ICS 25.040
N 10

中华人民共和国国家标准

GB/T 20438.3—2006/IEC 61508-3:1998

电气/电子/可编程电子安全相关系统的功能安全 第3部分:软件要求

Functional safety of electrical/electronic/programmable electronic safety-related systems—Part 3: Software requirements

(IEC 61508-3:1998,IDT)

2006-07-25 发布　　2007-01-01 实施

中华人民共和国国家质量监督检验检疫总局
中国国家标准化管理委员会　发布

目　次

前　言

GB/T 20438 由下列 7 部分构成：

——第 1 部分：一般要求；

——第 2 部分：电气/电子/可编程电子安全相关系统的要求；

——第 3 部分：软件要求；

——第 4 部分：定义和缩略语；

——第 5 部分：确定安全完整性等级的方法示例；

——第 6 部分：GB/T 20438.2 和 GB/T 20438.3 的应用指南；

——第 7 部分：技术和措施概述。

本部分是 GB/T 20438 的第 3 部分。

本部分等同采用国际标准 IEC 61508-3:1998《电气/电子/可编程电子安全相关系统的功能安全 第 3 部分：软件要求》(英文版)。

本部分的附录 A、附录 B 为规范性附录。

本部分与 IEC 61508-3:1998 在技术内容上没有差异，为便于使用做了下列编辑性修改：

a) 将“IEC 61508”改为“GB/T 20438”。

b) 本“国际标准”一词改为“本标准”。

c) 删除国际标准中 1.2 的注 2，因为此注所表述的是 IEC 61508 在美国和加拿大等国的应用情况，与我国的实际不符，所以删除。

d) 用小数点“.”代替作为小数点的逗号“,”。

本部分由中国机械工业联合会提出。

本部分由全国工业过程测量和控制标准化技术委员会(SAC/TC 124)归口。

本部分由机械工业仪器仪表综合技术经济研究所负责起草。

本部分主要起草人：王莉、冯晓升、梅恪、郑旭、欧阳劲松等。

引　言

由电气和电子器件构成的系统，多年来在许多领域中执行其安全功能，以计算机为基础的系统（一般指可编程电子系统(PES)）在许多领域中用于非安全目的，但也越来越多地用于安全目的，为使计算机系统技术更有效安全地使用，有必要进行安全方面的指导。

GB/T 20438 针对由电气或电子和可编程电子部件构成的、起安全作用的电气/电子/可编程电子系统(E/E/PES)的整体安全生命周期，提出了一个通用的方法。建立统一的方法的目的是为了针对以电子为基础的安全相关系统提出一种一致的、合理的技术方针，主要目标是促进应用领域标准的制定。

在许多情况下，可用多种基于不同技术的防护系统来保证安全（如机械的、液压的、气动的、电气的、电子的、可编程电子的，等等）。从安全战略角度，不仅要考虑各系统中元器件的问题（如传感器、控制器、执行器等），而且要考虑构成组合安全相关系统的所有安全相关系统。因此 GB/T 20438 对电气/电子/可编程电子(E/E/PE)安全相关系统进行了规定。GB/T 20438 还提出了一个框架，在这个框架内，基于其他技术的安全相关系统也可同时被考虑进去。

在各种应用领域里，存在着许多潜在的危险和风险，包含的复杂性也各不相同，从而需应用不同的 E/E/PES。对每个特定的应用，则根据应用的不同而确定所需的安全量。GB/T 20438 仅是使这些量值规范化。

GB/T 20438

——考虑了当使用 E/E/PES 执行安全功能时，所涉及到的整体安全生命周期、E/E/PES 安全生命周期以及软件生命周期的各阶段（如初始构思，整个设计、实现、运行和维护到停用）。

——针对飞速发展的技术，建立一个足够健壮而广泛的能满足今后发展需要的框架。

——有利于促进 E/E/PES 安全相关系统在不同领域中相关标准的制定，各应用领域和交叉应用领域相关标准应在 GB/T 20438 的框架下制定，使之具有高水平的一致性（如基础原理，术语等的一致性），并将既安全又经济。

——为达到 E/E/PE 安全相关系统所需的功能安全，提供了编制安全要求规范的方法。

——使用了一个安全完整性等级，此安全完整性等级规定了 E/E/PE 安全相关系统要实现的安全功能的目标安全完整性等级。

——采用了一种可确定安全完整性等级要求的基于风险的方案。

——建立了 E/E/PE 安全相关系统的数值目标失效量，这些量都同安全完整性等级相联系。

——建立了危险失效模式中目标失效量的一个下限，此下限是对单一 E/E/PE 安全相关系统的要求。这些系统运行在：

1）低要求操作模式下，为了执行它的设计功能，一旦要求时，就把下限设定成平均失效概率为 10^{-5}；

2）高要求操作模式或者连续操作模式下，下限设定成危险失效概率为 10^{-9}/h。

注：单一 E/E/PE 安全相关系统不一定是单通道结构。

——采用广泛的原理、技术和措施以达到 E/E/PE 安全相关系统的功能安全，但不使用失效-安全的概念，这个概念是在很好定义了失效模式，并且复杂性相对较低时的一个数值。由于 E/E/PE 安全相关系统的复杂性均在 GB/T 20438 范围之内，因此不适用失效-安全的概念。

电气/电子/可编程电子安全相关系统的功能安全　第3部分:软件要求

1　范围

1.1　GB/T 20438的本部分:

a)　使用应建立在充分理解GB/T 20438.1、GB/T 20438.2的基础上。

b)　适用于任何在GB/T 20438.1、GB/T 20438.2范围内构成与安全相关系统的一部分有关的或用于开发安全相关系统的软件。这种软件定义为安全软件。

——安全软件包括操作系统、系统软件、通信网络中的软件、人机界面功能、支持工具、固件以及应用程序。

——应用程序包括高级语言、低级语言程序和使用有限可变语言的特殊用途程序(见GB/T 20438.4—2006的3.2.7)。

c)　软件安全功能和软件安全完整性等级的要求应明确。

注1:如果这一要求作为电气/电子/可编程安全相关系统(见GB/T 20438.2—2006的7.2)有一部分已提出,则在此处不需重复。

注2:规定软件安全功能和软件安全完整性等级是一个重复的程序,见图2和图6。

注3:文档结构要求见GB/T 20438.1—2006的第5章和GB/T 20438.1—2006的附录A。文档结构应考虑公司规程和特殊应用领域的工作实际情况。

d)　建立安全生命周期阶段和在设计、开发与安全有关的软件(软件安全生命周期软件模块)阶段和行为的要求。这些要求包括根据安全完整性等级分等的、在软件中用于避免和控制故障及失效的措施和技术的应用。

e)　对向执行电气/电子/可编程集成的机构提供与软件安全性确认有关的信息提出要求。

f)　对操作和维护E/E/PE安全相关系统的用户所需的与软件有关的信息和规程的准备提出要求。

g)　对修改与安全有关的软件的机构提出要求。

h)　结合GB/T 20438.1、GB/T 20438.2提出对支持工具的要求,如设计开发工具、语言翻译器、测试和调试工具、配置管理工具。

注4:图4和图6表示了GB/T 20438.2和GB/T 20438.3之间的关系。

1.2　GB/T 20438.1、GB/T 20438.2、GB/T 20438.3和GB/T 20438.4是基础的安全标准,尽管它们不适用于简单E/E/PE安全相关系统(见GB/T 20438.4—2006的3.4.4),作为基础的安全标准,根据IEC导则104和ISO/IEC导则51中包含的原则,各技术委员会在起草标准时应考虑使用这些标准,因为技术委员会的责任之一是在起草自己标准时凡是适用之处都应贯彻基础安全标准。GB/T 20438同时也可作为独立的标准去使用。

1.3　图1表示了GB/T 20438的整体框架同时明确了在达到E/E/PE安全相关系统功能安全阶段中本部分的作用。GB/T 20438.6—2006的附录A描述了GB/T 20438.2和GB/T 20438.3的应用。

2　规范性引用文件

下列文档中的条款通过GB/T 20438的本部分的引用而成为本部分的条款。凡是注日期的引用文件,其随后所有的修改单(不包括勘误的内容)或修订版均不适用于本部分,然而,鼓励根据本部分达成协议的各方研究是否可使用这些文件的最新版本。凡是不注日期的引用文件,其最新版本适用于本部分。

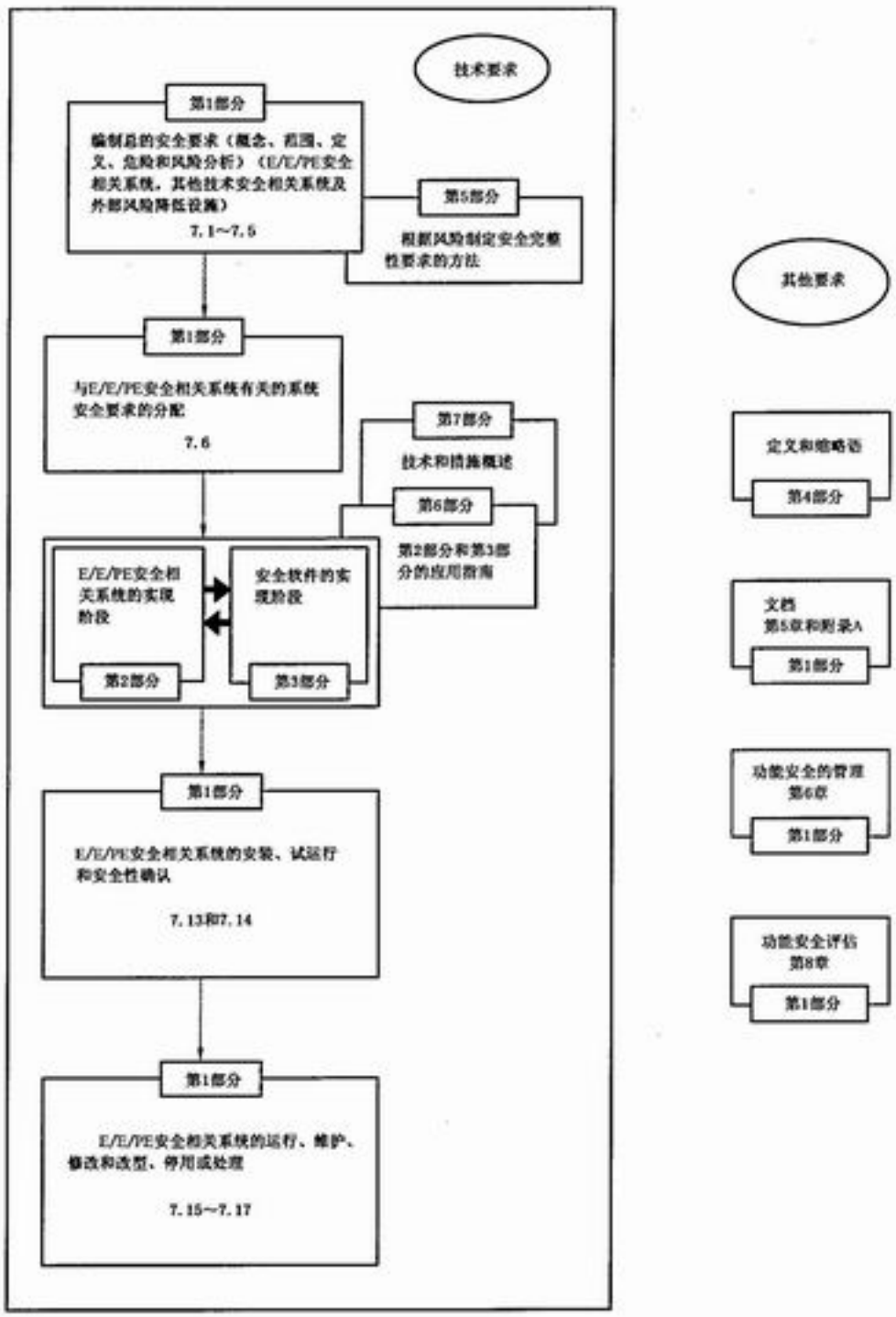


图 1　GB/T 20438 的总体框架

GB/T 20438.1—2006 电气/电子/可编程电子安全相关系统的功能安全 第1部分:一般要求(IEC 61508-1:1998,IDT)

