

T/CRSS

重庆市机器人学会团体标准

T/CRSS XXXX—XXXX

服务机器人 软件试验方法

Service robots-Software testing method

（征求意见稿）

在提交反馈意见时，请将您知道的相关专利连同支持性文件一并附上。

XXXX—XX—XX 发布

XXXX—XX—XX 实施

重庆市机器人学会 发布

目 次

前言 III

1 范围 4

2 规范性引用文件 4

3 术语和定义 4

4 服务机器人软件系统 4

5 软件等级 5

 5.1 软件等级的定义 5

 5.2 软件等级的确定 5

6 一般要求 5

 6.1 测试目的 5

 6.2 测试原则 5

 6.3 测试级别 5

 6.4 软件等级与测试级别的关系 6

 6.5 测试类型 6

 6.6 测试方法 6

 6.7 测试过程 6

 6.8 测试环境 6

 6.9 软件问题分级分类及处理 7

 6.10 测试文档 8

7 测试过程 8

 7.1 测试需求分析与策划 8

 7.2 测试设计与实现 9

 7.3 测试执行 9

 7.4 测试总结 10

8 测试技术要求 10

 8.1 单元测试 10

 8.2 集成测试 11

 8.3 系统测试 12

 8.4 回归测试 13

附录 A（规范性） 测试类型技术要求 14

 A.1 文档类测试 14

 A.2 代码类测试 14

 A.3 数据类测试 14

 A.4 功能类测试 15

 A.5 性能类测试 15

 A.6 接口类测试 16

 A.7 专项测试 17

附录 B（资料性） 软件测试文档模板..... 19

 B.1 软件测试用例编写模板 19

 B.2 软件测试问题单模板 20

 B.3 测评计划章节内容 21

附录 C（资料性）服务机器人软件级别策略参考..... 24

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。

请注意本文件的某些内容可能涉及专利。本文件的发布机构不承担识别专利的责任。

本文件由重庆市机器人学会提出并归口。

本文件起草单位：

本文件主要起草人：

服务机器人 软件试验方法

1 范围

本文件规定了服务机器人软件试验方法的术语和定义、技术要求、试验条件和试验方法。主要的技术内容有测试对象的安全性等级要求、测试目的、测试内容、测试级别、测试方法、测试过程、测试用例、测试管理、文档编写和测试工具，从而实现机器人软件测试与评估。

本文件适用于服务机器人软件测试与评估。

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。

GB/T 12643—2013 机器人与机器人装备 词汇

GB/T 25000.10—2016 系统与软件工程 系统与软件质量要求和评价(SQuaRE) 第10部分：系统与软件质量模型

GB/T 25000.51—2016 系统与软件工程 系统与软件质量要求和评价(SQuaRE) 第51部分：就绪可用软件产品(RUSP)的质量要求和测试细则

GB/T 38260—2019 服务机器人功能安全评估

GB/T 38634—2020 系统与软件工程 软件测试

3 术语和定义

下列术语和定义适用于本文件。

3.1

机器人 robot

具备一定程度的自主能力，可在其环境内运动以执行预期的任务的执行机构。

3.2

服务机器人 service robot

除工业自动化应用外，能为人类或设备完成有用任务的机器人。

注：工业自动化应用包括(但不限于)制造、检验、包装和装配。

注：用于生产线的关节机器人是工业机器人，而类似的关节机器人用于供餐的就是服务机器人。

[来源：GB/T12643—2013, 2.10]

3.3

安全相关软件 safety-related software

在服务机器人安全相关系统中，用于实现SRCF的软件。

[来源：GB/T38260—2019, 3.1.20]

4 服务机器人软件系统

4.1 机器人的软件设计应与硬件资源相适应。服务机器人软件系统应包括运行环境、驱动程序、应用软件，还应具备系统运行日志、数据共享接口及相应的异常检测和处理机制，包含故障自检和故障修复程序等。服务机器人软件架构通常包含但不限于以下四层。

4.2 人机交互及传感器网络管理层（必备）：用户通过语音、触摸屏、按键等向机器人发送指令；对传感器网络进行管理。

4.3 中间服务层（必备）：将机器人低、中、高级功能封装成相应的服务，导航规划、任务执行、操作动作等。

- 4.4 操作器及传感器驱动软件接口层（必备）：系统中各种传感器、操作器等硬件设备的软件接口。
- 4.5 中台管理层（选配）：采集、分析单个服务机器人终端数据，也可以支持远程控制。
- 4.6 服务机器人系统按功能模块分可以包含但不限于人机交互系统、业务系统、控制系统、图像识别系统、运动系统、导航定位系统、环境认知系统、大数据决策系统等。

5 软件等级

5.1 软件等级的定义

服务机器人软件等级是在系统安全评估过程中确定的，它是根据软件对潜在的失效状态的贡献率来划分的。软件等级随着失效状态类别的变化而变化。

服务机器人软件等级的定义如下：

- a) A 级：其异常状态将会导致或引起系统功能的失效并给服务机器人、使用者及周围环境带来灾难性的失效状态的软件，如安全相关软件；
- b) B 级：其异常状态将会导致或引起系统功能的失效并给服务机器人带来严重的失效状态的软件；
- c) C 级：其异常状态不会降低服务机器人的安全性和可操作性。

5.2 软件等级的确定

5.2.1 如果服务机器人软件部分的异常状态引起多个失效状态，软件部件中最严重的失效状态类别决定了软件部件的软件等级。

5.2.2 系统功能可以分配到一个或多个已划分的软件部件中，并行实施是用多个软件部件来实现一个系统功能。这样，只有多个部件的异常状态才能产生一个失效状态。对并行架构，至少有一个软件部件具有与系统功能最严重的失效状态类别相应的软件等级。

5.2.3 一个系统功能亦可用多个软件部件来串行实施。这样，任何部件的异常状态都能产生失效状态。在这种情况下，软件部件应具有与系统功能的最严重的失效状态类别相应的软件等级。

