МЕЖГОСУДАРСТВЕННЫЙ СОВЕТ ПО СТАНДАРТИЗАЦИИ, МЕТРОЛОГИИ И СЕРТИФИКАЦИИ
(МГС)
INTERSTATE COUNCIL FOR STANDARDIZATION, METROLOGY AND CERTIFICATION
(ISC)

МЕЖГОСУДАРСТВЕННЫЙ СТАНДАРТ

ГОСТ 34332.4— 2021

БЕЗОПАСНОСТЬ ФУНКЦИОНАЛЬНАЯ СИСТЕМ, СВЯЗАННЫХ С БЕЗОПАСНОСТЬЮ ЗДАНИЙ И СООРУЖЕНИЙ

Часть 4

Требования к программному обеспечению

(IEC 61508-3:2010, NEQ)
(IEC 61508-4:2010, NEQ)
(ISO/IEC Guide 51:2014, NEQ)

Издание официальное

Москва
Стандартинформ
2021

Предисловие

Цели, основные принципы и общие правила проведения работ по межгосударственной стандартизации установлены ГОСТ 1.0 «Межгосударственная система стандартизации. Основные положения» и ГОСТ 1.2 «Межгосударственная система стандартизации. Стандарты межгосударственные, правила и рекомендации по межгосударственной стандартизации. Правила разработки, принятия, обновления и отмены»

Сведения о стандарте

1 ПОДГОТОВЛЕН Федеральным государственным унитарным предприятием «Российский научно-технический центр информации по стандартизации, метрологии и оценке соответствия» (ФГУП «СТАНДАРТИНФОРМ») совместно с Международной ассоциацией «Системсервис» (МА «Системсервис»)

2 ВНЕСЕН Федеральным агентством по техническому регулированию и метрологии

3 ПРИНЯТ Межгосударственным советом по стандартизации, метрологии и сертификации (протокол от 30 апреля 2021 г. № 139-П)

За принятие проголосовали:

Краткое наименование страны по МК (ИСО 3166) 004—97	Код страны по МК (ИСО 3166) 004—97	Сокращенное наименование национального органа по стандартизации
Армения	AM	ЗАО «Национальный орган по стандартизации и метрологии» Республики Армения
Беларусь	BY	Госстандарт Республики Беларусь
Киргизия	KG	Кыргызстандарт
Россия	RU	Росстандарт
Таджикистан	TJ	Таджикстандарт
Узбекистан	UZ	Узстандарт

4 Приказом Федерального агентства по техническому регулированию и метрологии от 28 мая 2021 г. № 477-ст межгосударственный стандарт ГОСТ 34332.4—2021 введен в действие в качестве национального стандарта Российской Федерации с 1 января 2022 г.

5 В настоящем стандарте учтены основные нормативные положения следующих международных документов:

IEC 61508-3:2010 «Функциональная безопасность электрических, электронных, программируемых электронных систем, связанных с безопасностью. Часть 3. Требования к программному обеспечению» («Functional safety of electrical/electronic/programmable electronic safety-related systems — Part 3: Requirements for software», NEQ);

IEC 61508-4:2010 «Функциональная безопасность электрических/электронных/программируемых электронных систем, связанных с безопасностью. Часть 4. Термины и сокращения» («Functional safety of electrical/electronic/programmable electronic safety-related systems — Part 4: Definitions and abbreviations», NEQ);

ISO/IEC Guide 51:2014 «Аспекты безопасности. Руководящие указания по их включению в стандарты» («Safety aspects — Guidelines for their inclusion in standards», NEQ)

6 ВВЕДЕН ВПЕРВЫЕ

7 Настоящий стандарт подготовлен на основе применения ГОСТ Р 53195.4—2010[1)]

[1)] Приказом Федерального агентства по техническому регулированию и метрологии от 28 мая 2021 г. № 477-ст ГОСТ Р 53195.4—2010 отменен с 1 января 2022 г.

Информация о введении в действие (прекращении действия) настоящего стандарта и изменений к нему на территории указанных выше государств публикуется в указателях национальных стандартов, издаваемых в этих государствах, а также в сети Интернет на сайтах соответствующих национальных органов по стандартизации.

В случае пересмотра, изменения или отмены настоящего стандарта соответствующая информация будет опубликована на официальном интернет-сайте Межгосударственного совета по стандартизации, метрологии и сертификации в каталоге «Межгосударственные стандарты»

Содержание

1 Область применения 1
2 Нормативные ссылки 3
3 Термины, определения и сокращения 3
4 Общие требования 6
5 Требования к документации 6
6 Требования к управлению программным обеспечением Э/Э/ПЭ СБЗС систем 6
7 Требования к жизненному циклу связанного с безопасностью программного обеспечения 8
8 Оценка функциональной безопасности 34
Приложение А (обязательное) Руководство по выбору методов и средств 36
Приложение Б (обязательное) Подробные таблицы 44
Приложение В (справочное) Свойства стойкости к систематическим отказам пограммного обеспечения 49
Приложение Г (обязательное) Руководство по безопасности для применяемых изделий. Дополнительные требования к элементам программного обеспечения 81
Приложение Д (справочное) Взаимосвязь между ГОСТ 34332.3 и настоящим стандартом 83
Приложение Е (справочное) Методы, не допускающие взаимодействия между элементами программного обеспечения на одном компьютере 85
Приложение Ж (справочное) Руководство по адаптации жизненного цикла систем, управляемых данными 90
Библиография 93

Введение

Современные здания и сооружения (объекты капитального строительства) — это сложные системы, в состав которых входят система строительных конструкций и ряд инженерных систем в разных сочетаниях, в том числе для жизнеобеспечения, реализации технологических процессов, энерго- и ресурсосбережения, обеспечения безопасности, и другие системы. Эти системы взаимодействуют друг с другом, с внешней и внутренней средами и представляют собой единое целое, выполняя определенные функции назначения.

Объекты капитального строительства тесно связаны с особенностями конкретной местности. Рабочие характеристики зданий, сооружений и входящих в них систем могут быть реализованы, проверены и использованы только в том месте, в котором объекты построены и системы установлены.

Безопасность зданий и сооружений обеспечивается применением совокупности мер, мероприятий и средств снижения риска причинения вреда до приемлемого уровня риска и его поддержания в течение периода эксплуатации или использования этих объектов. К средствам снижения риска относятся системы, связанные с безопасностью зданий и сооружений (далее — СБЗС системы). К данным системам относятся системы, неполный перечень которых представлен в ГОСТ 34332.1—2017 (приложение А, раздел А.3). Среди СБЗС систем наиболее распространенными являются системы, содержащие электрические, и/или электронные, и/или программируемые электронные (Э/Э/ПЭ) компоненты. Такие системы, именуемые Э/Э/ПЭ СБЗС системами, в течение многих лет применяют для выполнения функций безопасности. Кроме них и вместе с ними используют системы, основанные на неэлектрических (гидравлических, пневматических) технологиях, а также прочие средства уменьшения риска. Для решения задач безопасности зданий и сооружений во всех больших объемах применяют программируемые электронные СБЗС системы (далее — ПЭ СБЗС системы).

Следующими по важности после характеристик назначения являются характеристики безопасности систем. Наиболее существенной среди характеристик безопасности систем признана их функциональная безопасность.

Настоящий стандарт устанавливает основные требования к функциональной безопасности программного обеспечения (ПО) ПЭ СБЗС систем и к ПО, используемому для разработки таких систем в рамках области применения ГОСТ 34332.1 — ГОСТ Р 34332.3. Настоящий стандарт ориентирован на обеспечение соблюдения требований безопасности и антитеррористической защищенности зданий и сооружений, в том числе объектов транспортных инфраструктур, установленных техническими регламентами Таможенного союза [1]—[3], и на развитие базовых требований этих технических регламентов.

Настоящий стандарт распространяется на ПО Э/Э/ПЭ СБЗС систем и ПО составляющих этих систем, включая сенсоры, исполнительные устройства и интерфейс «человек—машина». Он рассчитан на любой диапазон сложности Э/Э/ПЭ СБЗС систем и ориентирован на комплексное обеспечение безопасности зданий и сооружений гражданского и промышленного строительства, включая объекты инфраструктур промышленности и энергетики, транспорта и связи, гидротехнических и мелиоративных сооружений, включая линейные объекты.

Настоящий стандарт входит в комплекс стандартов с наименованием «Безопасность функциональная систем, связанных с безопасностью зданий и сооружений» и является четвертым стандартом этого комплекса — «Часть 4. Требования к программному обеспечению». Другие стандарты, входящие в этот комплекс:

- часть 1. Основные положения;
- часть 2. Общие требования;
- часть 3. Требования к системам;
- часть 5. Меры по снижению риска, методы оценки;
- часть 6. Прочие средства уменьшения риска, системы мониторинга;
- часть 7. Порядок применения ГОСТ 34332, примеры расчетов.

Структура комплекса ГОСТ 34332 приведена на рисунке 1.