GB/T 20438.2—2006 电气/电子/可编程电子安全相关系统的功能安全 第2部分:电气/电子/可编程电子安全相关系统的要求(IEC 61508-2:2000,IDT)

GB/T 20438.4—2006 电气/电子/可编程电子安全相关系统的功能安全 第4部分:定义和缩略语(IEC 61508-4:1998,IDT)

GB/T 20438.5—2006 电气/电子/可编程电子安全相关系统的功能安全 第5部分:确定安全完整性等级的方法示例(IEC 61508-5:1998,IDT)

GB/T 20438.6—2006 电气/电子/可编程电子安全相关系统的功能安全 第6部分:GB/T 20438.2和GB/T 20438.3的应用指南(IEC 61508-6:2000,IDT)

GB/T 20438.7—2006 电气/电子/可编程电子安全相关系统的功能安全 第7部分:技术和措施概述(IEC 61508-7:2000,IDT)

ISO/IEC 导则51:1990 安全方面 在标准中引入安全条款的指南

IEC 导则104:1997 安全出版物的编写及基本安全出版物和分类安全出版物的应用

3 定义和缩略语

见GB/T 20438.4。

4 标准的符合性

见GB/T 20438.1—2006的第4章。

5 文档

见GB/T 20438.1—2006的第5章。

6 软件质量管理系统

6.1 目的

见GB/T 20438.1—2006的6.1。

6.2 要求

6.2.1 见GB/T 20438.1—2006的6.2,以下为附加要求。

6.2.2 功能安全计划应定义E/E/PE安全相关系统的安全完整性等级所要求的软件获取、开发、集成、确认和修改的战略。

注:该方法的理念是在编制计划时考虑E/E/PE安全相关系统部件所要求的各种安全完整性,制定标准。本部分的7.4.2.8将考虑E/E/PE安全相关系统中使用不同安全完整性等级的组件时的情况。

6.2.3 软件配置管理

a) 应在软件安全生命周期阶段中使用行政和技术控制,以管理软件变化和保证有关软件安全的规定要求始终能得到满足。

b) 应确保所有必需的操作已被执行以说明获得了所要求的软件安全完整性。

c) 应保持精确的和维护E/E/PE安全相关系统完整性所必需的所有配置项的唯一识别。配置项至少包括:安全分析和要求;软件规范和设计文档;软件源代码模块;测试计划和结果;将要被安装于E/E/PE安全相关系统的已存在的软件组件和软件包;所有用于创建、测试或执行E/E/PE安全相关系统软件的工具和开发环境。

d) 应采用变更控制规程用于防止非授权的修改;对修改请求文档化;分析建议修改的影响以批准或拒绝请求;对所有准许修改的细节和授权文档化;在软件开发阶段中适当点建立配置基线,并对判断基线(部分)的集成测试文档化(见7.8);确保所有软件基线的构成(包括早期基

线的重建)。

注:为指导、加强行政和技术控制的使用,有必要进行管理决定和授权。

e) 应对下列信息文档化,以用于随后的审核:配置状态、发布状态、对所有修改的判断和通过、修改的细节。

f) 安全软件发布应正式文档化。软件的主要备份和所有有关文档在已发布软件的操作生命周期内应被保存以允许维护和修改。

注:对于配置管理的更详细的信息,见 ISO/IEC 12207。

7 软件安全生命周期要求

7.1 一般要求

7.1.1 目的

将软件开发纳入到规定的各阶段和活动中(见表 1 和图 2~图 5)。

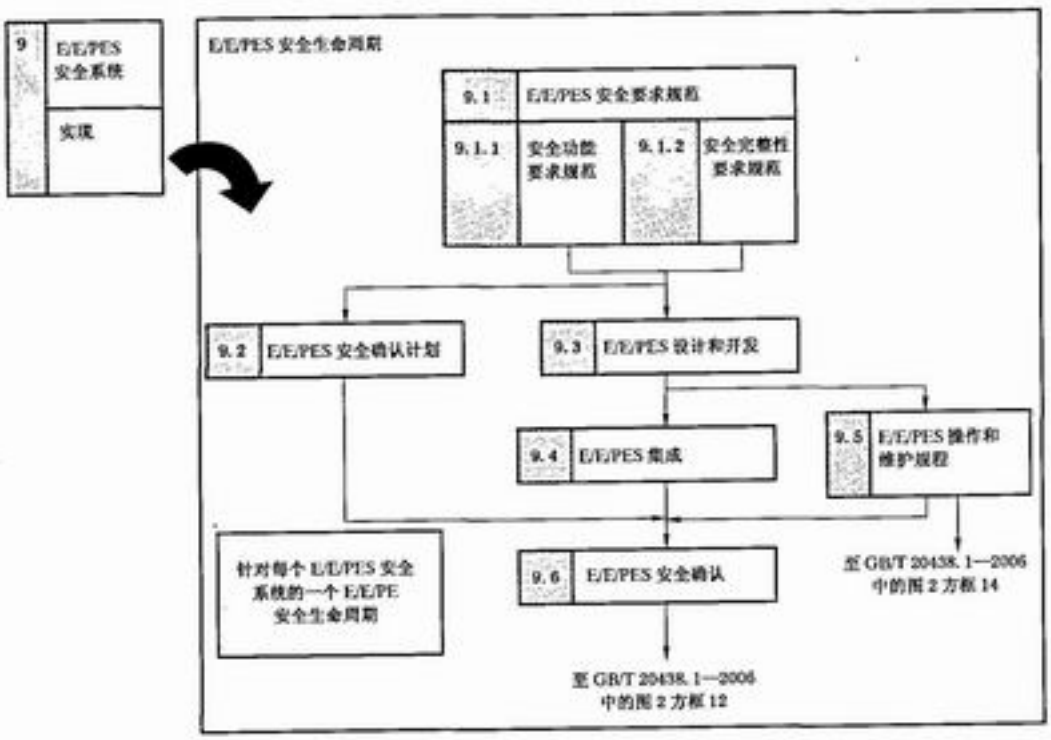


图 2 E/E/PE 安全生命周期(实现阶段)

7.1.2 要求

7.1.2.1 软件开发的安全生命周期应根据 GB/T 20438.1—2006 第 6 章在编制安全计划期间进行挑选和规定。

注:一个满足 GB/T 20438.1—2006 第 7 章要求的安全生命周期模型可根据某工程项目或机构的特殊需要来定制。

7.1.2.2 质量和安全保证规程应集成到安全生命周期活动中。

7.1.2.3 软件安全生命周期的各阶段应根据各阶段规定的范围、输入和输出分成基本的活动。

注 1:对于生命周期各阶段的更详细的信息,见 ISO/IEC 12207。

注 2:GB/T 20438.1—2006 第 5 章考虑了安全生命周期各阶段的输出。在开发一些 E/E/PE 安全相关系统的阶段中,一些安全生命周期阶段的输出可能是一个明确的文档,而从几个阶段输出的文档可被合并。基本的要求是安全生命周期阶段的输出应符合其预定的目的。在简单开发阶段中,一些安全生命周期阶段可被合并(见 7.4.5)。

7.1.2.4 假如软件安全生命周期满足图3和表1要求，允许根据项目的安全完整性和复杂性改变V模型中阶段的深度、数量和工作范围。

注：表1中所有生命周期阶段的列表适用于大型新开发的系统。在小系统中也可能适用，例如合并软件系统设计和结构设计阶段。

7.1.2.5 在本条所有的目的和要求可以满足时，允许以不同于GB/T 20438组织结构的其他方式编排软件工程项目(如使用其他的软件安全生命周期模型)。

7.1.2.6 对每一个生命周期阶段，应使用适当的技术和措施。附录A和附录B给出了推荐。只通过从附录A和附录B中选择技术不能保证能满足安全完整性的要求。

7.1.2.7 软件安全生命周期中的活动结果应文档化(见第5章)。

7.1.2.8 如果在软件安全生命周期的任一阶段，要求生命周期的前一阶段改变时，则应重复安全生命周期的前一阶段和随后的阶段。

表1 软件安全生命周期：概述

安全生命周期阶段		目的	范围	要求所在的条款	输入(要求的信息)	输出(产生的信息)
图3中的方框号	标题					
9.1	软件安全要求规范	根据软件安全功能要求和软件安全完整性要求规定软件安全要求： 对每个需实现一定安全功能的E/E/PES安全相关系统规定软件安全功能的要求； 规定每一个E/E/PES安全相关系统对于软件集成的要求，以保证获得这一E/E/PES系统分配的每一安全功能需达到的安全完整性等级	PES； 软件系统	7.2.2	E/E/PES安全要求规范(GB/T 20438.2)	软件安全要求规范
9.2	软件安全确认计划编制	拟定软件安全确认计划编制	PES； 软件系统	7.3.2	软件安全要求规范	软件安全确认计划
9.3	软件设计和开发	结构： 创建软件结构以满足不同的安全完整性等级中对软件安全规定要求； 复审和评价E/E/PES安全相关系统硬件结构对软件的要求，包括E/E/PES系统中软件和硬件相互作用对受控设备安全性的影响	PES； 软件系统	7.4.3	软件安全要求规范； E/E/PES硬件结构设计(见GB/T 20438.2)	软件结构设计描述； 软件结构集成测试规范； 软件/可编程电子集成测试规范(同GB/T 20438.2的要求)
		支持工具和编程语言： 在用于辅助验证、确认、评价和修改的软件的整个生命周期中，根据要求的安全完整性等级选择合适的工具集，包括语言和编译器	PES； 软件系统； 支持工具； 编程语言	7.4.4	软件安全要求规范； 软件结构设计描述	开发工具和编码标准； 开发工具的选择

表 1(续)

