ICS 43.040
T 35/39

团 体 标 准

T/CFEII 0016-2023

汽车软件开发能力要求

Automotive software development capability requirement

2023-12-25 发布 2023-12-25 实施

中国电子信息行业联合会 发布

目　　次

前　言

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

请注意本文件的其他内容可能涉及专利，本文件的发布机构不承担识别这些专利的责任。

本文件由软件成本度量和价值评估分会提出。

本文件由中国电子信息行业联合会归口。

本文件起草单位：广州赛宝认证中心服务有限公司、科大讯飞股份有限公司、东风汽车集团有限公司研发总院、中电金信数字科技集团股份有限公司、北京理想汽车有限公司、北京新能源汽车股份有限公司、潍柴动力股份有限公司电控与软件研究院、东软集团股份有限公司、宇通客车股份有限公司、陕西重型汽车有限公司、广西玉柴机器股份有限公司、麦格纳动力总成（江西）有限公司、星河智联汽车科技有限公司、武汉环宇智行科技有限公司、沈阳美行科技股份有限公司、威海神舟信息技术研究院有限公司。

本文件主要起草人：翟宏宝、金戈、黄岚、施展、王旭飞、毛志飞、周文峰、余来星、董雷、方文韬、张庆浩、段升龙、沈楠、吕传成、刘兴义、刘建飞、李春林、汪鹏、李进、安永杰、王辉、叶宇、朱赛春、王雷、金鑫、杜思宁、舒选才、黄莉、曹祁生、任振刚、谢成龙、闻艺、孙海铭、王松、毛慧丽、唐百惠。

本文件为首次制定。

引　言

新能源汽车和智能网联汽车已成为全球汽车产业发展的战略方向。软件作为新能源汽车和智能网联汽车的重要基础，在整车中的比重越来越大。据有关分析，汽车中的软件代码量已突破5亿行，“软件定义汽车”的时代已经到来。

然而，近些年因软件质量问题带来的汽车事故层出不穷，软件质量问题导致的汽车召回事件屡创新高。各汽车企业正在抓紧制定应对软件质量问题的策略。

我国相关部委高度重视汽车产业的发展，近些年密集出台了《智能汽车创新发展战略》、《工业和信息化部关于加强智能网联汽车生产企业及产品准入管理的意见》、《关于开展智能网联汽车准入和上路通行试点工作的通知》、《关于开展智能网联汽车“车路云一体化”应用试点工作的通知》等系列意见和通知要求。从相关意见和通知中可以看出，国家对智能网联汽车的功能安全、网络安全和预期功能安全等都提出了明确的要求。作为汽车功能安全、网络安全和预期功能安全中非常重要部分的汽车软件也受到了前所未有的关注。

本文件旨在指导汽车软件开发组织在开发车载软件时应考虑的质量要求，及对软件开发的正确性、合理性和有效性的评估要求。

本文件典型的应用场景包括：

1）软件开发组织利用本文件建设汽车软件开发能力，并进行测量、评估和改进；

2）软件采购方利用本文件对软件开发组织的软件能力进行评估；

3）第三方机构依据本文件对软件开发组织的软件开发能力及软件产品质量进行客观评估。

汽车软件开发能力要求

1 范围

本文件规定了软件开发组织管理、软件开发项目管理、分布式软件开发管理以及软件开发过程管理四个方面的要求。

本文件提供了汽车软件开发过程能力评估和产品评估的方法。

本文件适用于搭载在汽车电子控制单元（ECU）中的软件的开发，不适用于汽车有关的云端软件、工具软件的开发管理。

2 规范性引用文件

下列文件中的内容通过文中的规范性引用而构成本文件必不可少的条款。 其中，注日期的引用文件，仅该日期对应的版本适用于本文件；不注日期的引用文件，其最新版本（ 包括所有的修改单）适用于本文件。

ISO/IEC 24765-2017 系统和软件工程 术语

ISO 26262:2018 道路车辆 功能安全

ISO/SAE 21434:2021 道路车辆 网络安全工程

ISO 21448:2022 道路车辆 预期功能安全

3 术语、定义和缩略语

3.1 术语和定义

3.1.1

产品 Product

为交付给用户而设计的计算机程序、规程和相关文档的完整集合。

3.1.2

工作产品 Work Product

来自过程、活动或任务的输出，可以是独立输出或者是解决方案的一部分。

3.1.3

项目 Project

受管理的一组相互关联的活动和资源（包括人员），能够向客户或最终用户交付一个或多个解决方案。一个项目通常有一个预期的开始（即项目启动）和结束点，但也可以是连续不中断的。项目通常根据计划和需求运作。