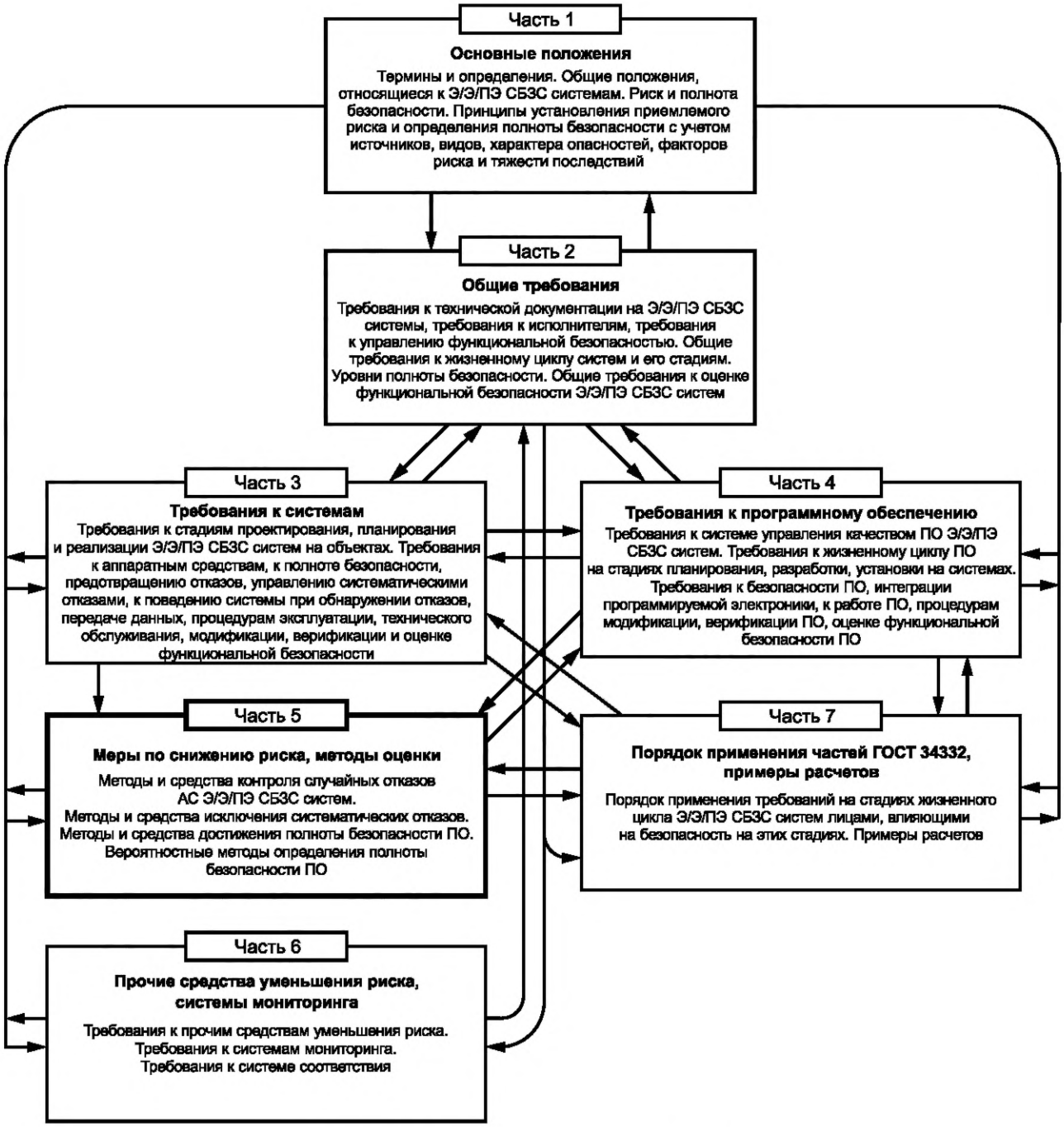

Рисунок 1 — Структура комплекса ГОСТ 34332

Поправка к ГОСТ 34332.4—2021 Безопасность функциональная систем, связанных с безопасностью зданий и сооружений. Часть 4. Требования к программному обеспечению

В каком месте	Напечатано	Должно быть		
Предисловие. Таблица согласования	—	Казахстан	KZ	Госстандарт Республики Казахстан

(ИУС № 4 2022 г.)

М Е Ж Г О С У Д А Р С Т В Е Н Н Ы Й С Т А Н Д А Р Т

БЕЗОПАСНОСТЬ ФУНКЦИОНАЛЬНАЯ СИСТЕМ, СВЯЗАННЫХ С БЕЗОПАСНОСТЬЮ ЗДАНИЙ И СООРУЖЕНИЙ

Часть 4

Требования к программному обеспечению

Functional safety of building/construction safety-related systems. Part 4. Requirements for software

Дата введения — 2022—01—01

1 Область применения

1.1 Настоящий стандарт:

- применяется совместно с ГОСТ 34332.1 — ГОСТ 34332.3 и ГОСТ 34332.5;

- распространяется на программное обеспечение (ПО) электрических, электронных, программируемых электронных (Э/Э/ПЭ), связанных с безопасностью зданий и сооружений системы (далее — Э/Э/ПЭ СБЗС системы), включая комплексные системы безопасности (КСБ), устанавливаемые или установленные во вновь возводимых или реконструируемых зданиях и сооружениях (далее — объекты) всех отраслей экономики независимо от форм собственности и ведомственной принадлежности, включая жилые, общественные и производственные здания и сооружения, в том числе на Э/Э/ПЭ СБЗС системы объектов инфраструктуры перерабатывающей промышленности, энергетики, транспорта, гидротехнических и мелиоративных сооружений, включая линейные объекты, для обеспечения их безопасности и антитеррористической защищенности;

- распространяется на любое ПО, являющееся частью Э/Э/ПЭ СБЗС системы, либо на ПО, используемое для разработки Э/Э/ПЭ СБЗС систем в области применения ГОСТ 34332.1 — ГОСТ 34332.3. Такое ПО, связанное с его безопасностью (СБ ПО), включает в себя операционные системы, системное ПО, программы, применяемые в коммуникационных сетях, интерфейсы пользователей и обслуживающего персонала, встроенные программно-аппаратные средства, а также прикладные программы;

- устанавливает требования к тем действиям и процедурам, которые должны быть выполнены на стадиях проектирования, планирования и реализации СБ ПО.

Примечания

1 Под реализацией СБ ПО понимается его установка на аппаратных средствах (АС), программируемых электронных (ПЭ) СБЗС системах (далее — ПЭ СБЗС системы), включая комплексные системы безопасности (КСБ), интеграцию, наладку, оценку и подтверждение соответствия, в том числе на объекте.

2 Области применения настоящего стандарта и ГОСТ 34332.3 тесно взаимосвязаны. Эту взаимосвязь (см. рисунок 2) следует учитывать при применении настоящего стандарта;

Издание официальное

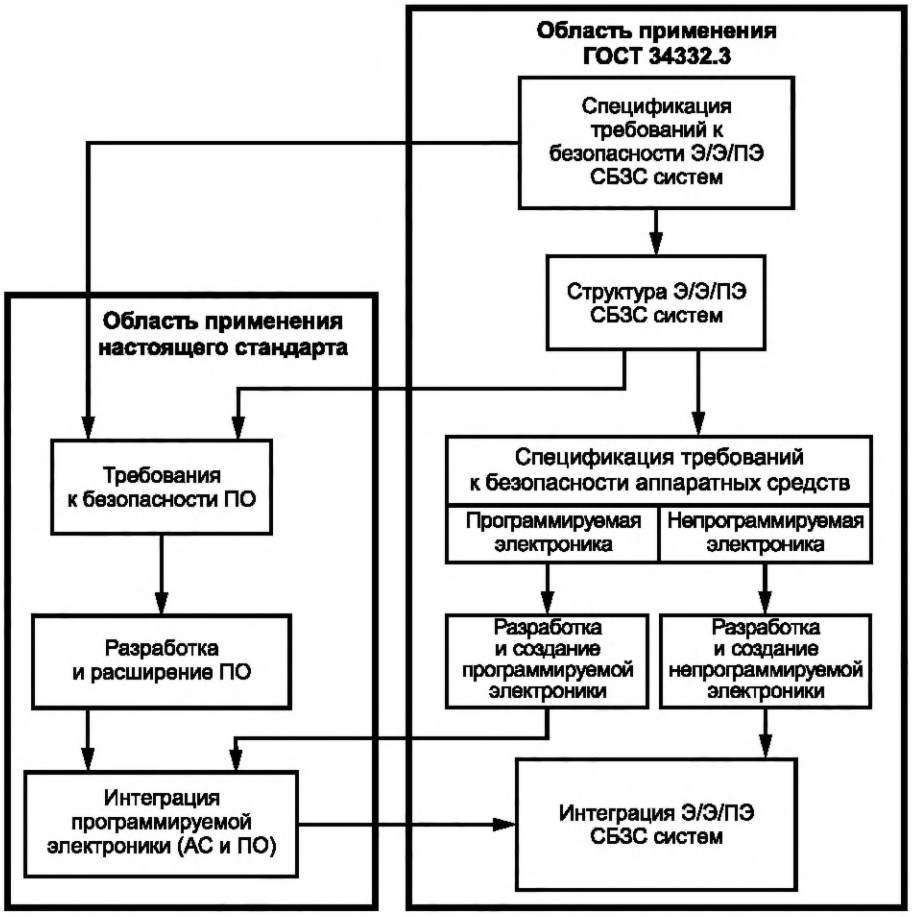

Рисунок 2 — Взаимосвязь областей применения настоящего стандарта и ГОСТ 34332.3

- устанавливает конкретные требования к инструментальным средствам поддержки, используемым для разрабатываемых и конфигурируемых Э/Э/ПЭ СБЗС систем, включая КСБ, в соответствии с ГОСТ 34332.2 и ГОСТ 34332.3.

Примечания

1 СБ ПО, применяемое при разработке концепции безопасности объекта (здания, сооружения), определении назначения и области применения Э/Э/ПЭ СБЗС систем, систем снижения риска на основе неэлектрических технологий и прочих средств уменьшения риска, при анализе опасностей и риска, подлежащих компенсации Э/Э/ПЭ СБЗС системами и другими прочими средствами уменьшения риска, при определении требований к функциям безопасности и распределении функций безопасности по Э/Э/ПЭ СБЗС системам, относят ко всем Э/Э/ПЭ СБЗС системам, включая КСБ [см. ГОСТ 34332.2—2017 (рисунок 1, блоки 1—5)].