安全生命周期阶段		目的	范围	要求所在的条款	输入(要求的信息)	输出(产生的信息)
图 3 中的方框号	标题					
9.3	软件设计和开发	详细设计和开发(软件系统设计): 设计和实现软件,以满足不同的安全完整性等级对软件安全的规定要求,这种软件可分析、验证并能被安全地修改	软件结构设计的主要组件和子系统	7.4.5	软件结构设计描述; 支持工具和编码标准	软件系统设计描述; 软件系统集成测试规范
		详细设计和开发(单个软件模块设计): 设计和实现软件,以满足不同的安全完整性等级对软件安全的规定要求,这种软件可分析、验证并能被安全地修改	软件系统设计	7.4.5	软件系统设计规范; 支持工具和编码标准	软件模块设计规范; 软件模块测试规范
		详细代码实现: 设计和实现软件,以满足不同的安全完整性等级对软件安全的规定要求,这种软件可分析、验证并能被安全地修改	单个软件模块	7.4.6	软件模块设计规范; 支持工具和编码标准	源代码清单; 代码复审报告
		软件模块测试: 验证已满足软件安全要求(根据规定的软件安全功能和软件安全完整性),说明每一软件模块实现其预定的功能,不实现非预定的功能	软件模块	7.4.7	软件模块测试规范; 源代码清单; 代码复审报告	软件模块测试结果; 验证和测试软件模块
		软件集成测试: 验证已满足软件安全要求(根据规定的软件安全功能和软件安全完整性),说明所有软件模块、组件和子系统相互正确作用来实现其预定的功能,不实现非预定的功能	软件结构; 软件系统	7.4.8	软件系统集成测试规范	软件系统集成测试结果; 验证和测试软件系统;
9.4	PE 集成(硬件和软件)	在目标可编程电子硬件上集成软件: 将软件和硬件结合到与安全有关的可编程电子上以保证其兼容性和满足预定安全完整性等级的要求	可编程电子硬件; 集成软件	7.5.2	软件结构集成测试规范; 可编程电子集成测试规范(同 GB/T 20438.2 要求); 集成可编程电子	软件结构集成测试结果; 可编程电子集成测试结果; 验证和测试集成的可编程电子

表 1(续)

安全生命周期阶段		目的	范围	要求所在的条款	输入(要求的信息)	输出(产生的信息)
图 3 中的方框号	标题					
9.5	软件操作和修改规程	提供软件有关的信息和规程以保持操作和修改阶段中 E/E/PE 安全相关系统的功能安全	同上	7.6.2	与上面所有内容相关的	软件操作和修改规程
9.6	软件安全确认	保证集成系统符合在预定安全完整性等级上对软件安全的规定要求	同上	7.7.2	软件安全确认计划	软件安全确认结果:已确认软件
—	软件修改	修正、增强或调整确认软件以保证维持所要求的软件安全完整性等级	同上	7.8.2	软件修改规程;软件修改请求	软件修改影响分析结果;软件修改日志
—	软件验证	达到所需的安全完整性等级,测试和评价给定软件安全生命周期阶段的输出,以保证当输入该阶段时提供的输出与标准的正确性和一致性	根据阶段	7.9.2	适当的验证计划(根据阶段)	适当的验证报告(根据阶段)
—	软件功能安全评估	调查和对 E/E/PE 安全相关系统所获得的功能安全做出判断	所有以上阶段	8	软件功能安全评估计划	软件功能安全评估报告

7.2 软件安全要求规范

注 1:另见表 A.1 和表 B.7。

注 2:这一阶段是图 3 中的方框 9.1。

7.2.1 目的

7.2.1.1 根据软件安全功能要求和软件安全完整性要求规定软件安全要求。

7.2.1.2 对每个需实现一定安全功能的 E/E/PES 安全相关系统规定软件安全功能的要求。

7.2.1.3 规定每一个 E/E/PES 安全相关系统对于软件集成的要求,以保证获得这一 E/E/PES 系统分配的每一安全功能需达到的安全完整性等级。

7.2.2 要求

注:这些要求大多情况下可由通用嵌入软件和特殊应用软件共同满足。要求两者结合来提供满足下列条款的特性。两者之间的精确划分依据所选择的软件结构(见 7.4.3 和图 6)。

7.2.2.1 如果软件安全的要求已在 E/E/PE 安全相关系统的要求中规定(见 GB/T 20438.2—2006 的 7.2),则此处不必重复。

7.2.2.2 软件安全要求的规定应由 E/E/PE 安全相关系统规定的安全要求和任一安全计划编制的要求(见第 6 章)中得出(见 GB/T 20438.2),软件开发人员应能获取这些信息。

注:这一要求并不意味着 E/E/PES 开发人员和软件开发人员之间没有重复(GB/T 20438.2 和 GB/T 20438.3),当软件安全要求和软件结构(见 7.4.3)变得更加精确时,将会对 E/E/PES 硬件结构产生影响,因此软件和硬件开发人员之间的密切合作就变得非常必要了,见图 4。

7.2.2.3 软件安全要求的规定应足够细致以使设计和实现能获得要求的安全完整性,并允许执行功能安全的评估。

注:规范的细致程度可根据应用的复杂程度确定。

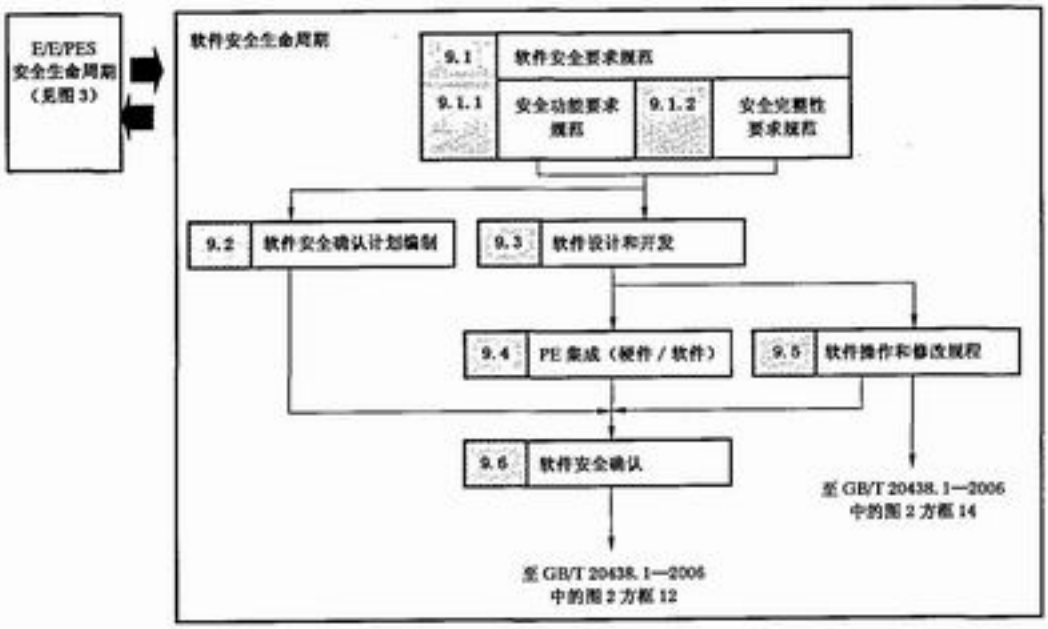


图 3　软件安全生命周期(实现阶段)

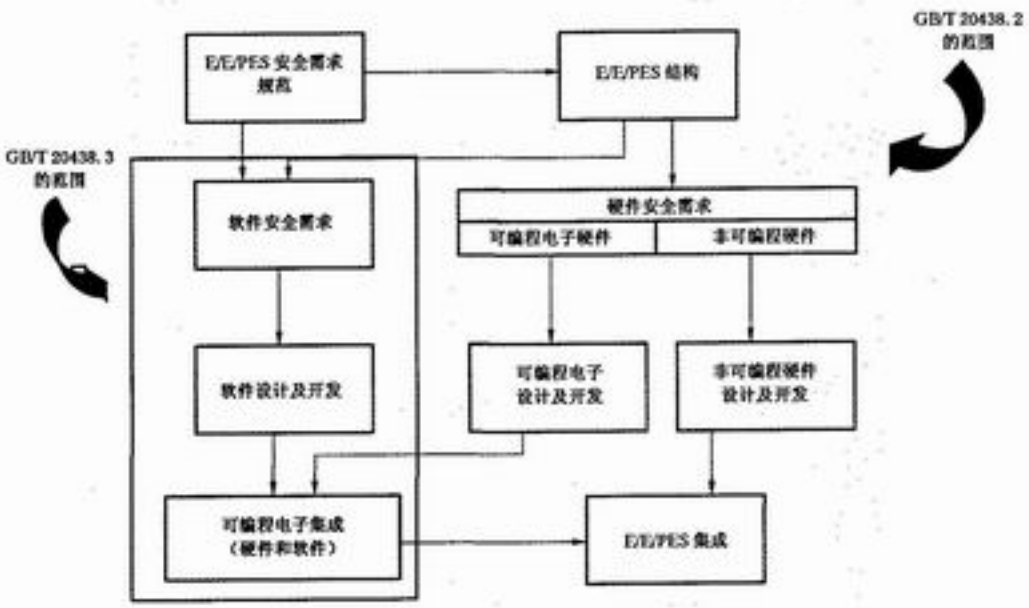


图 4　GB/T 20438.2 和 GB/T 20438.3 的范围及关系

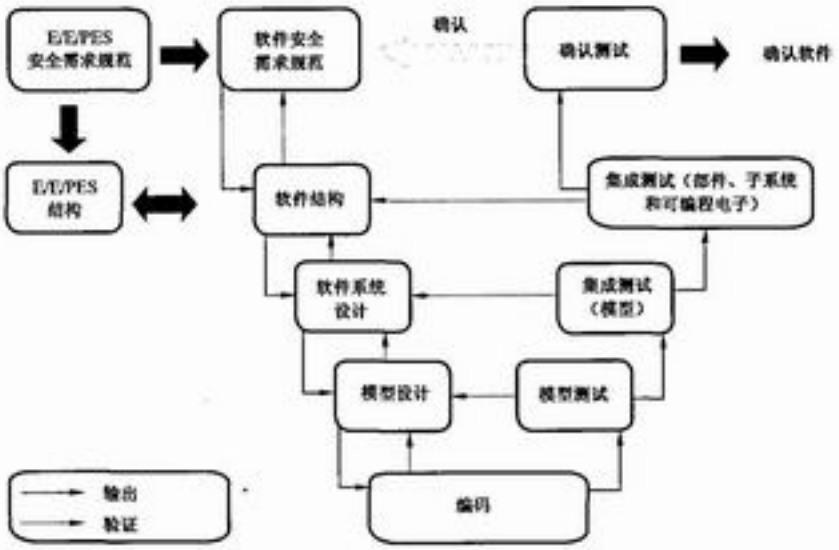
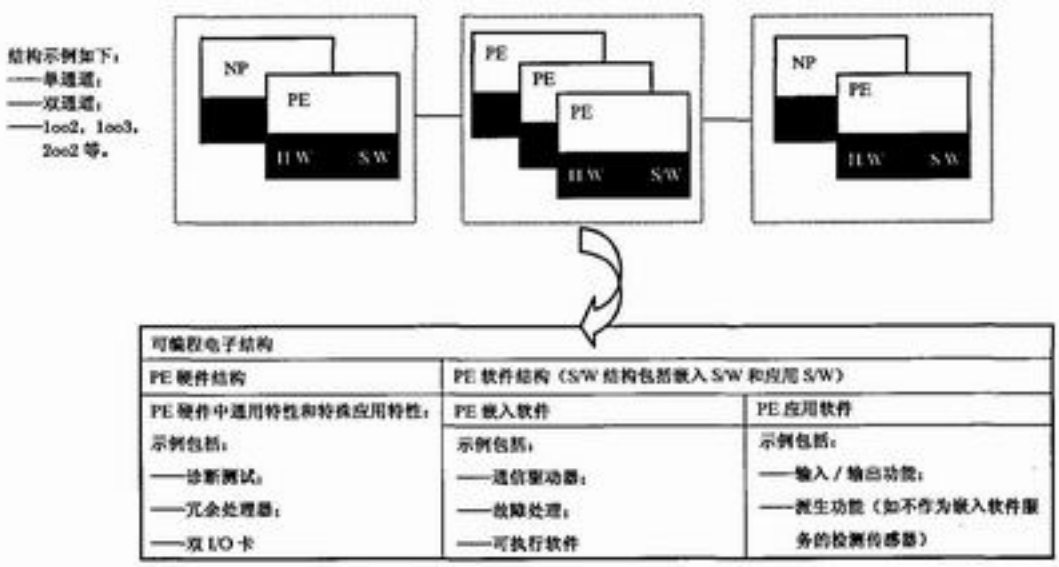