3.1.4

能力 Capability

组织或个人通过利用资源、知识和技能实现预期结果的本领。

注：组织可以是项目组、部门或整个公司。

3.1.5

软件重要性等级 Software Importance Level

根据软件危害等级和危害可控性等级确定的软件在汽车中的重要性级别。

3.1.6

配置项 Configuration Item

配置管理可识别的工作产品，作为一个单独的实体被配置管理活动管理。

4 标准综述

4.1 软件重要性等级

软件重要性等级根据软件危害等级和危害可控性等级确定。软件危害等级是软件失效可能造成影响的程度，按照下表确定：

表1 软件危害等级

软件危害等级	软件失效可能带来的影响
H1	无影响或轻微影响，可能是下列情形中的一种或多种： a）对人员的伤害可忽略； b）不会造成网络安全事件； c）对数据的损坏或遗失程度可忽略； d）对系统及周边系统的破坏可忽略； e）对经济或社会的损失和影响可忽略。
H2	轻度影响，可能是下列情形中的一种或多种： a）对人员可能造成轻度伤害； b）虽然存在网络安全事件，但不会影响正常驾驶； c）虽然会造成数据损坏、遗失，但不影响软件正常使用； d）对系统及周边系统可能造成轻度破坏，但不影响系统使用； e）轻度的经济或社会损失。
H3	严重性影响，可能是下列情形中的一种或多种： a）对人员会造成严重伤害，但不会危及生命； b）面临严重的网络安全事件，导致软件无法正常使用； c）重要数据损坏、遗失，导致软件无法正常使用； d）导致系统及周边系统无法正常工作； e）严重的经济或社会损失。
H4	灾难性影响，可能是下列情形中的一种或多种： a）导致人员死亡； b）面临灾难性的网络安全事件，导致汽车无法正常驾驶； c)核心数据损坏、遗失，导致系统无法正常使用； d）导致系统及周边系统瘫痪； e）重大经济或社会损失。

危害可控性是指通过所涉及人员的及时反应或通过外部措施的支持避免伤害的能力，危害可控性等级按照下表确定：

表2 危害可控性等级

软件可控性等级	软件失效的可控性
C1	可控，软件的失效完全可以通过驾驶员正常操作避免危害。
C2	简单可控，软件的失效可以通过驾驶员或者交通参与者的干预避免危害。
C3	一般可控，软件的失效需要驾驶员或者交通参与者通过高度的注意力和一系列专业的操作来避免危害。
C4	难以控制或不可控，软件的失效难以避免危害。

软件重要性等级根据软件危害等级和和危害可控性等级构成的二维矩阵来建立，软件重要性等级分为P，M，H三个级别。P表示软件重要性等级低，通过建立软件过程体系和维持软件过程体系的正常运行就可以满足需要；M表示软件重要性等级中，除了建立保持软件过程体系的正常运行外，还需要通过一定的软件技术方法的应用来满足需要；H表示软件重要性等级高，除软件过程体系外，需要严格的软件管控和质量措施来满足需要。软件重要性等级如下表所示：

表3 软件重要性等级

软件危害等级	软件可控性等级			
	C1	C2	C3	C4
H1	P	P	P	M
H2	P	M	M	M
H3	P	M	H	H
H4	M	M	H	H

4.2 标准过程框架

汽车软件开发能力要求分为组织级软件管理，项目级软件管理，分布式软件开发管理，软件开发过程管理四个过程组，每个过程组由一系列过程组成。组织级软件管理过程组包括软件开发治理、软件过程体系、软件工具管理、软件人员管理、信息安全管理五个过程要求。项目级软件管理过程组包括软件项目管理、软件配置管理、软件变更管理三个过程要求。分布式软件开发管理过程组包括供方选择、协议管理、供方监控三个过程要求。软件开发过程管理过程组包括软件需求分析、软件设计、软件实现、软件单元测试、软件集成和测试、软件合格测试、软件发布七个过程组成。