6 一般要求

6.1 测试目的

服务机器人软件的测试目的是：

- a) 验证软件是否满足服务机器人系统规格说明、系统设计说明、软件需求规格说明、软件设计说明等规定的软件功能、性能、接口、安全及其他特性要求；
- b) 通过软件测试，尽早的发现软件中的缺陷，通过修正各种缺陷提高软件质量，规避软件发布后由于潜在的软件缺陷造成的失效所带来的风险。；
- c) 为服务机器人软件产品确认、验收以及软件质量评估提供依据。

6.2 测试原则

服务机器人软件的测试原则主要包括：

- a) 充分性。软件测试应全面覆盖软件功能、性能、接口等需求及其他特性要求；
- b) 客观性。软件测试应采取客观公正的态度，测试过程、人员应保持相对的独立性；
- c) 追溯性。所有的测试点都可追溯到需求/用户。

6.3 测试级别

如下测试级别的详细描述见本文件第8章测试技术要求：

- a) 单元测试；
- b) 集成测试；
- c) 系统测试。

注1：回归测试可出现在上述每个测试类别中，并贯穿于整个软件全生命周期，单独分类进行描述。

注2：不同类型服务机器人应完成的测试级别参考附录C。

6.4 软件等级与测试级别的关系

针对服务机器人软件特点，A级软件必须完成单元测试、集成测试、系统测试。B级软件至少完成集成测试与系统测试。C级软件至少完成系统测试。

6.5 测试类型

应根据软件测试目的、要求及软件等级等特点，选取适当的测试类型，测试类型技术要求见附录A：

- a) 文档类测试。文档审查；
- b) 代码类测试。代码审查、静态分析；
- c) 数据类测试。数据处理测试；
- d) 功能类测试。功能测试、边界测试、可恢复性测试；
- e) 性能类测试。性能测试、余量测试、强度测试、容量测试；
- f) 接口类测试。接口测试、人机交互界面测试；
- g) 专项类测试。安全性测试、可靠性测试、兼容性测试。

6.6 测试方法

从是否实际执行程序的角度，测试方法可分为动态测试和静态测试。从是否考虑程序内部结构和内部特性的角度，测试方法可分为白盒测试、黑盒测试和灰盒测试。

测试方法的选用及要求包括：

- a) 单元测试一般采用白盒测试方法，辅助以黑盒测试方法；集成测试和系统测试一般采用黑盒测试方法，辅助以白盒测试方法；
- b) 测试用例设计时，应依据测试类型的特点，使用恰当的测试方法对测试用例进行分析与设计，确保测试用例的充分性、典型性、准确性和合理性：
 - 1) 充分性。如使用功能分解法对功能测试点进行分解，分解粒度应达到恰当的细度；
 - 2) 典型性。如使用等价类划分法设计具有代表性的测试用例，避免同类测试用例无实质意义的机械累加；
 - 3) 准确性。如使用边界值分析法确定边界条件，分析边界条件所对应的输入数据；
 - 4) 合理性。如使用判定表法生成测试用例，保证测试用例验证需求规格的合理性。
- c) 测试方法的使用结果应在测试用例中进行详细说明，基于某测试方法生成的测试用例集，应进行统一归类说明；
- d) 基于可量化度量的测试方法生成的测试用例，其测试结果应进行量化评价。

6.7 测试过程

服务机器人软件测试过程主要包括以下四个步骤，回归测试应在软件更改情况及影响域分析的基础上视情况执行 a)~c)。

- a) 测试需求分析与策划。确定需要测试的内容、测试的充分性要求，提出测试的基本方法；确定测试的资源、技术需求；分析测试风险，制定测试计划；进行测试计划评审；
- b) 测试设计与实现。设计和选取测试用例；获取并验证测试数据；根据测试资源、风险等约束条件，确定测试用例执行顺序，编制测试用例；获取测试资源，开发或选用测试工具；建立并校准测试环境；进行测试用例评审；
- c) 测试执行。执行测试用例，获取并记录测试结果数据；分析测试过程的正常或异常终止情况，视情补充或停止测试；对测试过程中发现的问题进行分析确认并填写问题报告单；
- d) 测试总结。汇总测试数据，总结测试工作，评估测试结果，描述测试状态；编制测试报告，进行测试总结评审。

6.8 测试环境

服务机器人软件测试环境通常包括被测软件运行所需的软件、硬件、数据、工具及接近服务机器人真实工作外部环境，如家居环境、商场环境、酒店环境、医院环境等。场景化测试是服务机器人测试的重点：

- a) 不同的测试级别一般使用不同的测试环境，应保证与软件实际运行环境的一致性 or 相容性。通常，单元测试可在仿真环境下进行，集成测试、系统测试应在至少接近服务机器人真实工作环境下进行；
- b) 应采取措施保证测试的软件环境没有被病毒感染；
- c) 测试环境应尽可能与开发环境分离；
- d) 测试环境应达到系统或软件对安全性、保密性的需求；
- e) 测试环境应考虑被测软件对设备、网络设施等硬件环境的适应能力，以及对系统软件、其他并行的应用软件等软件环境的适应能力；
- f) 当测试环境与实际环境存在差异时，应进行差异性分析，说明在该环境下测试结果的有效性；
- g) 应根据测试要求选用测试工具，包括采购商用测试工具和自行开发测试工具；
- h) 对测试结果有重要影响的测试数据、有指标要求的测试工具，在投入使用前应采用适当的方法对其是否符合测试要求进行校核、确认；
- i) 应对测试工具实施管理，包括版本控制、升级以及技术支持；
- j) 在测试执行前应对测试环境进行校核，在测试过程中应对测试环境进行管理和维护。

6.9 软件问题分级分类及处理

6.9.1 软件问题后果

软件中的错误可能导致故障的出现，产生服务机器人失效状态。

6.9.2 软件问题分类

软件测试过程中发现的问题可分为：

- a) 需求问题。产品定义或需求问题；
- b) 设计问题。系统设计或软件设计问题；
- c) 文档问题。文档描述问题；
- d) 编码问题。代码实现问题；
- e) 数据问题。数据规格及内容问题；
- f) 其他问题。上述问题之外的问题。

