了为拟议新型双P系统编写目的与意图说明的准备。

* 双P系统将拾取来自第11工位的部件，并将其放入盒中。该系统将确认所有盒槽已正确填满，拒绝任何不合格的盒子，然后将盒子以每分钟一个盒子的速率移动到第12工位的输入线上。

然后，吉尔考虑双P系统是否会与其他系统连接。结论是，没有其他互动。她断定存在用户界面，但无软件界面。

**制定确认计划**

分析了要实现自动化的业务流程并且确定了新型系统的目的和意图之后，吉尔已做好了在高层次上制定确认计划的准备。她稍后需要添加更多细节，但如果现在开始制定确认计划，她就能够确定所需确认工作的级别。

早期吉尔已确定，当前流程残留风险较低。因此，她感觉确认工作不需要多少细节或呈式。吉尔知道为新型系统定义用户业务流程要求和软件要求非常重要。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流程要求、软件需求和测试计划合并成一个文档。

由于新系统的风险相当低，吉尔断定不需要对确认工作进行广泛的管理审查，制造经理和质量保证代表批准就已足够了。但是，为了确保用户要求正确无误，她在流程中添加了由有代表性操作员进行审查的步骤。

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。

**定义系统与软件要求**

然后吉尔转向系统与软件要求。她决定软件要求要包括双P过程或系统步骤以及关于双P系统与第11工位和第12工位连接方式的接口规范。系统要求包括双P系统移动的速度和准确性。为了降低伤害风险，吉尔增加了一项安全要求，在操作员和双P臂之间提供了一道物理屏障。

**建立对软件的信心和控制力**

吉尔应该现在决定她应该使用的购买新系统的方法和技术。鉴于业务需求非常简单，交易量会很低。由于新系统风险较低，吉尔决定购买一套第三方双P系统。出于价格和质量原因，吉尔决定向Controlsys公司购买双P系统， Controlsys公司是双P系统行业的行业龙头。

Controlsys是一家外部系统供应商。因此，吉尔应该现在决定她应该采取哪类活动建立对Controlsys的信心。她评估了自己手里有关Controlsys的信息。吉尔指出，Controlsys的双P产品使用范围广泛，使用记录很强大。过去，产品的问题已快速识别出来，并公布在互联网留言板上。对此信息的审查表明，只有少数较小的已知问题，因此吉尔认为，这些问题与她预期软件的用途无关。此外，Controlsys还提出一种自动化安装资质/操作资质/性能资格认证（IQ/OQ/PQ）的测试套件。鉴于该公司的历史同时考虑了双P系统风险较低的事实，吉尔认定不需要对Controlsys进行现场供应商审核。她批准了Controlsys的供应商资格。

**分析软件故障风险**

吉尔已经断定要实现自动化的业务流程风险较低，但仍然需要分析软件故障风险。吉尔决定使用定量风险模型，按照如下标准为新系统打分：

* 按1-10分打分，吉尔给“严重性”的打分较低（3分），因为软件故障将在下游活动中检测出来。
* 她给“可能性”的打分很低（1分），因为系统设计非常简单，使得测试期间不会捕获所有关键错误的可能性降低。
* 她计算出风险评分为4分，4分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要求、选择了实施方法，并且分析了软件风险。因此，她有足够的信息完成确认计划。

由于所提议的系统具有较低残余风险，因此吉尔为剩下的开发确认工作选择以下工具：

* 设计、开发和配置工具：
* 软件架构文档和审查；
* 可追溯性矩阵（并入要求说明）；
* 风险控制措施将记录在用户规范中。
* 测试工具：
* 集成测试（记录在要求说明中）；
* 接口测试（记录在要求说明中）；
* 软件系统测试（记录在要求说明中）。
* 部署工具：
* 用户程序审查；
* 应用程序的内部培训；
* 供应商提供的测试套件（来自Controlsys）。

**制定维护计划**

吉尔现在考虑系统部署完成后哪些活动适合保证系统质量。由于系统残留风险较低，因此她在将运动机制标定工作加入标定计划时，采纳了制造商的建议。吉尔将该系统放在了该公司最长的确认审核周期（3年）。

**批判性思维审查**

最后，吉尔问自己，自己是否已经考虑了确保自己对自己的确认方法具有足够信心的所有要素。结论是，自己选择和完成的确认活动提供的信心水平可以接受，相信软件将按预期执行。

1. Microsoft Access是市场上可以买到的适用产品的实例。此信息是为了方便本文件用户而提供的，并不构成ISO对本产品的背书。 [↑](#footnote-ref-1)