图 5　软件安全完整性的开发生命周期(V 模式)



可编程电子结构		
PE 硬件结构	PE 软件结构（S/W 结构包括嵌入 S/W 和应用 S/W）	
PE 硬件中通用特性和特殊应用特性： 示例包括： ——诊断测试； ——冗余处理器； ——双 I/O 卡	PE 嵌入软件 示例包括： ——通信驱动器； ——故障处理； ——可执行软件	PE 应用软件 示例包括： ——输入/输出功能； ——派生功能（如不作为嵌入软件服务的检测传感器）

关键词：PE——可编程电子；NP——非可编程装置；
H/W——硬件；S/W——软件；
MooN——N 中的 M(如 1oo2 为 2 中的 1)

图 6　可编程电子硬件和软件结构的关系

7.2.2.4 软件的开发人员应复审 7.2.2.2 中的信息以确保对要求全面规定，应特别考虑以下环节：

a) 安全功能；

b) 系统配置或结构；

c) 硬件安全完整性要求(可编程电子、传感器和执行器)；

d) 软件安全完整性要求；

e) 能力和响应时间性能；

f) 设备和操作人员界面。

7.2.2.5 软件开发人员应建立一个规程，以解决任何软件安全完整性等级分配的矛盾。

7.2.2.6 在要求的安全完整性等级范围内，软件安全的规定要求应得到表达和组织，以使其：

a) 清楚、准确、不含糊、可验证、可测量、可维护、可行，并与安全完整性等级相当；

b) 可回溯到 E/E/PE 安全相关系统的安全要求的规定；

c) 不使用不明确的或在软件安全生命周期任一阶段使用这些文档的人所不能理解的术语和描述。

7.2.2.7 如果没有详细定义 E/E/PE 安全相关系统的特殊安全要求，所有 EUC 的有关操作模式应在软件安全的特殊要求中详细说明。

注：这种要求通常可通过通用的嵌入软件和特殊的应用软件来获得。两者的结合要求提供满足要求的特性，通用软件和应用软件之间的精确区分依赖于软件结构的选择(见 7.4.3 和图 6)。

7.2.2.8 软件安全要求规范应对软件和硬件间的任何与安全有关的或相应的约束进行规范并文档化。

7.2.2.9 在 E/E/PE 硬件结构设计描述的范围内，软件安全规范应考虑如下内容：

a) 软件自监视(如见 GB/T 20438.7—2006 中的 C.2.5 和 C.3.10 的示例)；

b) 可编程电子硬件、传感器和执行器的监视；

c) 在系统运行时对安全功能进行的阶段性测试；

d) 当 EUC 可操作时，能够对安全功能进行的测试。

7.2.2.10 当要求 E/E/PE 安全相关系统执行非安全功能时，软件安全性的规定要求将清楚鉴别这些功能。

7.2.2.11 软件安全要求规范将表示出产品要求的安全属性，但不是工程项目的安全属性。参考 7.2.2.1～7.2.2.10，应规定以下内容：

a) 软件安全功能的要求：

——使 EUC 获得或维持安全状态的功能；

——与可编程电子硬件中故障的探测、通告和管理有关的功能；

——与传感器和执行器故障的探测、通告和管理有关的功能；

——与软件自身(软件自监视)中的故障的探测、通告和管理有关的功能；

——与在线安全功能阶段性检查有关的功能(软件自监视)；

——与离线安全功能阶段性检查有关的功能；

——允许 PES 安全修改的功能；

——非安全功能界面；

——能力和反应性能；

——软件与 PES 之间的界面。

注 1：界面包括在线和离线。

b) 软件安全完整性要求包括：

——以上 a)中每一功能的安全完整性等级。

注 2：在软件组件中分配安全完整性信息见 GB/T 20438.5—2006 中的附录 A。

7.3 软件安全确认计划编制

注：这一阶段对应图 3 中的方框 9.2。

7.3.1 目的

拟定软件安全确认计划编制。

7.3.2 要求

7.3.2.1 应执行计划编制来规定规程上和技术上步骤，用以证明软件满足其安全要求(见 7.2)。

7.3.2.2 确认软件安全计划应考虑：

a) 确认时的细节问题。

b) 执行确认的人员的细节问题。

c) EUC 操作的有关模式的识别包括：

——使用的准备，包括设置和调整；

——启动、教学、自动化、手动、半自动化、操作的稳定状态；

——重置、关机、维护；

——合理的可预见异常条件。

d) 在开始试运行前，需要确认 EUC 操作的每一模式安全软件的识别。

e) 确认的技术战略(如分析方法、统计测试等)(见 7.3.2.3)。

f) 根据 e)，用于确定符合软件安全功能(见 7.2)规定要求和软件安全完整性(见 7.2)规定要求的每一安全功能的措施(技术)和规程。

g) 软件安全规定要求的特殊参考(见 7.2)。

h) 进行确认活动时所需的环境(如测试将包括调校工具和设备)。

i) 通过/失败准则(见 7.3.2.5)。

j) 评价确认结果，特别是评价失效的方针和规程。

注：这些要求基于 GB/T 20438.1—2006 中 7.8 的一般要求。

7.3.2.3 确认安全软件的技术战略应包括下列信息(见表 A.7)：

a) 手动或自动技术选一或选二；

b) 动态或静态技术选一或选二；

c) 分析或统计技术选一或选二。

7.3.2.4 作为确认安全软件规程的组成部分，确认软件安全性的计划的范围和内容应根据安全完整性等级的要求由评估方或代表评估方的一方进行复审(见 GB/T 20438.1—2006 中的 8.2.12)，这一规程应就在测试中评估方的出席做出说明。

7.3.2.5 完成软件确认的通过/失败准则应包括：

a) 要求的输入信号及其次序和值；

b) 预期的输出信号及其次序和值；

c) 其他可接受的准则，如内存使用、定时、值的允许偏差。

7.4 软件设计和开发

注：这一阶段为图 3 中的方框 9.3。

7.4.1 目的

7.4.1.1 创建软件结构以满足不同的安全完整性等级中对软件安全的规定要求。

7.4.1.2 复审和评价 E/E/PES 安全相关系统硬件结构对软件的要求，包括 E/E/PES 系统中软件和硬件相互作用对受控设备安全性的影响。

7.4.1.3 用于辅助验证、确认、评价和修改的软件的整个的生命周期中，根据要求的安全完整性等级选择合适的工具集，包括语言和编译器。

7.4.1.4 设计和实现软件，以满足不同的安全完整性等级对软件安全的规定要求，这种软件可分析、可验证并能被安全地修改。

7.4.1.5 验证已满足软件安全要求(根据规定的软件安全功能和软件安全完整性)。

7.4.2 一般要求

7.4.2.1 根据软件开发的固有特性，7.4 中符合性责任可单独取决于供方，或单独取决于用户，或取决于两者。责任的划分应在安全计划编制过程中确定(见第 6 章)。

7.4.2.2 根据要求的软件完整性等级，设计方法的选择应具有提供以下便利的特性：

a) 抽象化、模块化和其他控制复杂性的特性。

b) 以下表达式：

——功能性；

——组件间的信息流；

——与信息有关的次序和时间；

——定时约束；

——并发性；

——数据结构及其属性；

——设计假设及其依据。

c) 开发者和其他需要懂得设计的人员的理解。

d) 验证和确认。

注：另见附录 A 和附录 B 中的表。

7.4.2.3 在设计活动中应考虑安全修改的可测试性和能力，以便在最后的安全相关系统中方便地实现这些属性。

注：例子包括机械和过程设备的维护模式。

7.4.2.4 设计方法的选择应具有方便软件修改的特性，这些特性包括模块化、信息隐蔽和封装。

7.4.2.5 设计表达方式应依据清楚定义或限制于清楚定义特征的符号表示法。

7.4.2.6 可行的设计应将软件中的安全部分尽量最小化。

7.4.2.7 当软件实现安全功能和非安全功能，所有的软件都将被认为是与安全有关的，除非在设计中表明功能之间的充分独立性。

7.4.2.8 当软件执行不同安全完整性等级的安全功能时，所有的软件都被认为是属于最高安全完整性等级，除非在设计中表明不同安全完整性等级的安全功能之间的充分独立性。

注：软件安全完整性等级至少与所属安全功能的安全完整性等级一致。但是如软件组件用于与其他硬件组件结合，其结合的安全完整性等级至少与安全功能一致时，软件组件的安全完整性等级可低于软件组件所属安全功能的安全完整性等级。

7.4.2.9 只要可行，设计应包括执行验证测试和所有诊断测试的软件功能以满足 E/E/PE 安全相关系统的安全完整性要求(见 GB/T 20438.2)。

7.4.2.10 软件设计应包括与要求的安全完整性等级保持一致的、针对控制流和数据流的自监视。对于失效探测，应采用适当的动作。见表 A.2 和表 A.4。

7.4.2.11 如果标准或以前开发的软件用作设计的一部分(见表 A.3 和表 A.4)，应能清楚识别。并应判断软件在满足软件安全要求规范方面的适宜性(见 7.2)。适宜性应以在相似的应用操作满意的证据为基础，或已经过与任何新开发的软件相同的验证和确认规程。应对以前软件环境的约束(例如操作系统和编译器依据)进行评价。

注：可在安全计划编制期间拟定判断(见第 6 章)。

7.4.2.12 只要恰当，本条也可应用于包括任何数据生成语言的数据(见 7.4)。

7.4.3 软件结构的要求

注 1：另见表 A.2 和表 B.7。

注 2：软件结构定义软件主要组件和子系统，包括它们如何实现内部连接，如何获得所要求的属性，特别是安全完整性。主要软件组件包括操作系统、数据库、大型设备输入/输出子系统、通信子系统、应用程序、编程和诊断工具等。

注3：在某些工业领域中，软件结构可称作功能描述或功能设计规范(尽管这些文档也可包括硬件)。

注4：对于使用特别像PLC这类有限可变语言的用户应用程序编程(见GB/T 20438.6—2006的附录E)，结构将由供方作为PLC的一种标准特性提供。在这一标准下将要求供方确保用户产品的符合性满足7.4的要求。用户根据应用使用标准编程工具来定制PLC，例如梯形图。7.4.3～7.4.8的要求仍适用。结构的定义和文档要求可作为信息被用户用来选择适用的PLC(或相关物)。