6.9.3 问题等级

软件问题分为灾难、严重、一般、改进建议四个等级：

- a) 灾难问题。将会造成服务机器人失去控制并对周围环境造成破坏或对人造成伤害。
 - 1) 导致系统死机、崩溃或异常退出；
 - 2) 主要功能未实现或实现错误；
 - 3) 造成人员、设备、环境等重大损失；
 - 4) 重要数据丢失，且很难恢复。
- b) 严重问题。将会降低服务机器人功能水平大幅下降。
 - 1) 没有完整实现软件需求，对主要功能性能等有较大影响；
 - 2) 没有正确实现软件需求，对主要功能性能等有较大影响；
 - 3) 造成环境等严重损失；
 - 4) 重要数据丢失，但能以某种方式恢复；
 - 5) 软件文档对主要功能、性能描述缺失或错误。
- c) 一般问题。不会降低服务机器人的安全性和可操作性，软件问题对软件功能性能有较小影响。
 - 1) 没有完整实现软件需求，对软件主要功能性能影响较小，或对一般功能性能造成影响；
 - 2) 没有正确实现软件需求，对软件主要功能性能影响较小，或对一般功能性能造成影响；
 - 3) 软件操作与软件使用说明不符；
 - 4) 软件文档存在准确性、一致性、错别字等影响较小的问题。
- d) 改进建议。测试过程中发现的对软件功能有轻微影响的问题可提出改进建议。

6.9.4 问题处理

在软件测试过程中应如实记录测试过程、原始数据、结果及发现的故障现象,填写软件问题报告单:

- a) 测试人员应与开发人员共同确认发现的软件问题;
- b) 开发人员应对问题进行定位,开展原因分析,提出修改措施,说明修改对软件的影响,如不修改,应说明理由及其影响,在回归测试前提交给测试方;
- c) 对测试中有争议的问题,应组织利益相关方及领域专家共同确认。

6.10 测试文档

软件测试过程中的文档主要包括(参见附录B):

- a) 软件测试计划;
- b) 软件测试用例及测试记录;
- c) 软件问题单;
- d) 软件测试报告;
- e) 其他管理文档和记录,如:评审、质量保证、项目跟踪以及配置管理等记录和报告。

7 测试过程

7.1 测试需求分析与策划

7.1.1 过程输入

开展软件测试需求分析与策划活动的输入应包括:

- a) 软件测试任务书、合同或其他等效文件;
- b) 软件开发文档,例如,系统需求说明、接口需求说明、系统设计说明、接口设计说明、软件研制任务书、软件需求规格说明、软件设计说明、软件用户手册、数据库设计说明等;
- c) 软件更改及影响分析报告(必要时);
- d) 软件源程序;
- e) 软件运行资源。

7.1.2 过程输出

软件测试需求分析与策划阶段输出的主要产品为软件测试计划。

7.1.3 过程要求

软件测试需求分析与策划要求一般包括:

- a) 测试需求分析。根据输入信息分析测试需求并确定以下内容:
 - 1) 确定测试级别;
 - 2) 确定测试充分性要求。根据被测软件的重要性、测试目标和约束条件,确定测试范围及每一范围所要求的覆盖程度;
 - 3) 确定测试需求。分析被测软件的功能、性能、接口、数据结构、设计约束等,包括隐含需求及特殊需求;
 - 4) 根据测试需求确定测试类型及其测试点;
 - 5) 分析并确定测试环境需求。
- b) 测试策划。根据测试需求分析结果策划测试活动,确定以下内容:
 - 1) 确定测试资源要求,包括人员、设备、设施等;
 - 2) 确定测试策略、技术和方法,包括测试环境搭建策略、集成测试策略、采用的标准或非标准测试方法以及测试数据生成和验证方法、测试数据注入方法、测试结果捕获方法及分析方法、使用的测试工具、动静态测试先后顺序等;
 - 3) 确定测试结束条件;
 - 4) 确定被测软件的评价准则和方法;
 - 5) 根据任务要求、资源、风险、测试充分性等因素确定测试进度;
 - 6) 分析测试风险及应对措施,例如,技术风险、人员风险、资源风险和进度风险等;
 - 7) 确定测试点跟踪与控制、配置管理和质量保证等要求。

- c) 编写软件测试计划；
- d) 软件测试计划应受到变更控制和版本控制。

7.2 测试设计与实现

7.2.1 过程输入

开展软件测试设计与实现的输入应包括但不限于以下内容：

- a) 软件测试计划；
- b) 软件开发文档，例如，系统规格说明、接口需求规格说明、软件需求规格说明、系统设计说明、接口设计说明、数据库设计说明、软件设计说明、软件用户手册等；
- c) 软件源程序、可执行文件及软件运行所依赖的数据；
- d) 软件运行的硬件环境；
- e) 系统运行的场景。

7.2.2 过程输出

软件测试设计与实现阶段输出应包括：

- a) 软件测试用例；
- b) 软件测试数据；
- c) 自行开发的测试程序、软硬件工具；
- d) 仿真或真实的服务机器人运行场景。

7.2.3 过程要求

软件测试设计与实现的要求一般包括：

- a) 对所有测试点或测试子项设计测试用例，并进行标识；
- b) 根据测试资源、风险等约束条件确定测试用例/典型实例执行顺序；
- c) 针对测试输入要求，设计测试数据，准备和验证测试数据；
- d) 准备测试资源，例如，测试工具、搭建测试环境所必须的软硬件资源，必要时，开发测试执行所需测试程序、软硬件工具；
- e) 建立和校核测试环境，记录校核结果，说明测试环境的偏差及对测试结果的影响；
- f) 编写软件测试用例，确定软件测试用例与软件测试计划的追踪关系；
- g) 应对该阶段工作产品进行评审；
- h) 软件测试用例、测试程序应受到变更控制和版本控制。