汽车软件开发能力框架如下图：

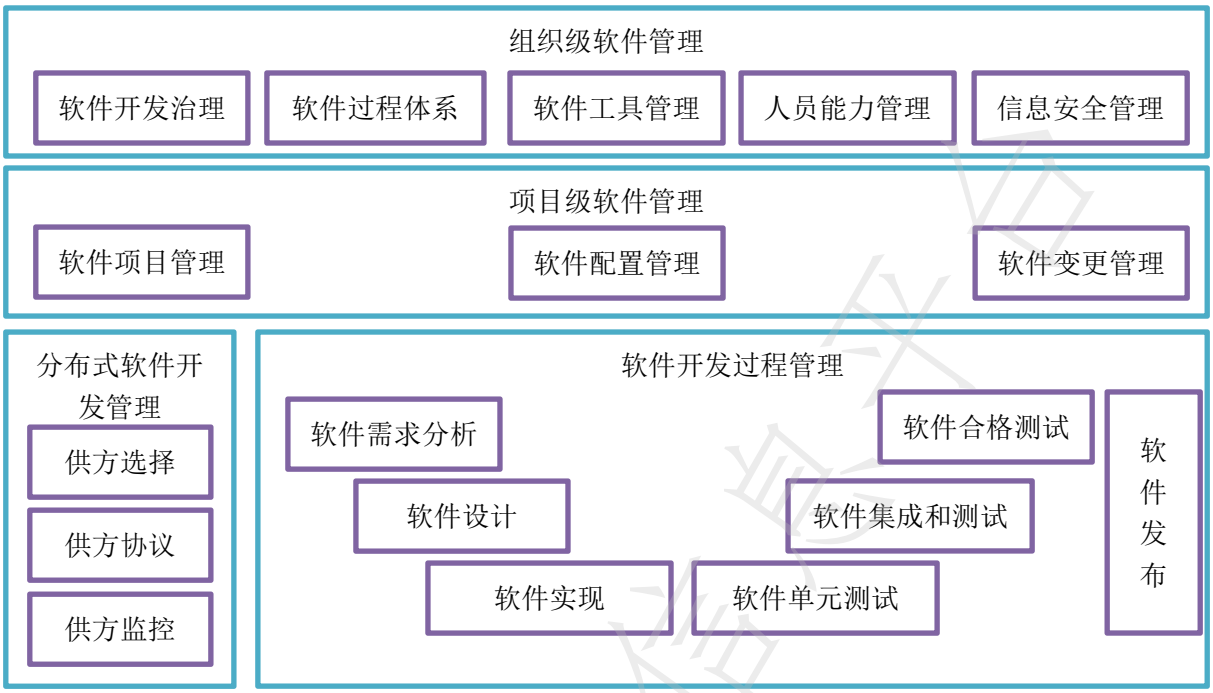

图1 汽车软件开发能力框架

4.3 标准结构说明

本文件第5章至第8章为标准的要求部分。每个章节分为若干个过程，每个过程包含了若干个要求。

以“5.4人员能力管理[PCM]”为例，说明标准结构如下：

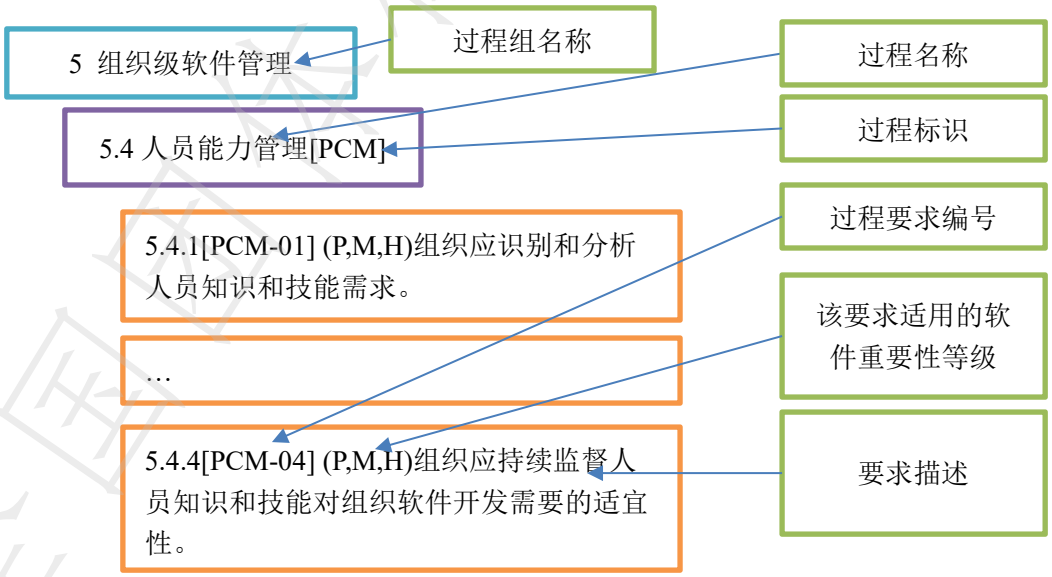

图2 标准结构说明

5 组织级软件管理

5.1 软件开发治理[SDG]