注5：在另一极端情况下，在使用完全可变语言的某些嵌入应用中，例如一个控制机器的微处理器，特别需要供方根据应用(或应用的种类)来建立结构。通常用户没有编程能力。在这些情况下，确保7.4的符合性的责任取决于供方。

注6：还有一些系统介于注4和注5提到的两类系统之间，确保符合性的责任由供方和用户共同承担。

注7：从安全角度讲，在软件结构阶段开发软件基本安全策略。

7.4.3.1 根据软件开发的固有特性，确保7.4.3的符合性要求的责任由供方或用户单独承担，或由两者共同承担(见上注)，责任的划分应在安全计划编制中文档化(见第6章)。

7.4.3.2 计划的软件结构设计将由软件供方和/或开发人员来建立，软件结构设计的描述应详细，描述将：

a) 在所需的软件安全生命周期中，在要求的安全完整性等级上，选择和判断满足软件安全性要求规范的集成的技术和措施集(见7.2)。这些技术和措施包括故障允许偏差(与硬件一致)和故障避免的软件设计策略，包括(适用时)冗余和多样性。

b) 根据组件/子系统的划分，对每一部分应提供以下信息：
——它们是否是新的、已存在的或专利的；
——它们是否已被验证、如果是，它们的验证条件；
——每一个组件/子系统是否安全有关；
——组件/子系统的软件安全完整性等级。

c) 确定所有软件/硬件相互作用和评价及细化它们的重要性。

d) 使用符号表示法表示清楚定义的或限制清楚定义特性的结构。

e) 选择用于保持所有数据安全完整性的设计特征。这种数据可包括大型设备输入/输出数据、通信数据、操作界面数据、维护数据和内部数据库数据。

f) 规定适当的软件结构集成测试来保证软件结构满足规定安全完整性等级上的软件安全要求(见7.2)。

7.4.3.3 应用7.4.3.2后，E/E/PE安全相关系统中规定安全要求的任何变化都应经E/E/PE开发人员同意并文档化。

注：软件和硬件结构不可避免会有重叠(见图5)，因此需要与硬件开发人员讨论可编程电子硬件和软件集成(见7.5)的测试规范等类问题。

7.4.4 支持工具和编程语言的要求

注1：另见表A.3。

注2：开发工具的选择将依据软件开发活动和软件结构的固有特性(见7.4.3)。

——对于使用有限可变语言的用户应用程序编程，在一个低安全完整性等级下，要求的工具和编程语言可被限定为标准PLC语言、编辑器、加载器。7.4.4符合性的责任主要由供方承担。

——在较高的安全完整性等级上，需限制PLC语言的子集，验证和确认工具如代码分析器和仿真器等。在这些环境下责任由供方和用户共同承担。

——即便是在低等级的安全完整性下，也应广泛使用完全可变性语言的嵌入应用工具。7.4.4符合性的责任主要由软件开发人员来承担。这包括使用完全可变语言为用户应用程序编程提供低可变语言的PLC供方。

7.4.4.1 根据软件开发的固有特性，确保7.4.4的符合性要求的责任由供方或用户单独承担，或由两者共同承担(见上注2)，责任的划分应在安全计划编制中文档化(见第6章)。

7.4.4.2 一套合适的集成工具，包括语言、编译器、配置管理工具、应用时的自动测试工具，应根据要求的安全完整性等级选择。应考虑在 E/E/PE 安全相关系统整个生命周期中提供相应服务的合适的开发工具(不是那些在系统开发的初始阶段期间使用的)的可用性。

7.4.4.3 在安全完整性等级要求的范围内，程序编程语言选择应：

a) 具备有国家标准或国际标准认可的确认证书的翻译器/编译器，或对其目的的适宜性建立评估；

b) 完全并清楚地定义或限制清楚定义特性；

c) 与应用的特征匹配；

d) 包括方便探测程序错误的特性；

e) 支持与设计方法匹配的特性。

7.4.4.4 当不能满足 7.4.4.3 时，软件结构设计描述中应记录另一种可选择语言的理由(见 7.4.3)。理由应足够详细说明语言目的的适宜性和任何说明语言缺点的附加措施。

7.4.4.5 编码标准应：

a) 由评估方复审其与使用目的是否适合；

b) 用于开发所有安全软件。

7.4.4.6 编码标准应规定好的编程习惯，描述非安全语言特性(如未定义的语言特性、非结构化设计等)和规定源代码文档规程。源代码文档中至少应包括下列信息：

a) 法律实体(如公司、作者等)；

b) 描述；

c) 输入和输出；

d) 配置管理历史。

7.4.5 详细设计和开发要求

注 1：另见表 A.4、表 B.1、表 B.7 和表 B.9。

注 2：此处定义的详细设计指软件系统设计——结构中的主要组件划分在软件模块、单独的软件模块设计和编码系统中。在小型应用中，软件系统设计和结构设计可结合起来。

注 3：详细设计和开发的固有特性根据软件开发行为和软件结构的固有特性而变化(见 7.4.3)。对于使用有限可变语言的用户应用程序编程，如梯形图和功能块，详细设计可看作是配置而不是编程。但是，它仍是一种以结构化方法设计软件的很好的习惯，包括将软件内置到分开(尽可能)安全部件的模块结构中；包括提供预防数据输入错误的范围检测和其他特性；使用以往确认过的软件模块和提供方便未来软件修改的设计。

7.4.5.1 根据软件开发的固有特性，确保 7.4.5 的符合性要求的责任由供方或用户单独承担，或由两者共同承担(见上注 3)，责任的划分应在安全计划编制中文档化(见第 6 章)。

7.4.5.2 下列信息应在详细设计开始前获得：软件安全要求规范(见 7.2)；软件结构设计的描述(见 7.4.3)；软件安全性的确认计划(见 7.3)。

7.4.5.3 软件的生产应具有模块化、可测试性、安全修改能力。

7.4.5.4 对于软件结构设计(见 7.4.3)描述中的每一个主要组件/子系统，设计的进一步优化应根据软件模块的划分(即软件系统设计的规范)。每一软件模块的设计和测试应适用于规定的每一软件模块。

注：对于标准的或以往开发的软件组件或软件模块，如果表明它们满足 7.4.2.11 的规定则不需要进行设计或测试规定。

7.4.5.5 规定适当的软件系统集成测试以保证软件系统满足在要求安全完整性等级上的软件安全规定要求(见 7.2)。

7.4.6 代码实现要求

注：另见表 A.4、表 B.1 和表 B.9。

7.4.6.1 源代码应：

a) 可读、可理解和可测试；

b) 满足软件模块设计的规定要求(见 7.4.5)；

c) 满足编码标准的规定要求(见 7.4.4)；

d) 满足安全计划编制中规定的所有相关要求(见第 6 章)。

7.4.6.2 每一软件代码模块应复审。

注：代码复审是一种验证活动(见 7.9)。

7.4.7 软件模块测试要求

注 1：另见表 A.5、表 B.2、表 B.3 和表 B.6。

注 2：测试软件模块正确满足测试规范是一种验证活动(见 7.9)，是代码复审和软件模块测试的结合，用以证明软件模块满足它的相关规范，即已验证。

7.4.7.1 每一软件模块在软件设计中都应根据规定进行测试。

7.4.7.2 这些测试表明每一软件模块执行其预定功能且不执行其非预定功能。

注 1：这不意味着测试所有输入组合和输出组合。所有相关种类测试(见 GB/T 20438.7—2006 中 C.5.7)或结构测试(GB/T 20438.7—2006 中 C.5.8)已足够。边界值分析(见 GB/T 20438.7—2006 中 C.5.7)、控制流分析(见 GB/T 20438.7—2006 中 C.5.9)或寄生回路分析(见 GB/T 20438.7—2006 中 C.5.11)可将测试用例减少至一个可接受的数量。可分析程序(见 GB/T 20438.7—2006 中 C.5.7)能方便地满足要求。

注 2：当使用形式方法(见 GB/T 20438.7—2006 中 C.5.7)，形式证明(见 GB/T 20438.7—2006 中 C.5.13)或断言(见 GB/T 20438.7—2006 中 C.5.7)开发时，此类措施可在一定范围内减少。

注 3：同时可以使用统计证据(见 GB/T 20438.7—2006 中的附录 D)。

7.4.7.3 软件模块测试的结果应文档化。

7.4.7.4 应规定测试失效的校正动作规程。

7.4.8 软件集成测试的要求

注 1：另见表 A.5、表 B.2、表 B.3 和表 B.6。

注 2：测试软件正确集成是一种验证活动(见 7.9)。

7.4.8.1 软件集成测试应在设计和开发阶段正确规定。

7.4.8.2 软件集成测试应规定以下内容：

a) 管理集成集中的软件划分；

b) 测试用例和测试数据；

c) 执行测试的种类；

d) 测试环境、工具、配置和程序；

e) 测试完成的准则应判断；并且

f) 测试失效校正动作的规程。

7.4.8.3 软件应根据规定的软件集成测试要求进行测试。这些测试应表明所有软件模块和软件组件/子系统内部正确作用以执行其预定的功能而不执行非预定的功能。

注 1：这不意味着测试所有输入组合和输出组合。所有相关种类测试(见 GB/T 20438.7—2006 中 C.5.7)或结构测试(GB/T 20438.7—2006 中 C.5.8)已足够。边界值分析(见 GB/T 20438.7—2006 中 C.5.7)、控制流分析(见 GB/T 20438.7—2006 中 C.5.9)或寄生回路分析(见 GB/T 20438.7—2006 中 C.5.11)可将测试用例减少至一个可接受的数量。可分析程序(见 GB/T 20438.7—2006 中 C.5.7)能方便地满足要求。

注 2：当使用形式方法(见 GB/T 20438.7—2006 中 C.5.7)，形式证明(见 GB/T 20438.7—2006 中 C.5.13)或断言(见 GB/T 20438.7—2006 中 C.5.7)开发时，此类措施可在一定范围内减少。

注 3：同种可以使用统计证据(见 GB/T 20438.7—2006 中的附录 D)。

7.4.8.4 软件集成测试的结果应文档化，说明测试结果是否满足目的和测试准则。如果出现失效，应记录失效原因。

7.4.8.5 在软件集成过程中，应对软件的任何修改或改变进行影响分析以确定对所有软件模块的影响和所需要的再验证和再设计活动。

7.5 可编程电子集成(硬件和软件)