7.3 测试执行

7.3.1 过程输入

开展软件测试执行活动的输入应至少包括：

- a) 通过评审的软件测试计划、软件测试用例；
- b) 已建立并通过验证的测试环境、测试数据、测试工具等；
- c) 被测软件已纳入配置管理。

7.3.2 过程输出

软件测试实施阶段输出应包括：

- a) 软件测试记录；
- b) 软件问题报告。

7.3.3 过程要求

软件测试执行的要求一般包括：

- a) 静态测试一般先于动态测试执行；
- b) 文档审查、代码审查应按照审查单要求逐项进行，记录审查情况、存在的问题等信息；

- c) 应按照软件测试用例的内容和要求执行测试用例，如实、具体、完整地记录测试输入数据、测试结果，当测试结果有量值要求时，应准确记录实际的量值；
- d) 根据每个测试用例的期望测试结果、实际测试结果和评估准则，分析并判定测试结果；
- e) 在执行测试的过程中，可根据测试的进展情况补充测试用例，必要时变更软件测试计划；
- f) 当所有的测试用例执行完毕，应对测试的充分性进行分析。如果发现测试工作不足，或测试未达到预期要求时，可增加新的测试用例或测试数据，直到达到充分性要求；
- g) 原始记录应有签署，并受到严格管理；
- h) 汇总测试中有异议的问题，组织问题确认评审。

7.4 测试总结

7.4.1 过程输入

开展软件测试总结的输入应包括：

- a) 软件测试任务书、合同或其他等效文件；
- b) 被测软件相关文档、代码和数据；
- c) 测试文件，包括软件测试计划、软件测试用例及测试记录、软件问题报告、软件回归测试方案（如需要）。

7.4.2 过程输出

软件测试总结阶段输出为软件测试报告。

7.4.3 过程要求

软件测试总结的要求一般包括：

- a) 分析和评价测试工作，一般包括：
 - 1) 总结软件测试计划和软件测试用例的变化情况及其原因；
 - 2) 分析测试工作完成情况，包括回归测试；
 - 3) 分析测试环境与软件实际运行环境之间的差异及其对测试结果的影响；
 - 4) 对测试异常终止情况，分析未能被测试活动充分覆盖的范围及其理由。
- b) 分析和评价被测软件，一般包括：
 - 1) 说明被测软件对研制任务书等文档规定的软件功能、性能、接口及质量特性等要求的满足情况；
 - 2) 统计并分析所发现的软件问题，对遗留的软件问题说明不能解决的理由，给出其可能给软件 and 系统带来的影响，可能时，推荐纠正方案或方法；
 - 3) 分析软件设计、代码中可能存在的缺陷和风险；
 - 4) 根据测试结果评估被测软件，给出评估意见和改进建议。
- c) 分析测试产生的数据和文档，积累测试资产，一般包括：典型软件问题、典型用例、测试脚本、管理数据（如生产率、工作量、进度等）；
- d) 编制软件测试报告；
- e) 应进行测试总结评审；
- f) 软件测试报告应受到变更控制和版本控制。

8 测试技术要求

8.1 单元测试

8.1.1 测试对象

单元测试的对象是具有输入/输出、完成特定功能、可被调用使用的最小代码集合的软件单元。在编程语言中，通常将一个函数、一个模块、一个过程、一个子程序视为一个软件单元。

8.1.2 测试目的

验证软件单元是否实现了软件设计规定的功能、性能、接口和其他设计约束等要求，发现单元内可能存在的错误，并保证代码质量。

8.1.3 开始条件

单元测试进入条件如下：

- a) 软件单元代码无错误地通过编译；
- b) 具备满足要求的测试环境及测试工具。

8.1.4 技术要求

具体要求如下：

- a) 单元测试应列表说明被测单元的清单，对单元的剪裁应说明理由，关键单元、重要单元不允许被剪裁；
- b) 单元测试的直接依据应是详细设计文档（软件设计说明中的详细设计部分），被测单元清单中应说明文档依据的索引；
- c) 应采用静态测试和动态白盒测试的测试方法开展单元测试；
- d) 一般应在动态测试前开展静态测试，静态测试发现的问题修改后再进行动态测试；
- e) 在动态测试中，应设计测试用例逐项验证软件单元的功能、性能、接口等设计要求；
- f) 测试用例的输入应覆盖单元接口输入变量的有效值、无效值和边界值；
- g) 单元测试覆盖率要求如下：
所有单元的语句覆盖率和分支覆盖率应达到80%及以上。
- h) 对于覆盖率未达到指标要求的单元，应说明原因，并通过代码审查进行辅助验证。

8.1.5 环境要求

要求如下：

- a) 应建立单元测试环境，配备软件单元测试工具；
- b) 单元测试环境可以是仿真环境、模拟环境、开发环境（推荐）；
- c) 单元测试环境应支持驱动模块和桩模块的编写与加载，并与测试用例一起进行有效管理。

8.2 集成测试

8.2.1 测试对象

- a) 任意一个软件单元及与其接口相连的其他软/硬件集成得到的局部系统及其集成过程。
- b) 任意一个组装得到的软件系统。

8.2.2 测试目的

软件集成测试的目的是检验软件单元之间、软件单元和已集成的软件系统之间的接口关系，并验证已集成软件系统是否符合设计要求。

8.2.3 开始条件

集成测试进入条件如下：

- a) 软件已纳入软件配置管理，所涉硬件技术状态受控；
- b) 具备与被测软件源代码版本对应的文档；
- c) 具备满足要求的测试环境。

8.2.4 技术要求

软件集成测试一般应符合以下技术要求：

- a) 应采用适合的集成测试策略，使系统中所有的软件和硬件都被集成和测试；
- b) 应对已集成软件进行必要的静态测试，并先于动态测试进行；
- c) 软件要求的每个特性应被至少一个正常的测试用例和一个被认可的异常测试用例覆盖；
- d) 测试用例的输入应至少包括有效等价类值、无效等价类值和边界数据值；