软件开发治理的目的是组织为开发汽车软件而确定的高层次方针目标、过程框架、人员职责权限和相关资源等。

5.1.1 [SDG-01] (P,M,H)组织应制定软件开发方针和目标。

组织应分析适用的软件重要性等级，并根据软件重要性等级建立软件开发方针和目标。

5.1.2 [SDG-02] (P,M,H)组织应确定软件开发过程要求。

组织应基于软件开发方针和目标，定义组织的软件开发生命周期，识别软件开发过程的要求，并持续维护软件开发过程要求的适宜性。

5.1.3 [SDG-03] (P,M,H)组织应明确软件开发职责和权限。

组织应为软件开发过程的实施明确职责和权限，并确保相关人员具备履行职责和权限的能力要求。

5.1.4 [SDG-04] (P,M,H)组织应为软件开发提供必要的资源。

组织应为软件开发活动提供必要的人员、设备、工具和资金等，确保软件开发活动的顺利开展。

5.2 软件过程体系[SPS]

软件过程体系的目的是组织为确保软件开发活动有序进行而建立的一套过程管理体系。

5.2.1 [SPS-01] (P,M,H)组织应建立和维护软件过程管理体系。

组织应基于确定的软件开发过程要求，建立软件开发过程管理体系，并确保在组织范围内的应用。

注：软件开发过程要求见[SDG-02]。

5.2.2 [SPS-02] (P,M,H)组织应持续改进软件过程管理体系，并持续保持其有效性。

组织应建立机制持续识别软件过程管理体系的改进需求，并持续保持软件过程管理体系能满足客户及组织发展的需要。

5.2.3 [SPS-03] (P,M,H)组织应定期对软件过程管理体系进行审计，并持续监督体系的有效性。

组织应建立对软件过程管理体系的监督机制，定期审计软件过程管理体系的符合性和有效性。

软件过程管理体系审计发现的改进机会，应按[SPS-02]实施。

5.3 软件工具管理[STM]

软件工具管理的目的是确保影响软件的工具能得到有效的管理。

5.3.1 [STM-01] (P,M,H)组织应选择合适的工具管理软件开发活动。

组织应基于软件过程管理体系要求选择合适的工具管理软件开发活动。可能的工具包括需求管理工具、设计工具、IDE工具、测试工具、配置管理工具、缺陷管理工具、变更管理工具等。

5.3.2 [STM-02] (M,H)组织应为保证软件的追溯性建立合适的工具链。

对于软件重要性等级为M和H的软件开发，组织应通过工具管理软件开发工作产品之间的追溯性。

注：具体追溯性管理要求，见“8 软件开发过程管理”要求。

5.3.3 [STM-03] (P,M,H)组织应确保软件工具持续可用。

组织应确保使用工具的合法性和可用性，确保工具在软件开发全生命周期中的可用性。

5.4 人员能力管理[PCM]

人员能力管理的目的是从组织层面识别软件开发人员的能力，并持续保持软件开发人员能力满足组织需要。

5.4.1 [PCM-01] (P,M,H)组织应识别和分析人员知识和技能需求。

组织应基于客户、软件过程管理体系等要求，识别软件开发人员的知识和技能需求。

5.4.2 [PCM-02] (P,M,H)组织应建立获取人员知识和技能的机制。

组织应建立合适的知识和技能获取机制（如：线下培训、E-Learning、导师制等），确保软件开发人员能够持续获得所需的知识和技能。

5.4.3 [PCM-03] (P,M,H)组织应确保人员的知识和技能满足组织软件开发需要。

组织应对软件开发人员的知识和技能进行评价，确保软件开发人员的知识和技能满足其职责的需要。

5.4.4 [PCM-04] (P,M,H)组织应持续监督人员知识和技能对组织软件开发需要的适宜性。

组织应对人员的知识和技能需求进行持续评估，并采取措施确保人员知识和技能满足其职责和业务发展需要。

5.5 信息安全管理[ISM]

信息安全管理的目的是确保软件开发相关信息的安全性，避免对软件开发活动带来影响。

5.5.1 [ISM-01] (P,M,H)软件开发相关信息应根据信息安全管理体系进行管理。

组织应建立信息安全管理体系，并基于信息安全管理体系管理软件开发信息的安全性。

6 项目级软件管理

6.1 软件项目管理[SPM]