注1:另见表A.6、表B.3和表B.6。

注2:这一阶段是图3中的方框9.4。

7.5.1 目的

7.5.1.1 在目标可编程电子硬件上集成软件。

7.5.1.2 将软件和硬件结合到与安全有关的可编程电子上以保证其兼容性和满足预定安全完整性等级的要求。

注1:测试软件是否正确集成到可编程电子硬件中是一种验证活动(见7.9)。

注2:根据固有特性这些活动可与7.4.8结合。

7.5.2 要求

7.5.2.1 应在设计和开发阶段中规定集成测试,以保证在安全有关可编程电子中硬件和软件的兼容性。

注:可能会要求与E/E/PE开发人员紧密合作以开发集成测试。

7.5.2.2 可编程电子(硬件和软件)的集成测试应规定:

a) 根据集成水平拆分系统;

b) 测试用例和测试数据;

c) 执行测试的种类;

d) 测试环境、工具、配置和程序;

e) 测试完成的准则应判断。

7.5.2.3 在进行可编程电子(硬件和软件)规定的集成测试时,应区别开发人员按自己的意图所执行的活动和从用户立场出发所进行的活动。

7.5.2.4 对可编程电子(硬件和软件)规定的集成测试应在下列行为中进行区分:

a) 将软件系统纳入目标可编程电子硬件;

b) E/E/PE集成,即增加接口如传感器和执行器;

c) EUC和E/E/PE安全相关系统的全部集成。

注:b)和c)已由GB/T 20438.1和GB/T 20438.2包含,此处为保证完整性将a)包括进来。

7.5.2.5 软件应根据可编程电子(硬件和软件)规定的集成测试和与安全有关的可编程电子硬件进行集成。

7.5.2.6 在安全有关可编程电子(硬件和软件)集成测试中,应对集成系统的任何修改或改变进行影响分析,以确定对所有软件模块的影响和所需要的再验证活动。

7.5.2.7 测试用例及其结果应记录用于随后的分析。

7.5.2.8 安全有关可编程电子(硬件和软件)的集成测试应文档化,说明测试结果是否满足测试目的和测试准则。如果出现失效,应记录失效原因。软件的任何修改或改变应进行影响分析以确定对所有软件组件/模型的影响和所需要的再验证和再设计活动。

7.6 软件操作和修改程序

注1:另见表A.8。

注2:该阶段是图3中方框9.5。

7.6.1 目的

提供软件有关的信息和规程以保持操作和修改阶段中E/E/PE安全相关系统的功能安全。

7.6.2 要求

本部分中7.8和GB/T 20438.2—2006中7.6给出的要求。

注:在GB/T 20438中,与其对软件(不同于硬件)进行维护,不如对其进行修改。

7.7 软件安全确认

注1:另见表A.7、表B.3和表B.5。

注2:这一阶段为图3中方框9.6。

7.7.1 目的

保证集成系统符合在预定安全完整性等级上对软件安全的规定要求(见7.2)。

7.7.2 要求

7.7.2.1 如果作为E/E/PE安全相关系统一部分的软件安全要求的一致性已被建立(见GB/T 20438.2—2006中7.7),则不必重复确认。

7.7.2.2 确认活动应根据软件安全确认计划编制中的规定进行。

7.7.2.3 软件安全确认的结果应文档化。

7.7.2.4 对每一个安全功能,软件安全确认应对以下结果文档化:

a) 确认活动的按时间顺序的记录;

b) 所用软件安全确认计划(见7.3)的版本;

c) 确认的安全功能(通过测试或分析)、连同软件安全确认计划(见7.3)的参考;

d) 使用带校正数据的工具和设备;

e) 确认活动的结果;

f) 预期和实际结果的差异。

7.7.2.5 当预期和实际结果出现差异时,对是否继续确认做出的分析和采取的决定,或发布变更请求和返回开发生命周期较早阶段,都应作为软件安全确认的结果的一部分文档化。

注:7.7.2.2~7.7.2.5的要求是根据GB/T 20438.1—2006中的7.14一般要求。

7.7.2.6 安全软件的确认应符合下列要求:

a) 测试应该是软件确认的主要方法;直观显示和建模可用作确认活动的补充。

b) 软件应通过仿真实验:

——正常操作中出现的输入信号;

——预期的事件;

——系统动作不期望的条件。

c) 供方和/或开发人员应使软件安全确认的文档化结果及其所有的附属文档对系统开发者及供方而言能使他们自己可用,以满足GB/T 20438.1和GB/T 20438.2的要求。

7.7.2.7 软件工具鉴定要求如下:

a) 所有用于确认的设备符合应可回溯到某一国际标准(如果可能)、国家标准(如果可能)、一个已被广泛认可的规程的要求;

b) 用于软件确认的设备应进行适当鉴定,使用的任何工具、硬件和软件应与目的相适应。

注:在GB/T 20438中,鉴定是说明满足特殊规定的活动,而不是适用于任何规定的通用一致性测试规程。

7.7.2.8 软件确认结果要求如下:

a) 测试将表明正确执行所有软件安全的规定要求并且软件系统不执行非预定的功能;

b) 测试用例和其结果应文档化以用于后续的分析,并且要求有安全完整性等级要求的独立的评估(见GB/T 20438.1—2006中的8.2.12);

c) 文档化的软件安全确认的结果将表明软件是否通过确认或失效的原因。

7.8 软件修改

注1:另见表A.8。

注2:这一阶段是图3中的方框9.5。

7.8.1 目的

修正、增强或调整确认软件以保证维持所要求的软件安全完整性等级。

注：在 GB/T 20438 中，与其对软件(不同于硬件)进行维护，不如对其进行修改。

7.8.2 要求

7.8.2.1 在执行任何软件修改之前，软件修改规程已可用(见 GB/T 20438.1—2006 中的 7.16)。

注 1：7.8.2.1～7.8.2.9 主要用于在软件操作阶段中改变。它们也可用于可编程电子集成和全部安装和试运行阶段(见 GB/T 20438.1—2006 中 7.13)。

注 2：一个修改程序模型的例子见 GB/T 20438.1—2006 中的图 9。

7.8.2.2 只有经批准的软件修改请求在安全计划编制阶段规定的规程(见第 6 章)下，发出并包括以下细节后，才可启动修改：

a) 可能被影响的危险；

b) 预期的改变；

c) 改变的原因。

注：修改请求的原因如下：

——功能安全低于规定要求；

——系统故障经验；

——新的或新修订的安全法规；

——EUC 或其使用的修改；

——全部安全要求的修改；

——操作和维护性能分析，表明性能低于目标值；

——例行功能安全审核。

7.8.2.3 应进行预定的软件修改对 E/E/PE 安全相关系统的功能安全性的影响的分析，以：

a) 确定是否需要危险和风险分析；

b) 确定哪个软件安全生命周期阶段需重复。

7.8.2.4 7.8.2.3 中影响分析结果应文档化。

7.8.2.5 所有对 E/E/PE 安全相关系统的功能安全有影响的修改应返回软件安全生命周期一个适当阶段。所有后续的阶段也应根据 GB/T 20438 对特殊阶段的规定规程执行。安全计划编制(见第 6 章)应细化所有后续的活动。

注：可能需要实现一个全面的危险和风险分析，对安全相关系统的安全完整性等级和外部风险降低设施所规定的某一个不同的安全完整性等级可能会有这样一个需求。

7.8.2.6 安全软件修改的安全计划编制应包括以下信息：

a) 人员识别和及其所需能力的规定；

b) 修改的详细规定；

c) 验证计划编制；

d) 在安全完整性等级要求的范围内，修改的再确认和测试范围。

7.8.2.7 修改应按照计划执行。

7.8.2.8 所有修改的细节应被文档化，包括：

a) 修改/改型请求；

b) 评估预定软件修改对功能安全影响的分析结果和有关判断的决定；

c) 软件配置管理历史；

d) 与正常操作和条件的偏差；

e) 受修改活动影响的所有文档化信息；

7.8.2.9 所有修改的详细信息(如日志)应文档化。文档应包括数据和结果的再验证和再确认。

注：7.8.2.1～7.8.2.9 主要用于在软件操作阶段中改变。它们也可用于可编程电子集成和全部安装和试运行阶段(见 GB/T 20438.1—2006 中 7.13)。

7.8.2.10 要求的修改或改型活动的评估将根据影响分析的结果和软件安全完整性等级。

7.9 软件验证

注：另见表 A.9、表 B.2 和表 B.8。

7.9.1 目的

达到所需的安全完整性等级，测试和评价给定软件安全生命周期阶段的输出，以保证当输入该阶段时提供的输出与标准的正确性和一致性。

注 1：本条考虑安全生命周期几个阶段通用的验证内容。本条不对 7.4.7(软件模块测试)、7.4.8(软件集成)和 7.5(可编程电子集成)中验证活动发生在内部的验证测试元素提出额外要求，也不对在 GB/T 20438 中作为规定安全要求的符合性证明的软件确认(见 7.7)提出要求(end—end 验证)，由本领域的专家来执行对安全要求规定是否正确的自检查。

注 2：根据软件结构，验证活动的责任可由开发和修改软件的所有有关机构共同承担。

7.9.2 要求

7.9.2.1 软件验证应与开发做好同步计划，对软件安全生命周期的每一个阶段，这个信息应文档化。

7.9.2.2 软件验证计划编制应参考确认活动中使用的准则、技术和工具，并应注明：

a) 安全完整性要求的评价；

b) 验证战略、活动和技术的选择和文档；

c) 验证工具的选择和使用(测试工具、专业测试软件、输入/输出仿真器等)；

d) 验证结果的评价；

e) 采用的校正动作 。

7.9.2.3 软件验证应根据计划执行。

注：验证技术和措施的选择以及验证活动的独立程度，取决于很多因素并可能在应用领域的标准中规定。

因素的例子如下：

——工程规模；

——复杂程度；

——设计的新颖程度；

——技术的新颖程度。

7.9.2.4 表明被验证的阶段已在所有方面圆满完成的证据应文档化。

7.9.2.5 每次验证后，验证文档应包括：

a) 被验证的项目识别；

b) 对应验证完成的信息识别；

c) 非符合性。

注：非符合性的例子包括软件模块、数据结构和不常采用的算法。

7.9.2.6 软件安全生命周期 N 阶段中所有 $N+1$ 阶段正确执行所需的信息都应可获得并被验证，N 阶段的输出包括：

a) N 阶段的规范、设计描述或代码应充分满足：

——功能性；

——安全完整性、性能和其他安全计划编制的要求(见第 6 章)；

——开发小组可读；

——进一步验证的可测试性；

——允许进一步改进的安全修改。

b) 对规定和描述 N 阶段的设计

N 阶段规定的确认计划和/或测试是充分性的。

c) 检查下列两点之间的不兼容性：

——N 阶段规定的测试和 $N-1$ 阶段规定的测试；

——N 阶段中的输出。

7.9.2.7 根据7.1.2.1,应执行下列验证活动:

a) 软件安全要求的验证(见7.9.2.8);
b) 软件结构的验证(见7.9.2.9);
c) 软件系统设计的验证(见7.9.2.10);
d) 软件模块设计的验证(见7.9.2.11);
e) 代码的验证(见7.9.2.12);
f) 数据验证(见7.9.2.13);
g) 软件模块测试(见7.4.7);
h) 软件集成测试(见7.4.8);
i) 可编程电子集成测试(见7.5);
j) 软件安全要求措施(软件确认)(见7.7)。

7.9.2.8 软件安全要求验证:一旦规定软件安全要求(见7.2),在下一阶段、软件设计和开发开始前,验证应:

a) 考虑规定的软件安全要求(见7.2)是否已充分满足E/E/PES规定的对功能、安全完整性、性能和其他安全计划编制的要求(见GB/T 20438.2)。
b) 考虑软件安全确认计划编制(见7.3)是否已充分满足规定的软件安全要求(见7.2)。
c) 检查下列两点之间的不兼容性:
——规定的软件安全要求(见7.2)和规定的E/E/PES安全要求(见GB/T 20438.2);
——规定的软件安全要求(见7.2)和软件安全确认计划编制(见7.3)。