2 После распределения функций безопасности по конкретным Э/Э/ПЭ СБЗС системам в настоящем стандарте подробно рассмотрены требования к СБ ПО части этих систем, а именно ПЭ СБЗС системам, на стадии их проектирования и реализации;

- предусматривает определение функции безопасности и стойкости к систематическим отказам СБ ПО.

Примечания

1 Описание части работы в отношении спецификации Э/Э/ПЭ СБЗС систем, выполненной по ГОСТ 34332.3—2021 (подраздел 8.2), не следует повторять в настоящем стандарте.

2 Определение функций безопасности и стойкости к систематическим отказам СБ ПО представляет собой итеративную процедуру.

3 Структура документации установлена в ГОСТ 34332.2—2017 (раздел 5 и приложение А). В структуре документации могут быть учтены процедуры, используемые в компаниях, а также практика, сложившаяся в отдельных областях применения систем;

- устанавливает требования к стадиям части жизненного цикла (ЖЦ) ПО ПЭ СБЗС системы и к действиям, которые следует предпринимать в процессе проектирования и разработки СБ ПО. Эти

требования включают в себя применение методов и средств, ранжированных по уровням требуемой стойкости к систематическим отказам и предназначенных для предотвращения ошибок и управления ошибками и отказами в СБ ПО;

- устанавливает требования к информации, относящейся к вопросам подтверждения соответствия аспектов ПО Э/Э/ПЭ СБЗС системы, которая должна быть предоставлена организации, осуществляющей интеграцию программируемой электроники в Э/Э/ПЭ СБЗС систему и интеграцию Э/Э/ПЭ СБЗС систем в КСБ;

- устанавливает:

- требования к подготовке предоставления информации и проведения процедур, касающихся СБ ПО, которое необходимо пользователям для эксплуатации и технического обслуживания Э/Э/ПЭ СБЗС систем,

- требования, предъявляемые к организациям, выполняющим модификацию СБ ПО, включая требования ГОСТ 34332.2 и ГОСТ 34332.3, требования к инструментальным средствам поддержки, таким как средства разработки и проектирования, трансляции, тестирования и отладки, управления конфигурацией.

1.2 Настоящий стандарт не распространяется на ПО Э/Э/ПЭ СБЗС системы, которая является единственной системой, способной осуществить необходимое снижение риска на объекте, и требуемая полнота безопасности этой системы ниже, чем определено уровнем полноты безопасности (УПБ) УПБ 1 — наиболее низким уровнем полноты безопасности. Настоящий стандарт не применяется к ПО медицинского оборудования.

1.3 Общая структура ГОСТ 34332.1 — ГОСТ 34332.7 и роль настоящего стандарта в достижении функциональной безопасности Э/Э/ПЭ СБЗС систем показана на рисунке 1.

2 Нормативные ссылки

В настоящем стандарте использованы нормативные ссылки на следующие межгосударственные стандарты:

ГОСТ 34332.1—2017 Безопасность функциональная систем, связанных с безопасностью зданий и сооружений. Часть 1. Основные положения

ГОСТ 34332.2—2017 Безопасность функциональная систем, связанных с безопасностью зданий и сооружений. Часть 2. Общие требования

ГОСТ 34332.3—2021 Безопасность функциональная систем, связанных с безопасностью зданий и сооружений. Часть 3. Требования к системам

ГОСТ 34332.5—2021 Безопасность функциональная систем, связанных с безопасностью зданий и сооружений. Часть 5. Меры по снижению риска, методы оценки

Примечание — При пользовании настоящим стандартом целесообразно проверить действие ссылочных стандартов и классификаторов на официальном интернет-сайте Межгосударственного совета по стандартизации, метрологии и сертификации (www.easc.by) или по указателям национальных стандартов, издаваемым в государствах, указанных в предисловии, или на официальных сайтах соответствующих национальных органов по стандартизации. Если на документ дана недатированная ссылка, то следует использовать документ, действующий на текущий момент, с учетом всех внесенных в него изменений. Если заменен ссылочный документ, на который дана датированная ссылка, то следует использовать указанную версию этого документа. Если после принятия настоящего стандарта в ссылочный документ, на который дана датированная ссылка, внесено изменение, затрагивающее положение, на которое дана ссылка, то это положение применяется без учета данного изменения. Если ссылочный документ отменен без замены, то положение, в котором дана ссылка на него, применяется в части, не затрагивающей эту ссылку.

3 Термины, определения и сокращения

3.1 Термины и определения

3.1 В настоящем стандарте применены термины по ГОСТ 34332.1 — ГОСТ 34332.3, а также следующие термины с соответствующими определениями:

3.1.1 **анимация** (animation): Имитация работы программного обеспечения (или отдельной его части), предназначенная для отображения существенных аспектов поведения программируемой электронной системы, связанной с безопасностью зданий и сооружений.

Примечания

1 Анимация применима, например, к спецификации требований для представления проекта системы на достаточно высоком уровне в соответствующем формате.

2 Анимация позволяет оценить специфическое поведение системы при задании параметров и данных, близких к реальным.

3.1.2 **данные** (data): Информация, представленная в виде, удобном для передачи, интерпретации либо при обработке компьютером.

Примечание — Данные могут быть представлены в виде статической информации (например, совокупности заданных значений либо представления географической информации) или команды для задания последовательности выполнения предварительно созданных функций.

3.1.3 **динамическое тестирование** (dynamic testing): Работа программного обеспечения и/или аппаратного средства, выполняемая под контролем и планомерно для демонстрации наличия/отсутствия установленного функционала.

Примечание — При динамическом тестировании в отличие от статистического анализа требуется выполнение программ.

3.1.4 **жизненный цикл программного обеспечения;** ЖЦ ПО (software lifecycle): Период времени, включающий в себя стадии разработки: требований к программному обеспечению, программного обеспечения, кодирования, тестирования, интеграции, установки, а также стадию модификации.

3.1.5 **избыточность** (redundancy): Наличие средств в дополнение к средствам или данным, достаточным для выполнения требуемой операции или предоставления информации.

Пример — Примерами избыточности являются дублирование функциональных компонентов и добавление избыточных битов, в том числе битов четности.

3.1.6 **инструментальные средства поддержки программного обеспечения** (инструментальные средства поддержки ПО): Средства разработки, проектирования, кодирования, тестирования, отладки, управления конфигурацией программного обеспечения.

3.1.7 **полнота безопасности программного обеспечения** (полнота безопасности ПО) (software safety integrity): Количественная характеристика, определяющая вероятность того, что программное обеспечение программируемой электронной системы будет выполнять заданные функции безопасности при указанных условиях в течение установленного периода времени.

3.1.8 **прикладное программное обеспечение** (application software): Часть программного обеспечения программируемой электронной системы, которая по специфицированным функциям выполняет задачу, связанную с безопасностью управляемого оборудования, но не обеспечивает функционирования и не предоставляет сервисы для программируемого устройства.

3.1.9 **программируемая электроника;** ПЭ (programmable electronic, PE): Средство, которое основано на использовании компьютерных технологий и может включать в себя аппаратное средство и программное обеспечение, а также устройства ввода и/или вывода.

Примечания

1 Данный термин включает в себя микроэлектронные устройства, основанные на одном или нескольких центральных процессорах и связанных с ними устройствах памяти и т. п.

2 К программируемым электронным устройствам относятся: микропроцессоры, микроконтроллеры, программируемые контроллеры, специализированные интегральные схемы, программируемые логические контроллеры, другие устройства на основе компьютерных технологий (например, микропроцессорные датчики, преобразователи, устройства привода).

3.1.10 **программное обеспечение;** ПО (software): Продукт интеллектуальной деятельности, включающий в свой состав программы, процедуры, данные, правила и ассоциированную информацию, имеющую отношение к работе системы обработки данных.

Примечание — Программное обеспечение является независимым от носителя записи, на котором оно записано.

3.1.11 **связанное с безопасностью программное обеспечение**; СБ ПО (safety-related software): Программное обеспечение, которое используется для реализации функций безопасности в системах, связанных с безопасностью.

3.1.12 **системное программное обеспечение** (system software): Часть программного обеспечения программируемой электронной системы, которая обеспечивает функционирование и предоставля-

ет сервисы для программируемого устройства, в отличие от прикладного программного обеспечения, которое по запрограммированным специфицированным функциям выполняет задачу безопасности управляемого оборудования.

3.1.13 **средство поддержки программного обеспечения в автономном режиме** (software off-line support tool): Программное средство, не имеющее непосредственного доступа к системе, связанной с безопасностью, в процессе ее функционирования.

Примечания

1 Средства поддержки программного обеспечения в автономном режиме могут быть разделены на три класса: Т1, Т2 и Т3. Средство поддержки класса Т1 не генерирует тех выходов, которые могут прямо или косвенно способствовать исполнению кода (включая данные) системы, связанной с безопасностью. Примерами средств поддержки класса Т1 служат текстовый редактор, требования или средства поддержки проектирования без возможности автоматического создания кода, а также инструменты управления конфигурацией.

2 Средство поддержки класса Т2 поддерживает тестирование либо проверку проекта или исполняемого кода, в которых ошибки в инструменте хотя не позволяют выявить дефекты, но и не могут напрямую создавать ошибки в исполняемом программном коде. Примерами средств поддержки класса Т2 служат генератор тестовых программ, средства измерения тестового охвата и средства статического анализа.

3 Средство поддержки класса Т3 генерирует программы, которые явно или неявно включаются в рабочую программу системы, связанной с безопасностью. Примерами средств поддержки класса Т3 служат оптимизирующий компилятор с неочевидной связью между исходным кодом программы и сгенерированным объектным кодом, а также компилятор, который включает исполнимый пакет программ в рабочую программу.

3.1.14 **средство поддержки программного обеспечения в режиме реального времени** (software on-line support tool): Программное средство, имеющее непосредственный доступ к системе, связанной с безопасностью, в процессе ее функционирования.