- e) 应测试运行条件（如数据结构、输入/输出通道容量、内存空间、调用频率等）在边界状态下，进而在人为设定的状态下，软件的功能和性能；
- f) 应验证局部系统内外接口的匹配性、协调性、一致性，具体包括：
 - 1) 集成后的软件子系统之间、软硬件之间交互接口数据及其格式；
 - 2) 局部系统的输出数据及其格式；
 - 3) 在任意外部输入情况下，局部系统从外部接口采集和发送数据的能力，包括对正常数据及状态的处理，对接口错误、数据错误、错误的识别及处理。
- g) 应验证集成后的软硬件工作时序之间的匹配性、协调性、一致性；
- h) 应验证局部系统对硬件资源使用及硬件资源配置之间的匹配性、协调性、一致性、合理性和资源余量。
- i) 对不同的实际问题应外加相应的专门测试，比如安全测试、兼容性测试等。

8.2.5 测试环境

集成测试环境要求如下：

- a) 集成测试环境推荐使用软件真实运行环境和真实外部硬件环境；
- b) 若选择仿真或模拟测试环境，应进行环境等效性分析；
- c) 应配备必要的软件测试工具、监测设备、数据分析软件等。

8.3 系统测试

8.3.1 测试对象

系统测试的对象是完整的、集成的服务机器人软硬件系统。

8.3.2 测试目的

系统测试的目的是在真实系统工作环境下检验完整的服务机器人软硬件系统的功能、性能、接口、安全性、可靠性、易用性等各项要求。

8.3.3 开始条件

系统测试进入条件如下：

- a) 软件已通过集成测试；
- b) 被测软件已纳入软件配置管理，所涉硬件技术状态受控；
- c) 具备软件系统测试要求的环境；
- d) 具备与被测软件源代码版本对应的文档。

8.3.4 技术要求

软件系统测试一般应符合以下技术要求：

- a) 开展测试需求分析，列表说明系统的测试点，并说明与需求点的对应关系。通常一个需求点应被若干个测试用例所覆盖，一般应被正常测试用例和异常测试用例所覆盖；
- b) 应采用文档审查和动态测试的测试方法开展系统测试，一般采用的是动态黑盒测试方法。
- c) 依据系统的任务剖面，从运行场景出发进行情景想定分析，开展系统任务想定设计；
- d) 应在动态测试前开展文档审查，文档审查应包含系统的所有相关文档，例如通讯协议、数据处理算法等，在文档审查问题得到有效处理后再进行动态测试；
- e) 测试用例的输入一般应被有效值、无效值和边界值所覆盖；
- f) 软件之间及软件与硬件之间的所有接口应进行测试用例设计；
- g) 建立系统测试环境。依据系统的特点及具体情况，系统测试环境可以是半实物仿真环境、全实物实装环境等，系统测试环境应能支持运行方案说明中描述的运行场景，支持系统任务过程所需情景想定的配置，支持系统任务过程测试用例的加载、执行、过程数据采集等，评估测试环境对测试结果的影响，分析系统测试环境的局限性，确认系统测试环境的有效性；
- h) 动态测试的测试类型选择要求：
 - 1) 至少应包括：功能测试、性能测试、接口测试、边界测试；

- 2) 关键重要系统的测试类型应增加：安全性测试、余量测试、强度测试。在实装系统上开展的安全性测试，应在安全关键部件模拟器的配合下进行测试；
- 3) 测试类型应结合软件的特点进行选择，如，具有人机交互界面的系统应进行人机交互界面的测试，具有双机热备份或冷备份功能的系统应进行恢复性测试，对可异步并发操作同一共享数据源的相关软件应进行互操作性测试等；
- i) 基于运行方案说明中的运行场景，将系统规格说明中的系统能力需求组合为系统的任务需求，逐一验证系统的任务运行能力。

8.3.5 测试环境

系统测试环境要求如下：

- a) 推荐使用全实物实装环境。若选择全数字仿真环境或半实物仿真环境，应进行环境等效性分析；
- b) 应配备必要的软件测试工具、监测设备、数据分析软件等。

8.4 回归测试

8.4.1 测试对象

更改后的软件，包括更改所影响到的软件单元、软件子系统、软件系统，还应包括因软件更改涉及到的集成过程。

8.4.2 测试目的

对更改后的软件重新进行测试，以确认更改正确且更改未引入新的软件问题，即更改未影响软件原有的、正确的功能、性能和其他规定的要求。

8.4.3 开始条件

回归测试进入条件如下：

- a) 被测软件已纳入配置管理；
- b) 具备软件开发文档、代码、数据、软件问题处理单（或软件更改及影响分析报告）等；
- c) 具备相关的测试文档及资源；
- d) 具备相应级别测试的进入条件。

8.4.4 技术要求

具体要求如下：

- a) 应统计软件修改的代码更改量，包括：
 - 1) 相同行。更改前与更改后完全相同的代码行；
 - 2) 修改行。更改前与更改后部分相同的代码行；
 - 3) 增加行。更改前没有而更改后有的代码行；
 - 4) 删除行。更改前有而更改后没有的代码行；
 - 5) 更改行。修改行、增加行、删除行之和；
 - 6) 更改率。更改行/(更改行+相同行)*100%。
- b) 当软件完成测试后，后续软件又发生了变化，如软件更改率大于 20%，则应按全新软件重新测试；
- c) 应依据软件更改影响分析结果确定回归测试范围，选用或修改已有的测试用例，或新增测试用例，并对测试用例使用情况进行分类统计：
 - 1) 沿用测试用例；
 - 2) 修改测试用例，即测试用例的名称、标识未改，但内容略有修改；
 - 3) 新增测试用例。
- d) 应论证或证明测试用例的执行覆盖了全部修改内容；
- e) 回归测试的技术要求应符合原测试级别的技术要求。

附 录 A
(规范性)
测试类型技术要求

A.1 文档类测试

A.1.1 文档审查

文档审查是开展的针对软件相关文档的审查。文档审查的具体要求如下：

- a) 审查软件文档种类是否齐套；
- b) 审查软件文档内容是否完整；
- c) 审查软件文档描述是否准确；
- d) 审查软件文档格式是否规范；
- e) 审查软件文档是否文文一致、文实相符；
- f) 编制审查所用文档检查单并通过评审。