7.9.2.9 软件结构验证:在建立软件结构设计后,验证应:

a) 考虑软件结构设计的描述(见7.4.3)是否已充分满足规定的软件安全要求(见7.2)。
b) 考虑软件结构集成规定的测试(见7.4.3)对软件结构设计描述是否充分(见7.4.3)。
c) 考虑每一主要组件/子系统的属性是否充分满足:
——要求的安全性能的柔性;
——进一步验证的可测试性;
——开发和验证小组可读;
——允许进一步改进的安全修改。
d) 检查下列不兼容性:
——软件结构设计的描述(见7.4.3)和规定的软件安全要求(见7.2);
——软件结构设计的描述(见7.4.3)和规定的软件结构集成测试(见7.4.3);
——软件结构集成的规定测试(见7.4.3)和软件安全确认计划编制(见7.3)。

7.9.2.10 软件系统设计验证:规定软件系统设计后,验证应:

a) 考虑规定的软件系统设计(见7.4.5)是否已充分满足软件结构设计(见7.4.3)。
b) 考虑软件系统集成规定的测试(见7.4.5)是否已充分满足规定的软件系统设计(见7.4.5)。
c) 考虑规定的软件系统设计的每一主要组件的属性(见7.4.5)是否已足够满足:
——要求的安全性能的柔性;
——进一步验证的可测试性;
——开发和验证小组可读;
——允许进一步改进的安全修改。

注:软件系统集成测试可作为软件结构集成测试的一部分。

d) 检查下列三点之间的不兼容性:
——规定的软件系统设计(见7.4.5)和软件结构设计的描述(见7.4.3);
——软件系统设计描述(见7.4.5)和软件系统集成规定的测试(见7.4.5);

——软件系统集成规定的测试(见 7.4.5)和结构集成规定的测试(见 7.4.3)。

7.9.2.11 软件模块设计验证:在规定每一软件模块设计后,验证应:

a) 考虑规定的软件模块设计(见 7.4.5)是否已充分满足规定的软件系统设计(见 7.4.5)。

b) 考虑每一软件模块的规定测试(见 7.4.5)对规定的软件模块设计是否充分(见 7.4.5)。

c) 考虑每一软件模块的属性是否充分满足:

——要求的安全性能的柔性;

——进一步验证的可测试性;

——开发和验证小组可读;

——允许进一步改进的安全修改。

d) 检查下列三点之间的不兼容性:

——规定的软件模块设计(见 7.4.5)和规定的软件系统设计(见 7.4.5);

——(对每一个软件模块)规定的软件模块设计(见 7.4.5)和规定的软件模块测试(见 7.4.5);

——规定的软件模块测试(见 7.4.5)和规定的软件系统集成测试(见 7.4.5)。

7.9.2.12 代码验证:源代码需通过静态方法验证,以确保软件模块的规定设计(见 7.4.5)、要求的编码标准(见 7.4.4)和安全计划编制要求(见 7.3)之间的符合性。

注:在软件安全生命周期的早期阶段,验证是静态的(如检查、复审、形式证明等)。代码验证包括软件检查和走查等技术。它是代码验证的结果和软件模块测试的结合,以保证每一软件模块满足其相关规范。此后,向前测试成为验证的主要方法。

7.9.2.13 数据验证

a) 设计中规定的数据结构应验证:

——完整性;

——自身一致性;

——对改变或破坏的防范;

——数据驱动系统功能要求的一致性。

b) 应用数据应验证:

——与数据结构的一致性;

——完整性;

——与基础系统软件的兼容性(如执行的次序、运行时间等);

——数据值的校正。

注:应用数据的例子有数字控制设备的部分程序。系统软件(典型的是一种子例程采集程序)作为应用数据的解释器。另外,应用数据也可考虑作为一种应用程序。

c) 所有修改参数应验证以防止:

——无效或未定义初始值;

——错误、不连续或不合理值;

——非批准改变;

——数据损坏。

d) 所有大型设备接口和有关软件(即传感器、执行器和离线界面,见 7.2.2.11)应验证,以:

——用于预期界面失效的探测;

——用于预期界面失效的容错。

e) 所有通信接口和有关软件应对下列事件的充分水平进行验证:

——失效探测;

——错误防范;

——数据确认。

8 功能安全评估

8.1 GB/T 20438.1—2006 第 8 章规定的目的和要求适用于安全软件的评估。

8.2 除非在应用领域标准中另有规定，执行功能安全评估的有关方面的最低独立性水平应与 GB/T 20438.1—2006中 8.2.12 的规定一致。

8.3 功能安全评估可利用表 A.10 中活动的结果。

注：选择附录 A 和附录 B 中的技术其自身不能保证获得要求的安全完整性(见 7.1.2.6)，评估方还要考虑：

——在整个开发周期中选择的方法、语言和工具的一致性和完整性；

——开发人员是否选择了他们完全理解的方法、语言和工具；

——选择的方法、语言和工具是否是在开发阶段中对所针对的特殊问题能被普遍认可的。

附 录 A
（规范性附录）
技术和措施选择指南

GB/T 20438 中一些条有相关表格，如 7.2（软件安全要求规范）与表 A.1 有关，附录 B 中更详细的表有些是由附录 A 中的表扩展来的，如表 B.2 根据表 A.5 中动态分析和测试的主题扩展而成。

附录 A 和附录 B 中提及的技术和措施可见 GB/T 20438.7。

表中的每一技术和措施都有安全完整性等级 1～4 的推荐，推荐如下：

——HR：在该安全完整性等级下极力推荐的技术或措施。如未采用这种技术或措施则应在安全计划中详细记录未使用该技术和措施的理由并须经评估方批准。

——R：在该安全完整性等级下，低于 HR 推荐程度的所推荐的技术或措施。

———：未推荐或不能使用的技术或措施。

——NR：在该安全完整性等级下绝不推荐的技术或措施。如果采用这类技术或措施，应在安全计划中详细记录使用该技术和措施的理由并须经评估方批准。

应根据安全完整性等级选择适当的技术/措施。其他可选择的或等价技术/措施应用一个后跟数字的字母说明。只需满足一种其他可选择的或等价技术/措施。

技术和措施的分类与 GB/T 20438.2 中所使用的有效性的概念有关。对于所有其他等价因素，HR 类的技术在软件开发中防止系统故障或（软件结构用例中）在软件执行中控制残余故障方面都比 R 类技术更加有效。

给出大量影响软件安全完整性的因素并不可能给出一种将技术和措施结合起来的算法来校正任何指定的应用。但 GB/T 20438.6 中通过两个工作实例给出了使用各表的指南。

对于特殊的应用，在安全计划编制中陈述了适当的技术或措施的结合，以及选择的适当的技术或措施，除非在表中注明做出了其他要求。

GB/T 20438.6 中给出对用户应用程序表格的解释初始指南。

表 A.1 软件安全要求规范（见 7.2）

技术/措施[a]	Ref	SIL1	SIL2	SIL3	SIL4
1 计算机辅助规范工具	B.2.4	R	R	HR	HR
2a 半形式方法	表 B.7	R	R	HR	HR
2b 形式方法包括如 CCS，CSP，HOL，LOTOS，OBJ，暂存逻辑，VDM 和 Z	C.2.4	—	R	R	HR
注 1：软件安全要求规范需对问题使用自然语言描述和任何反映应用所需的数学符号。 注 2：本表清楚、准确反映了规定的软件安全要求的附加要求。					
[a] 应根据安全完整性等级选择适当的技术/措施。其他可选择的或等价技术/措施应用数字说明，只需满足一种其他可选择的或等价技术/措施。					

表 A.2 软件设计和开发:软件结构设计(见 7.4.3)

技术/措施[a]	Ref	SIL1	SIL2	SIL3	SIL4
1 故障探测和诊断	C.3.1	—	R	HR	HR
2 错误探测和校正代码	C.3.2	R	R	R	HR
3a 失效断言编程	C.3.3	R	R	R	HR
3b 安全袋技术	C.3.4	—	R	R	R
3c 备种编程	C.3.5	R	R	R	HR
3d 恢复块	C.3.6	R	R	R	R
3e 反向恢复	C.3.7	R	R	R	R
3f 正向恢复	C.3.8	R	R	R	R
3g 再试故障恢复机制	C.3.9	R	R	R	HR
3h 存储被执行用例	C.3.10	—	R	R	HR
4 适度降级	C.3.11	R	R	HR	HR
5 人工智能故障校正	C.3.12	—	NR	NR	NR
6 动态再配置	C.3.13	—	NR	NR	NR
7a 结构化方法包括如 JSD、MASCOT、SADT 和 Yourdon	C.2.1	HR	HR	HR	HR
7b 半形式方法	表 B.7	R	R	HR	HR
7c 形式方法包括如 CCS、CSP、HOL、LOTOS、OBJ、暂存逻辑、VDM 和 Z	C.2.4	—	R	R	HR
8 计算机辅助规范工具	B.2.4	R	R	HR	HR
注:表中有关故障容错(失效控制)的措施应同 GB/T 20438.2 中可编程电子硬件中结构要求的失效控制结合起来考虑。					
[a] 应根据安全完整性等级选择适当的技术/措施。其他可选择的或等价技术/措施应用数字说明,只需满足一种其他可选择的或等价技术/措施。					

表 A.3 软件设计和开发:支持工具和编程语言(见 7.4.4)

技术/措施[a]	Ref	SIL1	SIL2	SIL3	SIL4
1 适当的编程语言	C.4.6	HR	HR	HR	HR
2 强类型编程语言	C.4.1	HR	HR	HR	HR
3 语言子集	C.4.2	—	—	HR	HR
4a 已认证的工具	C.4.3	R	HR	HR	HR
4b 工具:通过使用提高置信度	C.4.4	HR	HR	HR	HR
5a 已认证的翻译器	C.4.3	R	HR	HR	HR
5b 翻译器:通过使用提高置信度	C.4.4	HR	HR	HR	HR
6 可信的/经验证的软件模块和组件库	C.4.5	R	HR	HR	HR
[a] 应根据安全完整性等级选择适当的技术/措施。其他可选择的或等价技术/措施应用数字说明,只需满足一种其他可选择的或等价技术/措施。					

表 A.4 软件设计和开发:详细设计(见 7.4.5 和 7.4.6)
(包括软件系统设计、软件模块设计和编码)

技术/措施*	Ref	SIL1	SIL2	SIL3	SIL4
1a 结构方法包括如 JSD、MASCOT、SADT 和 Yourdon	C.2.1	HR	HR	HR	HR
1b 半形式方法	表 B.7	R	HR	HR	HR
1c 形式方法包括如 CCS、CSP、HOL、LOTOS、OBJ、暂存逻辑、VDM 和 Z	C.2.4	—	R	R	HR
2 计算机辅助设计工具	B.3.5	R	R	HR	HR
3 防御性编程	C.2.5	—	R	HR	HR
4 模块化方法	表 B.9	HR	HR	HR	HR
5 设计和编码标准	表 B.1	R	HR	HR	HR
6 结构化编程	C.2.7	HR	HR	HR	HR
7 可信的/经验证的软件模块和组件库(如可行)	C.2.10 C.4.5	R	HR	HR	HR
* 应根据安全完整性等级选择适当的技术/措施。其他可选择的或等价技术/措施应用数字说明,只需满足一种其他可选择的或等价技术/措施。					