软件项目管理的目的是在规定的需求和约束下，利用必要的资源实现项目的目标。

6.1.1 [SPM-01] (P,M,H)应明确软件开发项目的人员及其职责和权限。

为项目提供足够的人员，明确其职责和权限，并确保其理解职责和权限要求。

6.1.2 [SPM-02] (P,M,H)应对软件开发项目进行策划。

项目应定义符合客户和/或组织要求的生命周期模型，策划项目各项内容，确保相关干系人对策划内容达成一致。

必要时，应对项目的策划进行更新。

6.1.3 [SPM-03] (P,M,H)应根据组织定义的软件过程管理体系进行裁剪。

项目应裁剪组织定义的软件过程管理体系，以适应项目特征。

6.1.4 [SPM-04] (P,M,H)应对软件开发的重用进行分析。

项目应分析软件开发生命周期中的重用内容，界定本项目需开发的范围。

6.1.5 [SPM-05] (P,M,H)应对软件项目进行监控。

项目应采取合适的方式，监控项目执行与策划的偏差，并对偏差采取必要措施。

6.1.6 [SPM-06] (M,H) 应建立软件案例，以满足特定要求。

对于M和H类的软件项目，应基于客户要求及软件重要性等级分析结果，建立软件案例，确保软件相关论据得到有力支撑。

注1：如软件涉及到功能安全相关内容（ASIL A、B、C、D），应建立满足ISO 26262 Part2 第6.4.8章节 安全案例要求的软件案例。

注2：如软件涉及到网络安全相关内容，应建立满足ISO/SAE 21434 第6.4.7网络安全案例要求的软件案例。

6.1.7 [SPM-07] (M,H)应对软件开发过程和工作产品进行评估。

对于M和H类的软件项目，应对软件相关论据进行评估，确保满足相关标准要求，并有效控制软件质量风险。

6.2 软件配置管理[SCM]

软件配置管理的目的是保证软件配置项的一致性、完整性。

6.2.1 [SCM-01] (P,M,H)应为软件项目建立配置管理系统。

项目应建立配置管理系统，确保配置项得到一致和受控的访问。

6.2.2 [SCM-02] (P,M,H)应确保项目相关的工作产品纳入配置管理系统管理。

项目应识别需纳入配置管理的工作产品，并作为配置项在配置管理系统中受控。

6.2.3 [SCM-03] (P,M,H)应确保软件相关工作产品标识的唯一性。

项目应为配置项建立唯一标识，并确保在配置库中的唯一性。

6.2.4 [SCM-04] (P,M,H)应为软件相关工作产品建立基线。

项目应识别需建立的配置基线，并根据对配置管理的策划管理基线。

6.3 软件变更管理[CGM]

软件变更管理的目的是确保来自各方的变更请求得到有效的控制和实施。

6.3.1 [CGM-01] (P,M,H)应建立软件变更管理系统。

项目应建立变更管理系统，管理项目变更请求并跟踪变更的状态。

6.3.2 [CGM-02] (P,M,H)应确保软件变更得到有效控制。

项目应分析和评估变更的影响，对变更进行批准，并对变更实施的结果进行确认。

6.3.3 [CGM-03] (M,H)应确保变更内容的追溯性。

对于M类和H类软件的变更，应建立变更的内容及其来源之间的追溯关系。

7 分布式软件开发管理

7.1 供方选择[SSS]

供方选择的目的是确保识别出满足项目需要的潜在供方。

7.1.1 [SSS-01] (P,M,H)应制定供方有关的技术需求。

项目应制定由供方提供服务的技术需求。

注：供方提供的服务可能是提供软件，软件的一部分，或者是软件开发中的技术支持等。

7.1.2 [SSS-02] (P,M,H)应对备选供方的软件开发能力进行评估。

应根据技术需求对备选供方的能力进行评估，并确定最佳供方。

7.2 供方协议[SPA]

供方协议的目的是跟供方建立正式的协议，以协议作为双方工作依据。

7.2.1 [SPA-01] (P,M,H)应建立与软件供方之间的接口协议。

项目应和软件供方之间建立明确的接口协议，以明确供方提供软件的软件重要性等级、软件开发过程职责等。

7.2.2 [SPA-02] (P,M,H)软件供方的接口协议应得到签署。

相关方应充分沟通讨论接口协议内容，并正式签署接口协议。

7.2.3 [SPA-03] (P,M,H)应与供方签订正式的包供方协议。

项目应与供方签订正式的供方协议，供方协议应包含约定的交付范围、进度、质量、验收标准等方面的要求。

注：接口协议通常会作为正式协议的一部分。

7.3 供方监控[SSM]

供方监控的目的是对供方安装协议履行情况进行监控，确保供方的进展和工作产品质量符合项目需要。

7.3.1 [SSM-01] (P,M,H)应对供方进展情况进行评审。

项目应对供方工作进展情况进行评审，对出现的偏差应按照约定采取措施并跟踪关闭。

7.3.2 [SSM-02] (P,M,H)应对供方提供的工作产品进行评审。

项目应按照约定的方式，对供方提供的工作产品进行评审，对识别的工作产品质量问题应采取措施并跟踪关闭。

7.3.3 [SSM-03] (P,M,H)应对供方提供的交付产物进行验证和确认。

项目应对供方提供的交付结果进行验证和确认，对验证和确认的不符合项采取措施并跟踪关闭。

8 软件开发过程管理

8.1 软件需求分析[SRA]