3.1.15 **стойкость к систематическим отказам**; ССО (systematic capability): Мера уверенности (выраженная в диапазоне ССО 1 — ССО 4) в том, что систематическая полнота безопасности элемента соответствует требованиям заданного значения уровня полноты безопасности для определенной функции безопасности элемента, если этот элемент применен в соответствии с указаниями, определенными для этого элемента в соответствующем руководстве по безопасности.

Примечания

1 ССО определяется с учетом требований по предотвращению систематических отказов и управлению ими (см. ГОСТ 34332.3 и [4]).

2 Механизм систематического отказа зависит от природы элемента. Например, для элемента, представляющего ПО, должны быть рассмотрены только механизмы ошибок в программах. Для элемента, включающего в себя аппаратные средства (АС) и ПО, должны быть рассмотрены механизмы систематических отказов как для АС, так и для ПО.

3 Стойкость к систематическим отказам элемента ССО *N* при выполнении определенной функции безопасности означает, что элемент соответствует уровню полноты безопасности *N* для систематических отказов, если этот элемент применен в соответствии с указаниями для данного элемента в соответствующем руководстве по безопасности.

3.1.16 **существующее ранее программное обеспечение** (pre-existingsoftware): Компонент программного обеспечения, используемый в настоящее время, не разрабатываемый специально для выполняемого проекта либо для системы, связанной с безопасностью.

Примечание — ПО могло быть коммерчески доступным продуктом или, возможно, было разработано для ранее выпущенных изделия или системы. Существующее ранее ПО могло быть (или не могло быть) создано в соответствии с требованиями настоящего стандарта.

3.1.17 **тестовая программа** (test harness): Программный продукт, позволяющий имитировать ту среду, в которой будет действовать разрабатываемое программное обеспечение или аппаратное средство путем передачи тестовых данных в программу и регистрации ответа.

3.1.18 **уровень полноты безопасности программного обеспечения;** УПБ ПО (software safety integrity level): Стойкость к систематическим отказам элемента программного обеспечения, являющегося частью подсистемы или системы, связанной с безопасностью.

Примечание — УПБ характеризует функцию безопасности всей системы, но не любую из ее отдельных подсистем либо элементов, которые реализуют эту функцию безопасности. Поэтому ПО, как и любой его элемент, не имеет собственного УПБ. Однако фраза «программное обеспечение с УПБ N» означает, что «обоснована уверенность (выраженная значениями от 1 до 4) в том, что функция безопасности, реализуемая элементом (ПО), не будет приводить к сбою из-за соответствующих механизмов систематических отказов, если этот элемент

(ПО) применяется согласно указаниям, определенным в руководстве по безопасности, разработанном для такого элемента».

3.1.19 **функциональный блок** (functional unit): Объект аппаратного средства и/или программного обеспечения, выполняющий определенную задачу.

3.1.20 **язык с ограниченной варьируемостью** (limited variability language): Текстовый или графический язык программирования, предназначенный для коммерческих и промышленных программируемых электронных логических контроллеров, диапазон возможностей которого ограничен применением этих устройств.

Пример — К языкам с ограниченной варьируемостью, которые используют для представления прикладных программ для систем на основе программируемых логических контроллеров, относятся:

- многоступенчатые схемы: графический язык, состоящий из набора символов для входов (представляющих поведение, характерное для таких устройств, как контакты, которые в нормальном состоянии замкнуты или разомкнуты), соединенных с помощью линий (указывающих направление тока), с символами, обозначающими выходы (представляющими поведение, свойственное реле);

- булева алгебра: низкоуровневый язык, основанный на булевых операторах, таких как И, ИЛИ и НЕ, с возможностью добавления некоторых мнемонических инструкций;

- функциональные блок-диаграммы: в дополнение к булевым операторам допускают использование более сложных функций, таких как операции с файлами, считывание и запись блоков данных, команд для сдвиговых регистров и устройств, задающих последовательность;

- последовательные функциональные схемы: графическое представление многостадийной программы, состоящее из взаимосвязанных шагов, действий и ориентированных связей с промежуточными состояниями.

3.2 Сокращения

В настоящем стандарте использованы следующие сокращения:

АС — аппаратное(ые) средство(а);

ЖЦ — жизненный цикл;

КСБ — комплексная система безопасности;

ПЛК — программируемый логический контроллер;

ПО — программное обеспечение;

ПЭ — программируемая(ое, ый) электронная(ое, ый) — в отношении системы, средства, модуля, элемента;

СБ — связанная(ое, ый) с безопасностью — в отношении системы, средства, устройства, модуля, элемента;

СБЗС система — система, связанная с безопасностью здания(ий), сооружения(ий);

СБ ПО — связанное с безопасностью программное обеспечение;

УО — управляемое оборудование;

УПБ — уровень полноты безопасности;

УПБ ПО — уровень полноты безопасности программного обеспечения.

4 Общие требования

Для обеспечения соответствия настоящему стандарту следует выполнять требования, установленные в ГОСТ 34332.2—2017 (подраздел 5.1).

5 Требования к документации

Цели и требования, предъявляемые к документации, — по ГОСТ 34332.2—2017 (подраздел 5.2).

6 Требования к управлению программным обеспечением Э/Э/ПЭ СБЗС систем

6.1 Цели настоящего раздела

Цели настоящего раздела — по ГОСТ 34332.2—2017 (подраздел 6.1).

6.2 Требования

6.2.1 В дополнение к требованиям, установленным в ГОСТ 34332.2—2017 (подраздел 6.2), следует выполнять перечисленные ниже требования.

6.2.2 Планирование функциональной безопасности СБ ПО осуществляют таким образом, чтобы определять стратегию поставок, разработки, интеграции, верификации, подтверждения соответствия и модификации СБ ПО в такой мере, в какой этого требует УПБ функций, реализуемых Э/Э/ПЭ СБЗС системой.

Примечание — Философия данного подхода состоит в использовании планирования функциональной безопасности в качестве возможности для приспособления настоящего стандарта для учета требуемой полноты безопасности для каждой функции безопасности, реализуемой Э/Э/ПЭ СБЗС системой.

6.2.3 Система управления конфигурацией ПО должка быть организована таким образом, чтобы:

- использовать административные и технические средства контроля на протяжении ЖЦ СБ ПО для управления изменениями в программах и тем самым гарантировать непрерывное выполнение указанных в спецификациях требований к СБ ПО;
- гарантировать выполнение операций, необходимых для того, чтобы продемонстрировать достижение заданной стойкости к систематическим отказам СБ ПО;
- осуществлять тщательную поддержку с использованием уникальной идентификации всех элементов конфигурации, которые необходимы для обеспечения требований полноты безопасности Э/Э/ПЭ СБЗС системы. В элементы конфигурации включают, как минимум, следующее:
  - анализ системы безопасности и требования к системе безопасности,
  - спецификацию ПО и проектную документацию,
  - исходный текст программ,
  - план и результаты тестирования,
  - документацию о проверках,
  - ранее разработанные программные элементы и пакеты, которые должны быть включены в Э/Э/ПЭ СБЗС систему,
  - все инструментальные средства и системы разработки, которые использовались при создании, тестировании или выполнении иных действий с ПО Э/Э/ПЭ СБЗС системы;
- использовать процедуры контроля над внесением изменений для того, чтобы:
  - предотвращать несанкционированные модификации,
  - документально оформлять запросы на выполнение модификаций,
  - анализировать влияние предлагаемых модификаций и утверждать/не утверждать модификации,
  - подробно документально оформлять модификации и выдавать полномочия на выполнение всех утвержденных модификаций,
  - устанавливать основные параметры конфигурации системы для этапов разработки ПО и документально оформлять тестирование (частичное) интеграции системы,
  - гарантировать объединение и встраивание всех подсистем ПО (включая переработку более ранних версий).

Примечания

1 Для осуществления руководства и применения административных и технических средств контроля необходимы наличие полномочий и принятие управленческих решений.

2 С одной стороны, анализ влияния может включать в себя неформальную оценку. С другой стороны, анализ влияния может включать в себя строгий формальный анализ возможного неблагоприятного воздействия всех предложенных изменений, которые могут быть неправильно поняты или неверно осуществлены. Руководство по анализу влияния приведено в ГОСТ 34332.5;

- гарантировать осуществление соответствующих мер по корректной загрузке прошедших подтверждение соответствия элементов СБ ПО и данных в систему во время ее выполнения.

Примечание — Допускается рассматривать отдельные целевые системы, а также общие системы. Для СБ ПО, кроме приложений, может понадобиться безопасный метод загрузки, например для встроенных программ программируемых устройств;

- документально оформлять перечисленную ниже информацию для обеспечения возможности последующего аудита функциональной безопасности: состояния конфигурации, текущего состояния системы, обоснования (с учетом результатов анализа влияния) и утверждения всех модификаций, подробного описания всех модификаций;

- строго документально оформлять каждую версию СБ ПО, обеспечивать хранение всех версий ПО и всей относящейся к ним документации, а также версий данных для предоставления возможности сопровождения и выполнения модификаций на протяжении всего периода использования разработанного программного продукта.

Примечание — Дополнительная информация по управлению конфигурацией приведена в ГОСТ 34332.5.

7 Требования к жизненному циклу связанного с безопасностью программного обеспечения

7.1 Общие положения

7.1.1 Цель

Целью требований, излагаемых в настоящем подразделе, является разделение процесса разработки СБ ПО на этапы и процессы (см. рисунки 2—5).

7.1.2 Требования

7.1.2.1 В соответствии с ГОСТ 34332.2—2017 (раздел 6) при планировании Э/Э/ПЭ СБЗС системы должен быть выбран и специфицирован ЖЦ для разработки ПО этой системы.