表 A.5 软件设计和开发:软件模块测试和集成(见 7.4.7 和 7.4.8)

技术/措施*	Ref	SIL1	SIL2	SIL3	SIL4
1 概率测试	C.5.1	—	R	R	HR
2 动态分析和测试	B.6.5 表 B.2	R	HR	HR	HR
3 数据记录和分析	C.5.2	HR	HR	HR	HR
4 功能和黑盒测试	B.5.1 B.5.2 表 B.3	HR	HR	HR	HR
5 性能测试	C.5.20 表 B.6	R	R	HR	HR
6 接口测试	C.5.3	R	R	HR	HR
注 1:软件模块和集成测试是验证活动(见表 A.9)。 注 2:应根据安全完整性等级选择适当的技术/措施。					
* 应根据安全完整性等级选择已编号的技术/措施。					

表 A.6 可编程电子集成(硬件和软件)(见 7.5)

技术/措施*	Ref	SIL1	SIL2	SIL3	SIL4
1 功能和黑盒测试	B.5.1 B.5.2 表 B.3	HR	HR	HR	HR
2 性能测试	C.5.20 表 B.6	R	R	HR	HR
注 1:可编程电子集成是一种验证活动(见表 A.9) 注 2:应根据安全完整性等级选择适当的技术/措施。					
* 应根据安全完整性等级选择已编号的技术/措施。					

表 A.7 软件安全确认(见 7.7)

技术/措施*	Ref	SIL1	SIL2	SIL3	SIL4
1 概率测试 (略)	C.5.1	—	R	R	HR
2 仿真/建模	表 B.5	R	R	HR	HR
3 功能和黑盒测试	B.5.1 B.5.2 表 B.3	HR	HR	HR	HR
注:应根据安全完整性等级选择适当的技术/措施。					
* 应根据安全完整性等级选择已编号的技术/措施。					

表 A.8 修改(见 7.8)

技术/措施*	Ref	SIL1	SIL2	SIL3	SIL4
1 影响分析	C.5.23	HR	HR	HR	HR
2 再验证已改变的软件模块	C.5.23	HR	HR	HR	HR
3 再验证已受影响的软件模块	C.5.23	R	HR	HR	HR
4 再确认整个系统	C.5.23	—	R	HR	HR
5 软件配置管理	C.5.24	HR	HR	HR	HR
6 数据记录和分析	C.5.2	HR	HR	HR	HR
注:应根据安全完整性等级选择适当的技术/措施。					
* 应根据安全完整性等级选择已编号的技术/措施。					

表 A.9 软件验证(见 7.9)

技术/措施*	Ref	SIL1	SIL2	SIL3	SIL4
1 形式证明	C.5.13	—	R	R	HR
2 概率测试(略)	C.5.1	—	R	R	HR
3 静态分析	B.6.4 表 B.8	R	HR	HR	HR
4 动态分析和测试	B.6.5 表 B.6.8	R	HR	HR	HR
5 软件复杂性度量	C.5.14	R	R	R	R
软件模块测试和集成	见表 A.5				
可编程电子集成测试	见表 A.6				
软件系统测试(确认)	见表 A.7				

注 1：为方便进行本表中所有验证行为，本表不对表 A.5 和表 A.6 中对自身进行验证活动的动态测量元素确认提出额外要求，也不对 GB/T 20438 中对安全要求规定的符合性说明的额外的软件确认提出要求(end-end 验证)。

注 2：验证贯穿 GB/T 20438.1、GB/T 20438.2 和 GB/T 20438.3 的边界，因此安全相关系统的第一个验证可参照早期系统水平规范。

注 3：在软件安全生命周期的早期阶段，验证是静态的，例如通过检查、复审、形式证明。当产生代码后可进行动态测试。验证需要两类信息的综合。如通过静态方法对软件模块的代码验证包括软件检查、走查、静态分析、形式证明等技术，动态方法的代码验证包括功能测试、白盒测试、统计测试。通过两类证据的结合证明每一软件模块满足有关规定。

a 应根据安全完整性等级选择已编号的技术/措施。
应根据安全完整性等级选择适当的技术/措施。

表 A.10 功能安全评估(见第 8 章)

技术/措施*	Ref	SIL1	SIL2	SIL3	SIL4
1 检查列表	B.2.5	R	R	R	R
2 判定/真值表	C.6.1	R	R	R	R
3 软件复杂性度量	C.5.14	R	R	R	R
4 失效分析	表 B.4	R	R	HR	HR
5 多种软件的共同原因失效分析(如实际使用多种软件)等	C.6.3	—	R	HR	HR
6 可靠性块图	C.6.5	R	R	R	R

a 应根据安全完整性等级选择适当的技术/措施。

附 录 B
（规范性附录）
详 细 表 格

注：本部分给出 GB/T 20438.7 中技术/措施的详细描述。

表 B.1 设计和编码标准（参见表 A.4）

技术/措施*	Ref	SIL1	SIL2	SIL3	SIL4
1 编码标准的使用	C.2.6.2	HR	HR	HR	HR
2 无动态对象	C.2.6.3	R	HR	HR	HR
3a 无动态变量	C.2.6.3	—	R	HR	HR
3b 动态变量安装的在线检查	C.2.6.4	—	R	HR	HR
4 中断的有限使用	C.2.6.5	R	R	HR	HR
5 指针的有限使用	C.2.6.6	—	R	HR	HR
6 递归的有限使用	C.2.6.7	—	R	HR	HR
7 不存在高级语言程序中的无条件转移	C.2.6.2	R	HR	HR	HR
注：在使用编译器时如果在运行之前对所有动态变量和对象分配了足够的内存，或者对内存在线分配的校正插入了运行检查，则无需应用措施 2 和 3a。					
a 应根据安全完整性等级选择适当的技术/措施。其他可选择的或等价技术/措施应用数字说明，只需满足一种其他可选择的或等价技术/措施。					

表 B.2 动态分析和测试（参见表 A.5 和表 A.9）

技术/措施*	Ref	SIL1	SIL2	SIL3	SIL4
1 根据边界值分析执行测试用例	C.5.4	R	HR	HR	HR
2 根据错误推测执行测试用例	C.5.5	R	R	R	R
3 根据错误播种执行测试用例	C.5.6	—	R	R	R
4 性能建模	C.5.20	R	R	R	HR
5 等价类和输入划分测试	C.5.7	R	R	R	HR
6 基于结构的测试	C.5.8	R	R	HR	HR
注：测试用例分析在子系统级进行并基于规范和/或规范和代码。					
a 应根据安全完整性等级选择适当的技术/措施。					

表 B.3 功能和黑盒测试(参见表 A.5、表 A.6 和表 A.7)

技术/措施[a]	Ref	SIL1	SIL2	SIL3	SIL4
1 根据因果图执行测试用例	B.6.6.2	—	—	R	R
2 原型设计/模拟动作	C.5.17	—	—	R	R
3 边界值分析	C.5.4	R	HR	HR	HR
4 等价类和输入划分测试	C.5.7	R	HR	HR	HR
5 过程仿真	C.5.18	R	R	R	R
注 1:测试用例分析在软件系统级进行并只根据规范。 注 2:仿真的完整性将依靠安全完整性等级、复杂性和应用。					
[a] 应根据安全完整性等级选择适当的技术/措施。					

表 B.4 失效分析(参见表 A.10)

技术/措施[a]	Ref	SIL1	SIL2	SIL3	SIL4
1a 因果图	B.6.6.2	R	R	R	R
1b 事件树分析	B.6.6.3	R	R	R	R
2 故障树分析	B.6.6.5	R	R	HR	HR
3 失效模式、影响和危险程度分析	B.6.6.4	R	R	HR	HR
4 Monte-Carlo 仿真	C.6.6	R	R	R	R
注:为了将软件分类到最合适的软件完整性等级,应进行初步危险分析。					
[a] 应根据安全完整性等级选择适当的技术/措施。其他可选择的或等价技术/措施应用数字说明,只需满足一种其他可选择的或等价技术/措施。					

表 B.5 建模(参见表 A.7)

技术/措施[a]	Ref	SIL1	SIL2	SIL3	SIL4
1 数据流图	C.2.2	R	R	R	R
2 有限状态机	B.2.3.2	—	R	HR	HR
3 形式方法	C.2.4	—	R	R	HR
4 性能建模	C.5.20	R	HR	HR	HR
5 时间 Petri 网	B.2.3.3	—	R	HR	HR
6 原型设计/动画	C.5.17	R	R	R	R
7 结构图	C.2.3	R	R	R	HR
注:未在表中列出的特定技术也应在考虑范围之内,且应符合 GB/T 20438。					
[a] 应根据安全完整性等级选择适当的技术/措施。					

表 B.6 性能测试(参见表 A.5 和表 A.6)

技术/措施*	Ref	SIL1	SIL2	SIL3	SIL4
1 雪崩/过载测试	C.5.21	R	R	HR	HR
2 响应定时和存储约束	C.5.22	HR	HR	HR	HR
3 性能要求	C.5.19	HR	HR	HR	HR
* 应根据安全完整性等级选择适当的技术/措施。					

表 B.7 半形式方法(参见表 A.1、表 A.2 和表 A.4)

技术/措施*	Ref	SIL1	SIL2	SIL3	SIL4
1 逻辑/功能块图	见注	R	R	HR	HR
2 时序图	见注	R	R	HR	HR
3 数据流图	C.2.2	R	R	R	R
4 有限状态机制/状态转换图	B.2.3.2	R	R	HR	HR
5 时间 Petris 网	B.2.3.3	R	R	HR	HR
6 判定/真值表	C.6.1	R	R	HR	HR
注:逻辑/功能块图和时序图在 GB/T 15969.3 中有描述。					
* 应根据安全完整性等级选择适当的技术/措施。					

表 B.8 静态分析(参见表 A.9)

技术/措施*	Ref	SIL1	SIL2	SIL3	SIL4
1 边界值分析	C.5.4	R	R	HR	HR
2 检查列表	B.2.5	R	R	R	R
3 控制流分析	C.5.9	R	HR	HR	HR
4 数据流分析	C.5.10	R	HR	HR	HR
5 错误推测	C.5.5	R	R	R	R
6 Fagan 检查	C.5.15	—	R	R	HR
7 寄生回路分析	C.5.11	—	—	R	R
8 符号执行	C.5.12	R	R	HR	HR
9 走查/设计复审	C.5.16	HR	HR	HR	HR
* 应根据安全完整性等级选择适当的技术/措施。					

表 B.9 模块化方法(参见表 A.4)

技术/措施[a]	Ref	SIL1	SIL2	SIL3	SIL4
1 软件模块规模限制	C.2.9	HR	HR	HR	HR
2 信息隐蔽/封装	C.2.8	R	HR	HR	HR
3 参数号限制	C.2.9	R	R	R	R
4 子程序和功能的单入口/单出口	C.2.9	HR	HR	HR	HR
5 充分定义接口	C.2.9	HR	HR	HR	HR
注:除去信息隐蔽/封装的所有技术的信息见 GB/T 20438.7—2006 中 C.2.9。					
[a] 没有一个单一的技术是充分的,应考虑所有适当的技术。					