软件需求分析的目的是将输入给软件的要求转化为对软件需求。

8.1.1 [SRA-01] (P,M,H)应建立条目化的软件需求。

项目应基于输入给软件的要求进行条目化分析和整理，形成对软件的需求定义。

注：输入给软件的要求可能来自于客户要求，也可能是系统需求等。

8.1.2 [SRA-02] (P,M,H)应对每条软件需求进行分析，并确定优先级。

项目应对每条软件需求，分析可实现性和可验证性，并识别潜在风险。建立优先级准则，并基于准则确定每条需求的优先级。

注：对于M类或H类的软件的需求，可能存在软件功能安全需求或软件网络安全需求，应对每条需求标明功能安全完整性等级（ASIL）或网络安全CAL的要求。

8.1.3 [SRA-03] (P,M,H)应明确软件接口需求。

项目应识别与软件交互的接口，并对这些接口进行需求定义。

注：可能和软件交互的接口包括，软件和硬件的接口，或软件之间的接口，通讯接口等。

8.1.4 [SRA-04] (M,H)应分析软件与周边环境之间的影响。

应分析软件与周边环境之间可能存在的相互影响。

注：对于涉及网络安全要求的软件，应采用TARA分析相关的影响。

8.1.5 [SRA-05] (M,H)应建立软件需求与其来源之间的双向追溯机制。

应建立软件需求及其来源之间的双向追溯关系。

注：软件需求的来源可能是客户需求或系统需求。

8.1.6 [SRA-06] (P,M,H)应采用合适的方式验证软件需求。

应采用合适的验证方式对软件需求的正确性、完整性、一致性等进行验证。可能的验证方式包括评审、仿真、模拟等。

8.2 软件设计[SDA]

软件设计的目的是将软件需求转化为软件实现的表述，并确保满足软件需求的要求。

8.2.1 [SDA-01] (P,M,H)应对软件进行开发和重用分析。

应对软件进行开发、重用和/或购买分析。

8.2.2 [SDA-02] (P,M,H)应对软件进行架构设计。

应设计软件组件及组件间的接口关系，确保软件需求分配到各架构设计要素。

注1：如果软件存在功能安全要求，应按照ISO 26262 Part6 第7章 软件架构设计的要求进行设计。

注2：如果软件存在网络安全要求，应按照ISO/SAE 21434 第10.4.1 设计的要求进行设计。

8.2.3 [SDA-03] (P,M,H)应对软件组件进行详细设计。

应对每个开发的软件组件进行详细设计，包括组件的单元及其接口设计等。

注1：如果软件存在功能安全要求，应按照ISO 26262 Part6 第8章 软件单元设计和实现的要求进行设计。

注2：如果软件存在网络安全要求，应按照ISO/SAE 21434 第10.4.1 设计的要求进行设计。

8.2.4 [SDA-04] (P,M,H)应开展软件动态行为设计。

应分析软件设计的动态行为，确保业务逻辑的实现。应区分架构层面和详细设计层面的动态行为。

8.2.5 [SDA-05] (M,H)应对软件进行最坏情况分析。

应分析最坏情况下的软件运行场景，确保软件的健壮性。通常需要考虑的最坏情况包括存储空间、CPU负载、网络带宽等。

8.2.6 [SDA-06] (M,H)应建立软件设计与其输入之间的追溯机制。

应建立软件架构设计与软件需求之间的双向追溯关系，以及设计各层级之间的追溯关系。

8.2.7 [SDA-07] (P,M,H)应对软件设计进行评审。

应组织对软件设计的评审，确保软件设计的符合软件需求及相关规范的要求。

8.3 软件实现[SIM]

软件实现的目的是将软件设计转化为软件代码，以实现软件需求。

8.3.1 [SIM-01] (P,M,H)应针对软件设计开发软件单元。

应根据软件详细设计实现软件单元。

注1：软件单元可能是基于模型的实现，或手工代码方式的实现。

注2：如果软件存在功能安全要求，应按照ISO 26262 Part6 第8章 软件单元设计和实现的要求进行实现。

注3：如果软件存在网络安全要求，应按照ISO/SAE 21434 第10.4.1 设计的要求进行实现。

8.3.2 [SIM-02] (P,M,H)应对实现的软件单元进行评审。

应对实现的软件单元进行评审，确保软件单元符合详细设计及组织相关规定。

8.3.3 [SIM-03] (M,H)应对软件单元执行静态分析。

应使用静态分析工具，执行单元的静态分析。静态分析应符合相关标准和法规的要求。

注1：如果软件存在功能安全要求，应按照ISO 26262 Part6 第9章 软件单元验证的要求进行静态分析。

注2：如果软件存在网络安全要求，应按照ISO/SAE 21434 第10.4.1 设计的要求进行静态分析。

8.3.4 [SIM-04] (M,H)应建立软件单元与其输入之间的追溯机制。

应建立软件单元与软件详细设计之间的追溯关系，以及软件单元与软件需求之间的追溯关系。

8.4 软件单元测试[SUT]