A.2 代码类测试

A.2.1 代码审查

代码审查是依据相关标准及软件文档开展的针对软件程序代码的审查。代码审查的具体要求如下：

- a) 以人工阅读方式对代码进行审查，可以借助工具辅助完成分析。
- b) 代码审查包含编程准则检查、代码流程审查、软件结构审查、需求实现审查四个审查类型，测试需求分析中应确定需要开展的审查类型。
- c) 编程准则检查：依据编程准则的要求，对程序的编码进行编程准则的符合性检查。编程准则检查应依据语言特点，确定编程准则的检查标准并通过评审，使用专业工具扫描出的警告信息应经过人工核实确认。
- d) 代码流程审查：审查程序代码的条件判别、控制流程、数据处理等是否满足设计要求。
- e) 软件结构审查：依据设计文档，审查程序代码的结构设计，包括程序结构设计和数据结构设计，
- f) 在程序设计层发现问题。需求实现审查：依据需求文档及其他相关资料，审查程序代码的需求层的功能实现，审查中应形成所有变量物理含义及取值含义的变量字典，依据数学模型、逻辑模型、时序模型、处理模型等和变量字典审查程序代码的处理流程，发现需求实现的问题。
- g) 代码审查的软件单元应列表汇总，并针对软件单元说明开展的审查类型。
- h) 应根据软件的特点及审查内容，确定审查所用的代码审查单并通过评审。

A.2.2 静态分析

静态分析是可借助专业工具对程序代码特性进行机械性和程序化的专项分析，静态分析的内容通常包括程序结构分析、数据结构分析、控制流分析、数据流分析、接口分析、表达式分析、语言使用分析、软件质量指标度量等。

静态分析应对程序代码的质量度量元进行统计与度量。程序质量度量的具体要求如下：

- a) 质量度量元包括：软件的代码行数、有效代码行数、注释行数、模块数、模块代码行数、模块圈复杂度、模块基本复杂度、模块扇入数、模块扇出数等。
- b) 通常指标要求如下：
 - 1) 软件总注释率不小于 20%（注释行数/代码行数*100%）；
 - 2) 模块的平均规模不大于 200 行（模块代码行数之和/模块数）；
 - 3) 模块的平均圈复杂度不大于 10（模块圈复杂度之和/模块数）；
 - 4) 模块的平均扇出数不大于 7（模块扇出数之和/模块数）。
- c) 对圈复杂度、规模行数、扇出数不满足指标要求的模块，应进行专项代码审查。
- d) 基于指标要求并结合其他度量结果，给出软件编码质量的评价。

A.3 数据类测试

A.3.1 数据处理测试

数据处理测试是对完成专门数据处理功能所进行的测试。数据处理测试的具体要求如下：

- a) 应对数据文件存取、数据库操作、数据采集、数据融合、数据转换、数据解析等专门数据处理功能进行测试；
- b) 应对剔除坏数据、数据滤波、数据容错等数据特殊处理功能进行测试；
- c) 应针对数据读取/写入过程中的容错、保护、超时等进行测试；
- d) 应对大数据处理算法、模型的实现正确性进行测试。

A.4 功能类测试

A.4.1 功能测试

功能测试是对软件的功能需求逐项进行的测试，以验证其功能是否满足要求。功能测试的具体要求如下：

- a) 应对软件功能进行分析，通过等价类、边界值、判定表、因果图、猜错法等分析方法确定软件功能的输入；
- b) 输入等价类应包括正常等价类和异常等价类；
- c) 输入边界值应包括合法边界值和非法边界值；
- d) 确定功能的输出及预期的输出结果和判定条件；
- e) 应用真实数据测试超负荷、饱和及其他最坏情况等极端条件；
- f) 应对功能控制流程、状态转换、模式切换等的正确性和合理性进行验证；
- g) 在系统测试中，应在任务剖面和业务流程中进行测试；
- h) 建议采用组合测试法、蜕变测试法等方法提高关键功能的测试充分性。

A.4.2 边界测试

边界测试是对软件处在边界或端点情况下运行状态的测试。边界测试的具体要求如下：

- a) 应对输入域或输出域的端点或边界点进行测试；
- b) 针对数据结构（如，数组、字符串、堆栈等）进行端点或边界点测试；
- c) 针对状态的转换条件（如，阈值判别、区间判别等）进行端点或边界点测试；
- d) 针对状态的出现概率（如，设备状态、通讯状态等）进行小概率极端情况的测试；
- e) 功能、性能、容量等涉及到的极限情况均视为广义端点或边界点进行测试；
- f) 需要时，应考虑接近边界、超越边界、连续来回穿越边界等各种情况的测试。

A.4.3 恢复性测试

恢复性测试是对有恢复或重置功能的软件的每一类导致恢复或重置的情况逐一进行的测试，以验证其恢复或重置功能。恢复性测试是要证实在克服软硬件故障后，系统能否正常地继续进行工作，且不对系统造成任何损害。

恢复性测试的具体要求如下：

- a) 应对探测错误并通过容错恢复其正常工作的能力进行测试；
- b) 应对自复位或备机切换措施恢复继续工作的能力进行测试；
- c) 应对系统恢复后，依据记录数据恢复故障前运行作业、相关数据和系统状态等能力进行测试；
- d) 应对恢复时间是否满足规定要求进行测试。

A.5 性能类测试

A.5.1 性能测试

性能测试是对软件的性能需求逐项进行的测试，以验证其性能是否满足要求。性能测试的具体要求如下：

- a) 应进行数据精度的测试，如数值计算的精确度等；
- b) 应进行时间精度的测试，如执行时间、响应时间等；
- c) 应进行空间占用的测试，如软件运行所占用的内存空间等；

- d) 应进行处理能力的测试，如功能所处理的数据量等；
- e) 应进行数据传输吞吐量的测试；
- f) 应关注软件并发处理能力的测试；
- g) 在系统测试中，应关注软件性能和硬件性能的集成；
- h) 测试结果应得到具体的量化数值；
- i) 对具有不确定性的数值：
 - 1) 应至少得到 10 组以上的实测值；
 - 2) 应给出最大值、最小值、平均值的统计结果；
 - 3) 对波动性较大的测量值，应统计出实测值的方差。