7.1.2.2 Может быть использована любая модель ЖЦ ПО Э/Э/ПЭ СБЗС системы при условии, что она соответствует всем целям и требованиям настоящего подраздела.

7.1.2.3 Каждая стадия ЖЦ СБ ПО может быть разделена на элементарные процессы. Для каждой стадии должны быть определены область применения, входные данные и выходные данные.

Примечание — На рисунках 3 и 4 представлены модели части ЖЦ проектирования и реализации Э/Э/ПЭ СБЗС системы и ПО ПЭ СБЗС системы соответственно.

7.1.2.4 Для рассмотрения ЖЦ сложных систем обеспечения безопасности объекта, включая КСБ, с учетом АС и ПО данных систем на стадиях проектирования и реализации может быть применена подробная V-образная модель.

Примечания

1 V-образная модель стадий проектирования и реализации КСБ, состоящей из ряда Э/Э/ПЭ СБЗС систем, представлена на рисунке 5, а на рисунке 6 — V-образная модель стадий проектирования и реализации СБ ПО для КСБ, содержащих ряд ПЭ СБЗС систем. На рисунках отражены итерационные процессы.

2 Модель стадий ЖЦ СБ ПО, которая соответствует требованиям настоящего раздела, может быть соответственно настроена для конкретных потребностей проекта или организации. Полный список стадий ЖЦ, приведенный в разделе 7, относится к большим заново разрабатываемым системам. Для небольших систем может оказаться целесообразным объединить стадии проектирования системы СБ ПО и проектирования архитектуры ПО.

3 Характеристики систем, управляемых данными, описаны в приложении Ж (например, языки программирования с полной или ограниченной изменчивостью, степень конфигурации данных) и могут быть использованы для настройки ЖЦ СБ ПО.

7.1.2.5 Любая настройка ЖЦ СБ ПО должна быть обоснована функциональной безопасностью.

7.1.2.6 Процедуры обеспечения качества и безопасности должны быть интегрированы в процессы ЖЦ СБ ПО.

7.1.2.7 Для каждой стадии ЖЦ следует использовать соответствующие методы и средства.

Примечания

1 Рекомендации по выбору методов и средств, а также ссылки на ГОСТ 34332.5 приведены в приложениях А и Б.

2 В ГОСТ 34332.5 приведены рекомендации по выбору конкретного метода для обеспечения свойств, требуемых для достижения систематической полноты безопасности. Методы и средства, выбранные в соответствии с этими рекомендациями, сами по себе не гарантируют достижения необходимой полноты безопасности.

3 Достижение систематической полноты безопасности СБ ПО зависит от выбора методов с учетом следующих факторов:

- согласованность и взаимодополняющий характер выбранных методов, языков и инструментов для всего цикла разработки:
- уровень восприятия разработчиками используемых ими методов, языков и инструментальных средств;
- в какой степени методы, языки и инструментальные средства отвечают тем задачам, с которыми сталкиваются разработчики в процессе разработки.