软件单元测试的目的是确保软件单元符合软件详细设计要求。

8.4.1 [SUT-01] (M,H)应确定软件单元测试策略。

应建立软件单元测试的策略，包括软件单元测试覆盖率目标，软件单元测试准入准出准则、软件单元回归测试策略、软件单元测试环境及资源等。

8.4.2 [SUT-02] (M,H)应对软件单元进行测试设计并形成单元测试规范。

应设计并开发软件单元测试用例，形成软件单元测试规范。

注1：如果软件存在功能安全要求，应按照ISO 26262 Part6 第9章 软件单元验证的要求进行单元测试。

注2：如果软件存在网络安全要求，应按照ISO/SAE 21434 第10.4.2 集成和验证的要求进行单元测试。

8.4.3 [SUT-03] (M,H)应对软件单元执行测试并记录观测结果。

应记录测试用例执行的观测结果，并及时沟通每轮单元测试结果。

8.4.4 [SUT-04] (M,H)应总结并报告单元测试结果。

对单元测试及单元回归测试结果进行总结分析和报告并及时和相关人员进行沟通。

8.4.5 [SUT-05] (M,H)应建立软件单元测试及其输入之间的追溯机制。

应建立单元测试用例与详细设计之间的追溯关系，以及单元测试结果和单元测试用例之间的追溯关系。

8.5 软件集成和测试[SIT]

软件集成和测试的目的是将软件代码形成可执行的软件，并确保集成的软件符合软件架构设计的要求。

8.5.1 [SIT-01] (P,M,H)应建立软件集成策略和集成测试策略。

应建立软件集成的策略和软件集成测试的策略。

软件集成策略可能包括集成内容、集成顺序、集成环境要求，集成准入准出准则等。

软件集成测试策略可能包括软件集成测试内容，软件集成测试准入准出准则、软件集成回归测试策略、软件集成测试环境及资源等。

8.5.2 [SIT-02] (P,M,H)应对软件不同集成层级进行集成测试设计，并形成集成测试规范。

应基于软件架构设计开展软件集成测试设计，并形成软件集成测试用例，形成集成测试规范。

注1：如果软件存在功能安全要求，应按照ISO 26262 Part6 第10章 软件集成和验证的要求进行集成和集成测试。

注2：如果软件存在网络安全要求，应按照ISO/SAE 21434 第10.4.2 集成和验证的要求进行集成和集成测试。

8.5.3 [SIT-03] (P,M,H)应对软件开展不同层级的集成。

应根据软件集成策略，从代码单元开始，逐层集成，形成可执行的软件。

8.5.4 [SIT-04] (P,M,H)应对软件不同层级的集成进行测试并记录测试结果。

应对每层级集成的结果进行验证，确保最终的结果符合软件架构设计要求，并记录每次集成测试结果。

8.5.5 [SIT-05] (P,M,H)应总结和报告软件集成和集成测试结果。

应对软件集成结果以及软件集成测试结果进行总结和报告，并及时和相关干系人进行沟通。

8.5.6 [SIT-06] (M,H)应建立软件集成测试及其输入之间的追溯机制。

应建立软件集成测试用例和软件架构设计之间的追溯关系，应建立软件集成测试结果和集成测试用例之间的追溯关系。

8.6 软件合格测试[SQT]

软件合格测试的目的是确保开发出来的软件满足软件需求的要求。

8.6.1 [SQT-01] (P,M,H)应建立软件合格测试的策略。

应建立软件合格测试的策略。

软件合格测试策略可能包括软件合格测试内容，软件合格测试准入准出准则、软件合格回归测试策略、软件合格测试环境及资源等。

8.6.2 [SQT-02] (P,M,H)应对软件进行合格测试设计，并形成合格测试规范。

应基于软件需求开展软件合格测试设计，并形成软件合格测试用例，形成合格测试规范。

注1：如果软件存在功能安全要求，应按照ISO 26262 Part6 第11章 嵌入式软件测试的要求进行软件合格测试。

注2：如果软件存在网络安全要求，应按照ISO/SAE 21434 第10.4.2 集成和验证的要求进行软件合格测试。

8.6.3 [SQT-03] (P,M,H)应对软件执行合格测试并记录测试结果。

应对每条软件需求进行验证，确保软件符合软件需求的要求，并记录每次软件合格测试结果。

8.6.4 [SQT-04] (P,M,H)应总结和报告软件合格测试结果。

应对软件合格测试结果进行总结和报告，并及时和相关干系人进行沟通。

8.6.5 [SQT-05] (P,M,H)应建立软件合格测试用例和软件需求之间的追溯机制。

应建立软件合格测试用例与软件需求之间的追溯关系，以及软件合格测试结果与软件合格测试用例之间的关系。

8.7 软件发布[REL]