A.5.2 余量测试

余量测试是对软件是否达到需求要求的余量的测试。余量测试的具体要求如下：

- a) 针对时间约束要求，应测试出实际执行时间相对于时间约束要求的余量；
- b) 针对空间约束要求，应测试出实际占用空间相对于空间约束要求的余量；
- c) 针对处理约束要求，应测试出软件具备的处理能力相对于处理约束要求的余量；
- d) 针对通讯约束要求，应测试出数据传输吞吐量相对于带宽的余量；
- e) 如无明确规定，最少应有 20% 以上的余量。

A.5.3 强度测试

强度测试是检验软件的外部可变性影响条件恶劣到何种程度将导致软件无法正常工作的测试。强度测试的具体要求如下：

- a) 应首先确定软件运行所依赖的外部可变性影响条件；
- b) 控制外部可变性影响条件的范围变化（如，处理的信息量越来越大、通讯的数据量越来越大、监测报警数越来越多），测试出直到软件故障或条件已达极限时的范围极限条件；
- c) 控制外部可变性影响条件的频度变化（如，越来越频繁的外部错误、越来越小的通讯周期、越来越频繁的中断信号），测试出直到软件故障或条件已达极限时的频度极限条件；
- d) 对软件进行业务流程工作状态下的规定的长时间连续不中断运行的测试（并不要求一定运行至出现故障）；
- e) 当软件运行环境资源不能保证时，应在测试中逐步恶化运行环境条件，测试出直到软件故障时的极限运行环境条件；
- f) 对具有降级处理能力的软件，应对降级条件进行极限情况测试。

A.5.4 容量测试

容量测试是检验软件的能力最高能达到什么程度的测试。容量测试一般应测试到在正常情况下软件所具备的最高能力。容量测试的具体要求如下：

- a) 针对具有时间约束要求的功能，应测试出正常工作条件下实际执行时间的最值范围；
- b) 针对具有空间约束要求的功能，应测试出正常工作条件下实际占用空间的最值范围；
- c) 针对通讯接口，应测试出正常工作条件下实际传输时间、传输数据量的最值范围；
- d) 针对软件的处理能力，如处理目标数等，应测试出正常工作条件下处理能力的最值范围。

A.6 接口类测试

A.6.1 接口测试

接口测试是对软件的接口需求逐项进行的测试，以验证其接口是否满足要求。功能测试的具体要求如下：

- a) 应对接口的信息格式是否正确进行测试，如帧格式是否满足要求；
- b) 应对接口的信息内容是否正确进行测试，如内容的解析是否正确；
- c) 应对接口的时间特性是否满足要求进行测试，如传输时间、时序关系等；
- d) 应对外部干扰、丢帧、错帧、误码等异常模式予以容错性验证；
- e) 集成测试和系统测试中，应重点对软件的所有外部接口进行测试；

f) 软硬件系统中应特别关注软硬件接口，应关注信号触发类的接口测试。

A.6.2 人机交互界面测试

人机交互界面测试是对所有人机交互界面提供的操作和显示界面进行的测试，以检验是否满足用户的要求。人机交互界面测试的具体要求如下：

- a) 应依据用户手册或操作手册，逐条验证文实的一致性；
- b) 应对界面显示的符合性、准确性、直观性等进行测试；
- c) 应对操作输入的方便性、健壮性、提示性等进行测试；
- d) 应对人机交互的友好性、导航性、适宜性等进行测试；
- e) 软硬系统中作为软件输入的操作杆、旋钮、开关等均属于操作界面范畴，作为软件输出的警示灯、蜂鸣器等均属于显示界面范畴。

A.7 专项测试

A.7.1 安全性测试

安全性测试是检验软件功能安全性以及信息安全性是否满足要求的测试。安全性测试的具体要求如下：

- a) 应对软件安全性需求中确定的与软件相关的所有故障模式进行逐一测试，验证软件处理故障模式的安全性措施正确并有效。
- b) 应对系统故障后的降级处理能力进行测试。
- c) 软硬件系统中，应进行软硬混合故障模式的测试。
- d) 软件的安全关键单元或部件，必须进行安全性测试。
- e) 对涉及安全性措施的结构、算法、容错、冗余及中断处理等设计，必须进行针对性的测试。
- f) 应对多点组合故障模式进行测试，并结合各种最坏情况的组合进行测试。
- g) 应对双工切换、多机替换等安全性的冗余设计措施进行测试。
- h) 应对可能的异常事件进行测试，包括：
 - 1) 可能的硬件异常，如，外设故障等；
 - 2) 可能的软件异常，如，程序跑飞等；
 - 3) 可能的操作异常，如，操作失误等；
 - 4) 可能的输入异常，如，数据丢帧等；
 - 5) 可能的时序异常，如，控制流程的时间顺序紊乱等。
- i) 应对软件的信息保密与防护能力进行测试：
 - 1) 应对软件使用的身份识别、权限保护能力进行测试；
 - 2) 应对重要数据保护能力（如，抗非法访问能力、加密传输能力等）进行测试；
 - 3) 应对软件 and 系统被恶意篡改或被攻击的防护能力进行测试。

A.7.2 可靠性测试

可靠性测试是在真实的或仿真的环境中，以软件可靠性评估为目的，按照运行剖面和使用的概率分布进行的软件功能测试。软件可靠性测试的具体要求如下：

- a) 测试环境应与典型使用环境的统计特性相一致，必要时使用测试平台；
- b) 从用户视角出发进行情景想定分析，建立软件的使用剖面（任务剖面/业务剖面/运行剖面/操作剖面等）；
- c) 应对软件使用程度进行定量度量，如，使用剖面的概率分布、使用特征的覆盖率等；
- d) 必须保证输入覆盖，应覆盖重要的输入变量值（所有被测输入值域的概率之和必须大于软件可靠性要求）、各种使用功能、相关输入变量可能组合以及不合法输入域等；
- e) 对于可能导致软件运行方式改变的一些边界条件和环境条件，必须进行针对性测试；
- f) 监测软件出现的故障，通常情况下，软件一旦出现故障，应进行软件的纠错性修改，修改后的软件继续进行后续的测试；
- g) 记录并统计软件的故障数据，依据故障数据对软件可靠性指标进行量化评估。