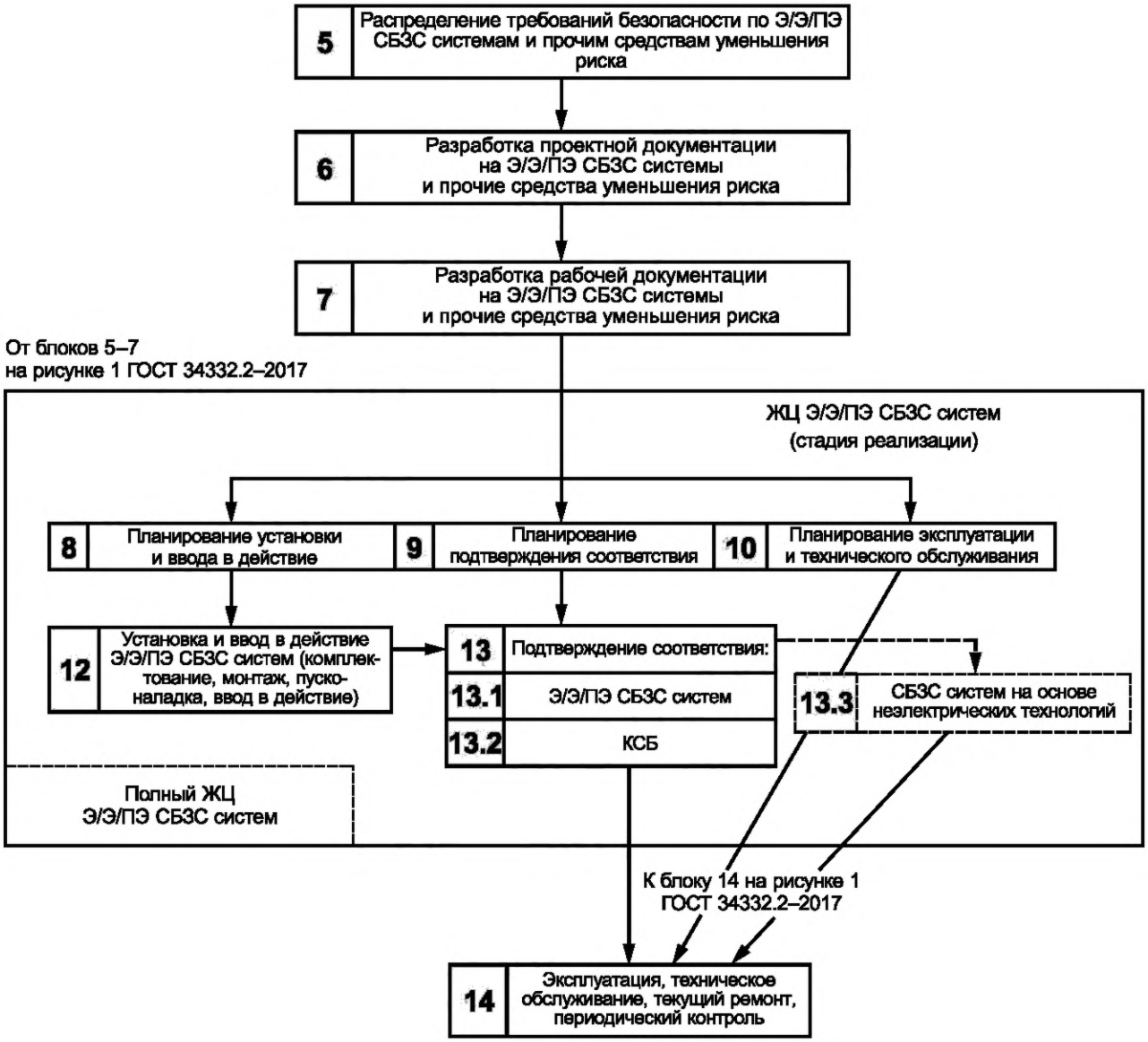

Рисунок 3 — Часть ЖЦ Э/Э/ПЭ СБЗС системы

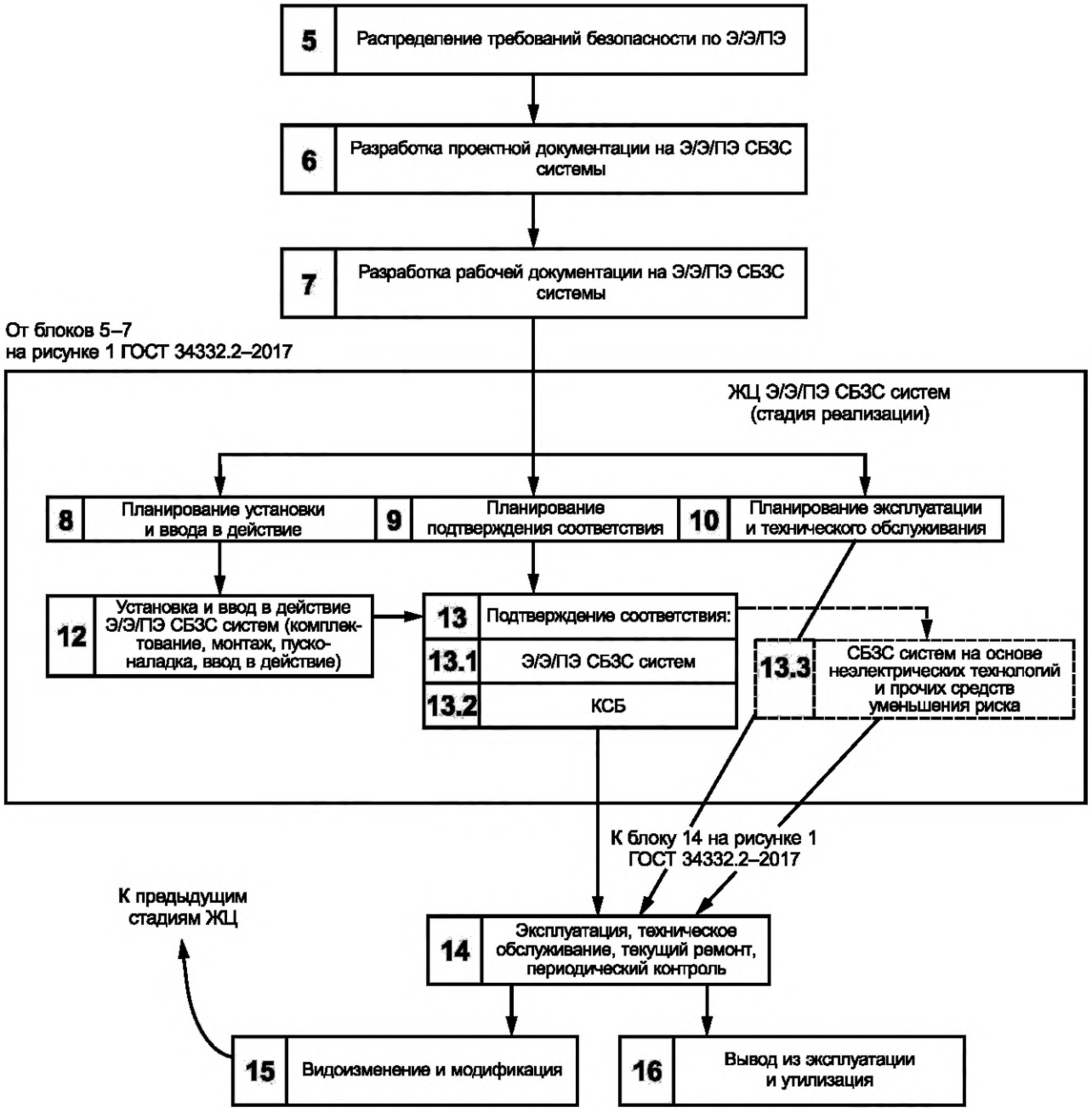

Рисунок 4 — Часть ЖЦ ПО ПЭ СБЗС системы

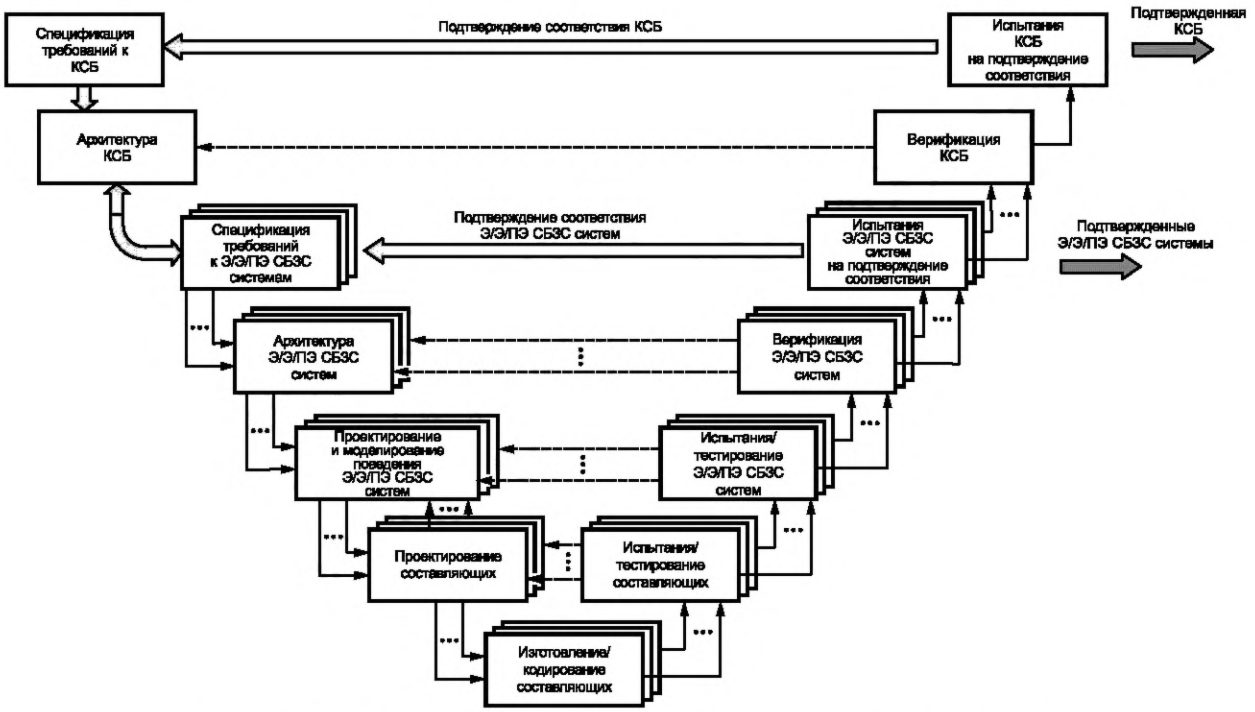

Рисунок 5 — V-образная модель стадий разработки и реализации ЖЦ КСБ

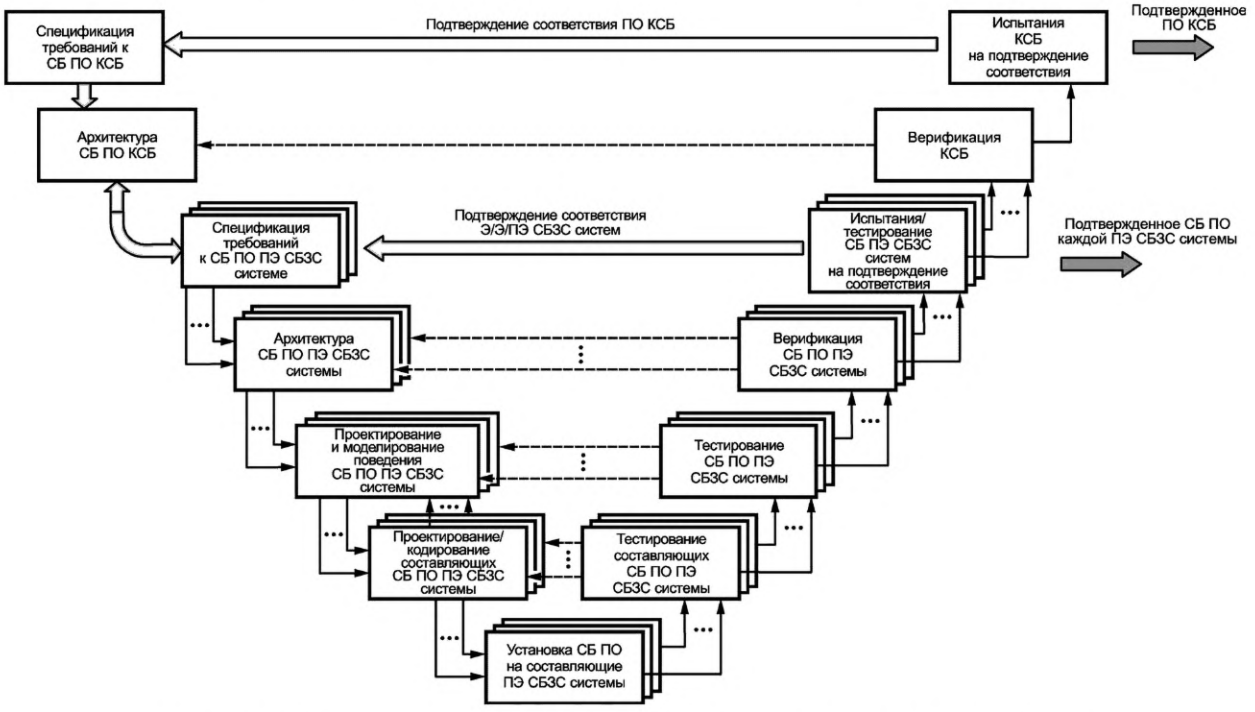

Рисунок 6 — V-образная модель стадий разработки и реализации ЖЦ СБ ПО КСБ

7.1.2.8 Результаты процессов ЖЦ программного СБ ПО должны быть документально оформлены (см. раздел 5).

Примечание — В ГОСТ 34332.2—2017 (раздел 5) рассмотрено документальное оформление результатов стадий ЖЦ Э/Э/ПЭ СБЗС системы. При разработке некоторых Э/Э/ПЭ СБЗС систем результаты определенных стадий ЖЦ системы могут быть оформлены в виде отдельных документов, тогда как результаты других стадий могут быть объединены в один документ. Существенным является требование, чтобы результаты стадии ЖЦ Э/Э/ПЭ СБЗС системы соответствовали ее предназначению.

7.1.2.9 Если на какой-либо стадии ЖЦ СБ ПО возникает необходимость внести изменение, относящееся к более ранней стадии ЖЦ, то, во-первых, используя анализ влияния, следует определить, какие модули и элементы СБ ПО будут изменены и, во-вторых, какие действия на более ранней стадии ЖЦ Э/Э/ПЭ СБЗС системы должны быть выполнены повторно.

Примечание — С одной стороны, анализ влияния может представлять собой неформальную оценку. С другой стороны, он может включать в себя строгий формальный анализ предполагаемых неблагоприятных последствий потенциальных изменений, которые, возможно, были не до конца продуманы или не должным образом реализованы. Руководящие указания по анализу влияния приведены в ГОСТ 34332.5—2021 (пункт Б.5.23).

7.2 Спецификация требований к связанному с безопасностью программному обеспечению

Примечание — Спецификацию требований к СБ ПО формируют на стадии проектирования ПО (см. блок 6 на рисунке 4) после распределения требований безопасности в отношении Э/Э/ПЭ СБЗС систем, систем на основе неэлектрических технологий (в случае их применения) и прочих средств уменьшения риска (см. блок 5 на рисунке 4).

7.2.1 Цели

7.2.1.1 Во-первых, следует определить требования к СБ ПО как требования к функциям безопасности СБ ПО и требования к стойкости к систематическим отказам СБ ПО.

7.2.1.2 Во-вторых, должны быть установлены требования к функциям безопасности ПО каждой Э/Э/ПЭ СБЗС системы, которые необходимы для реализации данных функций.

7.2.1.3 И в-третьих, следует сформировать требования к стойкости относительно систематических отказов СБ ПО, которые необходимы для достижения УПБ, установленного для каждой функции безопасности, реализуемой Э/Э/ПЭ СБЗС системой.

7.2.2 Требования

Примечания

1 В большинстве случаев требования данного пункта выполняются комбинацией базового встраиваемого ПО и программных модулей, которые разработаны специально для конкретного применения. Именно комбинация этих двух видов ПО позволяет достигать характеристик, описанных в подразделах, приведенных ниже. Точная граница между базовым и прикладным ПО зависит от выбранной архитектуры программной системы (см. 7.4.3).