软件发布的目的是控制交付给客户的软件版本，确保发布软件的一致性。

8.7.1 [REL-01] (P,M,H)应建立软件发布计划，明确每个发布版本包含的需求内容。

应根据客户或组织要求，策划项目应发布的软件版本，以及每个版本包含的实现的软件功能和其他软件质量属性。

8.7.2 [REL-02] (P,M,H)应确定每个发布版本的发布包的内容、打包形式和发布渠道等。

应在每次发布前，明确每个软件发布包包含的可执行程序文件，说明文件以及其他需要交付给客户的内容，并确定需采取的交付媒介类型，和交付的渠道等。

8.7.3 [REL-03] (P,M,H)应提供每次发布的发布说明。

为每次发布提供发布的说明，通常发布说明可能包括：发布的版本，发布的内容，已解决的缺陷，遗留的已知缺陷，安装或升级要求，兼容性要求，其他需注意事项等。

8.7.4 [REL-04] (P,M,H)应确保每次发布得到批准。

软件发布前，应充分分析软件发布可能的影响，并得到正式的批准。

8.7.5 [REL-05] (P,M,H)应从约定的渠道获取待发布内容。

应与客户建立唯一的发布版本获取渠道，确保获取版本的一致性。

8.7.6 [REL-06] (P,M,H)应确保每个发布包及时交付给客户。

项目应及时确认每个发布得到客户的接收，且获取版本的一致性。

9 能力评估要求

9.1 能力评估概述

软件能力评估根据被评组织最高的软件重要性等级确定。被评组织可以选择重要性等级要求及以上的要求在组织内部实施，但不可以对重要性等级及以下的内容进行删减，除非组织提供合理的适用性声明，说明相关要求在组织内部不适用，并得到客户或第三方评估机构的认可。

组织可以通过实施和评估相关软件开发要求来声称组织具备开发某个重要性等级软件的能力，但不得声称其软件达到了某个软件重要性等级。如果需要声称软件达到重要性等级为M和H等级，需对其软件产品进行评估。

表4 软件重要性等级和能力评估关系

软件重要性等级	过程能力评估	软件产品评估
P	可以	不适用
M	可以	可以
H	可以	可以

9.2 过程能力评估要求

过程能力评估可以评估组织是否达到P级、M级、或H级软件重要性等级的过程能力。

组织可以根据自身软件开发情况，声称除“组织级软件管理”过程组之外的过程的裁剪，过程的裁剪应提供《适用性声明》并得到组织管理层的确认。

《适用性声明》应得到评估方的确认。适用的过程以及相应软件重要性等级的要求应被评估，被裁剪的过程应在评估报告和相应的证明性材料中标明。

过程能力评估方法见《汽车软件开发能力要求 之 过程能力评估指南》。

9.3 软件产品评估要求

本文件支持对M级和H级软件产品的评估。组织要对软件产品进行评估，除了要达到相应级别的过程能力要求外，还应在软件开发过程和技术上，满足M和H相关的要求。

组织应首先基于软件重要性等级分析软件失效风险，建立软件重要性等级目标，并将该目标的实现分解到软件开发全声明周期活动中。软件产品评估在确认软件重要性等级目标分析合理性后，将对软件开发过程进行评估。对于H级别的软件，组织应提供CNAS（中国合格评定国家认可委员会）认可的具备相应检测能力的实验室提供的软件检测报告。

本文件支持功能安全软件和网络安全软件的开发。若组织软件满足功能安全要求，可以参照ISO26262 Part6部分相应的完整性等级开发。若组织软件满足网络安全要求，组织可以参照ISO/SAE 21434第7、9、10、11和15章内容实施。

参考文献

[1] ISO/IEC 33002:2015 信息技术-过程评估-实施过程评估的要求
[2] ISO/IEC 12207:2017 系统和软件工程-软件生命周期过程
[3] ISO/IEC TR 19759-2015 软件工程-软件工程知识体系指南
[4] CMMI Model v3.0，CMMI 研究院，2023
[5] Automotive SPICE v4.0，VDA QMC，2023
[6] ISO 26262:2018 道路车辆 功能安全
[7] ISO/SAE 21434:2021 道路车辆 网络安全工程
[8] ISO 21448:2022 道路车辆 预期功能安全