A.7.3 兼容性测试

兼容性测试是检验软件不同版本之间、不同软件产品之间、不同软硬件环境之间兼容程度的测试。兼容性测试的具体要求如下：

- a) 当新版本软件替代旧版本软件时，应进行向下兼容性测试；
- b) 当多个软件版本可以同时使用时，应进行相互兼容性测试；
- c) 当两个软件产品可在同一硬件环境中替换使用时，应进行交错兼容性测试；
- d) 当软件产品可能在不同的硬件设备中使用时，应进行适配兼容性测试；
- e) 当软件产品可能在不同的软件环境中使用时，应进行环境兼容性测试。

附 录 B
(资料性)
软件测试文档模板

B. 1 软件测试用例编写模板

表B. 1 测试用例设计单

被测软件版本			测试用例名称		
测试用例标识			测试用例		
用例设计方法			用例属性		
用例初始化					
前提与约束					
终止条件					
测试过程					
序号	输入及操作说明		期望测试结果	实际测试结果	
评估准则					
设计人员			设计日期		
执行情况		执行结果		问题标识	
测试人员		测试监督员		测试执行日期	

模板说明：

- a) 测试用例名称：测试用例名称应尽量体现该测试用例的核心意图；在同一个测试点目中，测试用例名称必须唯一；
- b) 测试追踪：相应的测试点的标识；
- c) 测试用例：简要描述测试的对象、目的和所采用的测试方法；
- d) 测试用例设计方法：如等价类划分、边界值分析、猜错法、因果图、功能图等；
- e) 用例属性：对于测试软件正常功能和接口的测试用例，填写“正常”测试软件异常功能和接口的测试用例，填写“异常”；
- f) 测试用例初始化：包括软件配置、测试配置（如测试工具、模拟系统等）、参数设置等的初始化要求；
- g) 前提与约束：说明实施测试用例有关的硬件配置情况，例如测试环境中各设备连接情况、某个设备的状态设置情况等；
- h) 终止条件：说明测试用例的测试正常终止和异常终止条件；
- i) 输入及操作说明：记录测试执行的输入，包括：
 - 1) 测试输入项的名称、具体内容（如确定的数值、状态或信号等）、性质（如有效值、无效值、边界值等；
 - 2) 测试输入的来源（如：测试程序生成、磁盘文件读取、网络数据接收、人机交互界面输入等），以及真实的还是模拟的；
 - 3) 测试输入的时间顺序或事件顺序。
- j) 评估准则：对于功能性测试用例，评估准则可填写“与期望结果一致”；对于非功能性测试用例，给出具体评估方法，例如：实际测试结果所需的精确度，允许的实际测试结果与期望结果之间差异的上、下限，时间的最大或最小间隔，时间数目的最大或最小值等。

B.2 软件测试问题单模板

表B.2 软件问题报告单

单编号			项目名称			
问题名称						
问题数目			软件版本			
问题来源						
问题类型	☐ 需求问题	☐ 设计问题	☐ 文档问题	☐ 编码问题	☐ 数据问题	☐ 其它问题
问题等级	☐ 灾难问题		☐ 严重问题		☐ 一般问题	☐ 改进建议
问题描述						

- 5.2 测试内容
- 5.3 测试的有效性、充分性说明
- 5.4 软件问题类型及严重性等级
- 6 测试进度表
- 7 测试终止
 - 7.1 正常终止条件
 - 7.2 异常终止条件
- 8 软件质量评价内容与方法
 - 8.1 文档质量评价
- 9 测评通过标准
- 10 质量保证计划
- 11 配置管理计划
- 12 测试风险分析

B.4 测评报告章节内容

1 范围

1.1 标识

1.2 文档概述

1.3 系统概述

1.4 测试依据

2 引用文件

3 系统需求

3.1 运行环境需求

3.2 测试内容

4 测试资源

4.1 测试环境

4.2 测试场所

4.3 测试数据来源

4.4 人员配置

4.5 使用工具

5 测试过程

5.1 测试资料接收

5.2 测试需求分析及策划

5.3 测试设计

5.4 测试执行

5.5 测试总结

5.6 测试需求覆盖完整性

6 测试结果

6.1 测试级别和类型

6.2 不确定度

6.3 X 轮测试

6.4 第X轮回归测试

6.5 遗留、解释问题

7 测试风险分析

8 测试结论

附录一 测试用例

附录二 问题报告单

附 录 C
(资料性)
服务机器人软件级别策略参考

本附录为可能涉及的服务机器人提供了一份综合列表，推荐了相关服务机器人宜进行的软件测试级别。

表 C.1 服务机器人软件测试宜进行的软件测试级别

序号	功能区域	相关机器人	单元测试	集成测试	系统测试
1	生活配套	扫地机器人、擦窗机器人等。			●
		送餐机器人等。		●	●
		安防机器人等。	●	●	●
2	商业配套	迎宾机器人、礼仪讲解机器人等。			●
		导购机器人、银行业务办理机器人等。		●	●
3	休闲配套	棋牌机器人、竞技娱乐机器人等。			●
4	医疗配套	手术机器人、消毒防疫机器人、远程诊断机器人等。	●	●	●
5	教育配套	教育机器人、智慧教室等。		●	●
6	运动配套	羽毛球机器人、乒乓球机器人等。			●
7	养老配套	个人陪护机器人、助老助残机器人、智能轮椅、智能护理床、助力外骨骼机器人等。	●	●	●
8	物流配套	快递投送机器人、园区巡检机器人等。		●	●
9	健康养生	营养搭配机器人、按摩机器人等。	●	●	●
10	生态农业	采摘机器人等。			●
		消防灭火机器人等。		●	●