2 При выборе соответствующих методов и средств (см. приложения А и Б) для осуществления требований настоящего пункта необходимо рассмотреть следующие свойства [см. приложение В относительно интерпретации данных свойств и ГОСТ 34332.5—2021 (приложение Е), в котором приведены их неформальные определения] спецификации требований к ПО Э/Э/ПЭ СБЗС системы:

а) полнота охвата потребностей безопасности ПО;

б) корректность охвата потребностей безопасности ПО;

в) отсутствие ошибок в самой спецификации, включая отсутствие неоднозначности;

г) четкость требований к системе;

д) отсутствие неблагоприятного взаимовлияния функций, не связанных с безопасностью, и функций, связанных с безопасностью, реализуемых ПО Э/Э/ПЭ СБЗС системы;

е) способность обеспечения проведения оценки и подтверждения соответствия ПО.

3 Уровень безопасности, которому должно соответствовать СБ ПО, представлен набором функций безопасности и соответствующими требованиями к полноте безопасности, определенными для функций ПО в проекте Э/Э/ПЭ СБЗС системы. Полный набор требований к Э/Э/ПЭ СБЗС системе гораздо шире, так как включает в себя также функции безопасности, которые не выполняются ПО. Полнота спецификации требований к ПО Э/Э/ПЭ СБЗС системы решающим образом зависит от эффективности более ранних стадий ЖЦ системы.

7.2.2.1 Требования к СБ ПО приведены в требованиях к Э/Э/ПЭ СБЗС системе по ГОСТ 34332.3—2021 (подраздел 8.2).

7.2.2.2 Спецификация требований к СБ ПО должна быть выработана на основе заданных требований к функциональной безопасности Э/Э/ПЭ СБЗС системы по ГОСТ 34332.3 и других требований к планированию безопасности (см. раздел 6). Эта информация должна быть доступна для разработчика СБ ПО.

Примечания

1 Это требование означает, что должно быть тесное взаимодействие между разработчиком Э/Э/ПЭ системы и разработчиком СБ ПО (см. ГОСТ 34332.3 и настоящий стандарт). По мере того как требования к безопасности и архитектура СБ ПО (см. 7.4.3) становятся более определенными, может усиливаться влияние на архитектуру АС Э/Э/ПЭ СБЗС системы, и по этой причине становится значимым конструктивное сотрудничество между разработчиками АС и ПО (см. рисунок 2).

2 В проект ПО может быть включено действующее, зарекомендовавшее себя ПО. Разрабатываемое ПО может быть создано без учета спецификации требований к формируемой системе. В 7.4.2.12 представлены требования к функционирующему ПО, соответствующему спецификации требований к СБ ПО Э/Э/ПЭ СБЗС системы.

7.2.2.3 Спецификация требований к СБ ПО должна быть достаточно подробной для обеспечения стадий проектирования и реализации необходимой информации с целью достижения требуемой полноты безопасности (включая требования к независимости, см. ГОСТ 34332.3) и выполнения оценки уровня функциональной безопасности.

Примечание — Уровень детальности спецификации может быть изменен в зависимости от сложности применения. Соответствующая спецификация функционального поведения СБ ПО может включать в свой состав требования к точности, синхронизации и быстродействию, емкости, устойчивости, допустимой перегрузке и другим свойствам Э/Э/ПЭ СБЗС системы, характеризующим конкретное применение.

7.2.2.4 Для решения проблемы независимости должен быть выполнен соответствующий анализ отказов СБ ПО по общей причине. Если выявлены вероятные механизмы отказа, то должны быть приняты эффективные меры защиты.

Примечание — В приложении Е приведены методы достижения одного аспекта независимости ПО.

7.2.2.5 Разработчик СБ ПО должен ознакомиться с информацией согласно 7.2.2.2 для того, чтобы убедиться в том, что требования определены адекватным образом. В частности, разработчик СБ ПО должен учесть:

- функции безопасности;
- конфигурацию или архитектуру системы;
- требования к полноте безопасности АС (программируемой электроники, датчиков и исполнительных устройств);
- требования к стойкости к систематическим отказам СБ ПО;
- быстродействие и время отклика;
- синхронизуемость времен взаимодействия Э/Э/ПЭ СБЗС систем и их составляющих;
- интерфейсы оборудования и оператора, включая объективно прогнозируемые нарушения.

Примечание — Необходимо рассмотреть совместимость с любыми действующими способами применения ПО.

7.2.2.6 В специфицированных требованиях к СБ ПО должны быть подробно описаны все соответствующие режимы работы УО Э/Э/ПЭ СБЗС системы и любых АС или систем, подсоединенных к Э/Э/ПЭ СБЗС системе в том случае, если отсутствует их адекватное определение в требованиях к безопасности, специфицированных для Э/Э/ПЭ СБЗС системы.

7.2.2.7 В спецификации требований к СБ ПО должны быть определены и документально оформлены все связанные с безопасностью ограничения и иные ограничения, относящиеся к взаимодействию между АС и ПО.

7.2.2.8 В той степени, в которой это требуется при описании проекта архитектуры АС Э/Э/ПЭ системы, и с учетом возможного увеличения ее сложности в спецификации требований к СБ ПО должны быть учтены:

- самоконтроль ПО (например, по ГОСТ 34332.5);
- мониторинг ПЭ аппаратуры, датчиков и исполнительных устройств;
- периодическое тестирование функций безопасности во время выполнения программы;
- предоставление возможности тестирования функций безопасности во время работы УО;
- функции ПО для выполнения контрольных испытаний и всех диагностических тестов для обеспечения соблюдения требований полноты безопасности Э/Э/ПЭ СБЗС системы.

Примечание — Увеличение сложности, которое может возникнуть вследствие вышеупомянутых соображений, может потребовать пересмотра архитектуры Э/Э/ПЭ СБЗС системы.

7.2.2.9 Если Э/Э/ПЭ СБЗС система должка выполнять функции, не относящиеся к безопасности, эти функции должны быть четко указаны в спецификации требований к СБ ПО.

Примечание — Требования относительно отсутствия взаимовлияния функций, связанных и не связанных с безопасностью, приведены в 7.4.2.8 и 7.4.2.9.

7.2.2.10 В спецификации требований к СБ ПО должны быть указаны необходимые характеристики безопасности программной продукции, а не ее проекта, как это определяется при планировании

системы безопасности [см. ГОСТ 34332.3—2021 (раздел 6)]. С учетом 7.2.2.1—7.2.2.9 в зависимости от конкретных обстоятельств должны быть определены следующие положения:

требования к функциям ПО Э/Э/ПЭ СБЗС системы:

а) функции, которые позволяют УО достигать или поддерживать безопасное состояние,

б) функции, связанные с обнаружением, оповещением и обработкой ошибок АС программируемой электроники,

в) функции, связанные с обнаружением, оповещением и обработкой ошибок датчиков и исполнительных устройств,

г) функции, связанные с обнаружением, оповещением и обработкой ошибок в самом СБ ПО (самоконтроль ПО),

д) функции, связанные с периодическим тестированием функций безопасности в режиме реального времени (в предопределенной операционной среде),

е) функции, связанные с периодическим тестированием функций безопасности в автономном режиме (т. е. в тех условиях, в которых функция безопасности УО не выполняется),

ж) функции, обеспечивающие модификацию программируемой электроники Э/Э/ПЭ СБЗС системы,

и) интерфейсы функций, не связанных с безопасностью,

к) интерфейсы между ПО и ПЭ системой,

л) быстродействие и время отклика АС системы,

м) синхронизуемость времен взаимодействия Э/Э/ПЭ СБЗС систем и их составляющих.

Примечание — В интерфейсы должны быть включены средства программирования в автономном и неавтономном режимах;

н) средства коммуникации, связанные с безопасностью [см. ГОСТ 34332.3—2021 (пункт 8.3.15)];

требования к стойкости к систематическим отказам СБ ПО:

а) уровень(уровни) полноты безопасности для каждой функции безопасности по первому перечислению,

б) требования независимости между функциями.

7.2.2.11 Если требования к СБ ПО выражены или выполнены в виде конфигурации данных, то эти данные должны быть:

- согласованы с требованиями к Э/Э/ПЭ СБЗС системе;
- выражены значениями из допустимого диапазона и санкционированными комбинациями реализующих их параметров;
- определены способом, который совместим с базовым ПО (например, последовательность выполнения, время выполнения, структуры данных и т. д.).

Примечания

1 Эти требования к прикладным данным относятся, в частности, к применениям, управляемым данными. Они характеризуются следующим образом: исходный код уже существует, а главная цель разработки состоит в том, что конфигурация данных правильно задает поведение, требуемое от применения. Между элементами данных могут быть сложные зависимости, и достоверность данных может меняться с течением времени.

2 Указания по системам, управляемым данными, приведены в приложении Ж.

7.2.2.12 Если данные определяют интерфейс между ПО и внешними системами, то в дополнение к ГОСТ 34332.3—2021 (пункт 8.3.15) необходимо рассмотреть следующие факторы и характеристики:

- согласованность во времени при определении данных;
- недостоверные, находящиеся вне определенного диапазона или несвоевременные значения;
- время отклика и пропускная способность, включая условия максимальной загрузки;
- время выполнения в наиболее/наименее приемлемых случаях и зависание;
- переполнение и недостаточная емкость хранилища данных.

7.2.2.13 Параметры эксплуатации должны быть защищены:

- от недостоверных, находящихся вне определенного диапазона или несвоевременных значений:
  - несанкционированных изменений,
  - искажений

Примечания

1 Следует рассмотреть защиту от несанкционированных изменений как программных, так и непрограммных механизмов. Необходимо отметить, что эффективная защита от несанкционированных изменений ПО может от-

рицательно отразиться на безопасности, например в том случае, если изменения необходимо выполнить быстро и в напряженных условиях.

2 Несмотря на то что человек может быть частью СБ системы, относящейся к безопасности [см. ГОСТ 34332.2—2017 (раздел 1)], требования, обусловленные человеческим фактором и связанные с проектированием Э/ЭЛ1Э СБЗС систем, в настоящем стандарте подробно не рассматриваются. Однако при необходимости должны быть изучены следующие соображения:

- в информационной системе оператора следует использовать общепринятые пиктографические представления и терминологию. Они должны быть четкими, понятными и лишенными ненужных деталей и/или аспектов;
- информация об УО, выведенная оператору на экран, должна быть подробной, достоверной и отображающей физическое состояние УО;
- если на экране дисплея оператору отражена информация о выполняющихся в УО процессах и/или если оператор выполняет определенные действия, последствия которых невозможно сразу заметить, то выведенная на экран в автоматическом режиме информация должна быть представлена таким образом, чтобы отображалось то состояние, в котором находится данная система, или та последовательность действий, в соответствии с которой указано, какое состояние последовательности достигнуто, какие операции могут быть выполнены и какими могут быть возможные последствия.

7.3 Планирование подтверждения соответствия безопасности системы для аспектов программного обеспечения

Примечания

1 Эта стадия относится к блоку 9 на рисунке 3.

2 Подтверждение соответствия для ПО обычно не может быть выполнено отдельно от используемых АС и системной среды.

7.3.1 Цель

Целью требований настоящего подраздела является разработка плана подтверждения соответствия связанных с безопасностью программных аспектов безопасности.

7.3.2 Требования

7.3.2.1 В ходе планирования должны быть определены процедурные и технические шаги, которые необходимо выполнить для демонстрации того, что ПО соответствует требованиям безопасности.

7.3.2.2 В плане подтверждения соответствия программных аспектов безопасности системы должно быть отражено следующее:

а) точная дата, когда должно происходить подтверждение соответствия;

б) перечень лиц, осуществляющих подтверждение соответствия;

в) идентификация соответствующих режимов работы УО, включая:

1) подготовку к использованию, в том числе установку (загрузку) и настройку,

2) работу в режиме запуска и обучения, в автоматическом, ручном, полуавтоматическом и стационарном режимах,

3) переустановку, выключение, сопровождение,

4) разумно прогнозируемые ненормальные условия и ошибки оператора;

г) идентификация СБ ПО, для которого должна быть проведена процедура подтверждения соответствия, для каждого режима работы УО до момента его ввода в эксплуатацию;

д) техническая стратегия для подтверждения соответствия (например, аналитические методы, статистическое тестирование и т. п.);

е) методы/средства и процедуры в соответствии с перечислением д), которые должны быть использованы для подтверждения того, что каждая функция безопасности соответствует установленным требованиям к функциям безопасности и требованиям к стойкости к систематическим отказам СБ ПО;

ж) условия, в которых должны происходить процедуры подтверждения соответствия (например, при тестировании может потребоваться использование калиброванных инструментов и оборудования);

и) критерии прохождения/непрохождения подтверждения соответствия;

к) политика и процедуры, используемые для оценки результатов подтверждения соответствия, в частности при оценке отказов.

Примечание — Эти требования основаны на общих требованиях ГОСТ 34332.2—2017 (подраздел 7.10).

7.3.2.3 Подтверждение соответствия служит обоснованием выбранной стратегии. В техническую стратегию для подтверждения соответствия СБ ПО должна быть включена следующая информация о выборе:

- ручных или автоматических методов или и тех и других;

- статических или динамических методов или и тех и других;
- аналитических или статистических методов или и тех и других:
- критериев приемки на основе объективных факторов или экспертной оценки или и того и другого.

7.3.2.4 В рамках процедуры подтверждения соответствия аспектов СБ ПО, если этого требует УПБ [см. ГОСТ 34332.2—2017 (раздел 8)], область применения и содержание плана подтверждения соответствия безопасности системы, аспектов ПО должны быть изучены экспертом или третьей стороной, представляющей эксперта. В эту процедуру включают также заявление о присутствии эксперта при испытаниях.

7.3.2.5 В критерии прохождения/непрохождения при завершении подтверждения соответствия СБ ПО включают:

- необходимые входные сигналы, включая их последовательность и значения;
- предполагаемые выходные сигналы, включая их последовательность и значения;
- другие необходимые критерии приемки, например: использование памяти, синхронизацию, допустимые интервалы значений.

Последовательность выполнения требований относительно проектирования и разработки ПО приведена в 7.4.1.1—7.4.1.6.

7.4 Проектирование и разработка программного обеспечения

Примечание — Эта стадия относится к блокам 6 и 7 на рисунке 3.

7.4.1 Цели

7.4.1.1 Создание такой архитектуры ПО, которая соответствовала бы заданным требованиям к СБ ПО с необходимым УПБ.

7.4.1.2 Оценка требований, предъявляемых к ПО со стороны архитектуры АС Э/Э/ПЭ СБЗС, включая значение взаимодействия между АС и ПО Э/Э/ПЭ СБЗС системы, для обеспечения безопасности УО.

7.4.1.3 Выбор надлежащего набора инструментальных средств, включая языки программирования и компиляторы, интерфейсы системы времени выполнения, интерфейсы пользователя и форматы представления данных, который должен соответствовать заданному УПБ на протяжении всего ЖЦ СБ ПО и способствовать выполнению процессов верификации, оценки, подтверждения соответствия и модификации.

7.4.1.4 Проектирование и реализация ПО, соответствующего специфицированным требованиям к СБ ПО для достижения необходимого УПБ. Это ПО должно быть пригодным для анализа и верификации и обладать способностью к безопасной модификации.

7.4.1.5 Проверка выполнения требований к СБ ПО (в отношении необходимых функций безопасности и стойкости к систематическим отказам ПО).

7.4.1.6 Гарантирование в той мере, в которой это уместно, того, что конфигурирование данных программируемой электроники СБЗС системы соответствует указанным в настоящем подразделе требованиям стойкости к систематическим отказам ПО.

7.4.2 Общие требования

7.4.2.1 В зависимости от технологии процесса разработки ПО ответственными за соответствие требованиям 7.4 могут быть следующие лица: только поставщик СБ ПО (например, поставщик), только пользователь, имеющий непосредственное отношение к решаемым задачам (например, разработчик прикладных программ), или и поставщик, и пользователь. Распределение ответственности должно быть определено во время планирования системы безопасности (см. раздел 6).

Примечание — Характеристики системы и архитектуры ПО, для которых необходима определенность при выборе подразделения, ответственного за соответствие требованиям 7.4, приведены в 7.4.3.

7.4.2.2 В соответствии с определенным УПБ и конкретными техническими требованиями к функции безопасности выбирают метод проектирования с теми характеристиками, которые способствуют:

абстрактному представлению, разделению на модули и другие характеристики, контролирующие уровень сложности;

выражению:

а) выполняемых функций,
б) обмена данными между элементами,
в) информации, относящейся к последовательности и времени выполнения программ,
г) ограничений синхронизации,

д) параллельного и синхронизированного доступа к совместно используемым ресурсам,

е) структур данных и их свойств,

ж) проектных предположений и их зависимостей,

и) обработки исключений,

к) проектных предположений (предварительных условий, постусловий, инвариантов),

л) комментариев;

возможности описания нескольких представлений проекта, включая представление структуры и представление поведения;

пониманию разработчиками и другими лицами специфики достижения УПБ и требований проекта;

верификации и оценке соответствия.

7.4.2.3 На этапе проектирования должны быть предусмотрены тестируемость и способность к модификации системы безопасности для облегчения реализации этих свойств в окончательной версии Э/Э/ПЭ СБЗС системы.

7.4.2.4 Выбор метода проектирования определен характеристиками, облегчающими модификацию ПО. К числу таких характеристик относят модульность, скрытие информации и инкапсуляцию.

7.4.2.5 Представление проекта следует основывать на той нотации, которая является однозначно определенной или ограничена до однозначно определенных свойств.

7.4.2.6 В проект, по мере возможности, включают ту часть ПО, которая связана с безопасностью.

7.4.2.7 В проект ПО включают (соразмерно требуемому УПБ) средства самоконтроля потоков управления и потоков данных. При обнаружении отказа должны быть выполнены соответствующие действия.

7.4.2.8 Если при использовании ПО должны быть реализованы функции как относящиеся, так и не относящиеся к безопасности, то его в целом следует рассматривать как связанное с безопасностью ПО, если в проекте не предусмотрены соответствующие меры, гарантирующие, что отказы функций, не связанных с безопасностью, не смогут оказать негативного влияния на функции, связанные с безопасностью.

7.4.2.9 Если при использовании ПО должны быть реализованы функции безопасности, имеющие различные УПБ, то следует считать, что все ПО имеет уровень наивысший среди этих уровней, если только в проекте не будет продемонстрирована независимость функций, имеющих различные УПБ. Должно быть продемонстрировано, что либо независимость обеспечена в пространстве и во времени, либо любые нарушения независимости находятся под контролем. Обоснование независимости функций должно быть документально оформлено.

Примечание — Методы достижения одного аспекта независимости ПО приведены в приложении Е.

7.4.2.10 Если стойкость к систематическим отказам элемента ПО ниже, чем требуется для УПБ функции безопасности, к которой он относится, то этот элемент следует использовать в сочетании с другими элементами, что будет гарантировать стойкость к систематическим отказам и соответствовать УПБ функции безопасности.

7.4.2.11 Если функция безопасности реализована с использованием комбинации элементов ПО, для которых известны их значения стойкости к систематическим отказам, то к такой комбинации элементов следует применять требования стойкости к систематическим отказам, представленные в ГОСТ 34332.3—2021 (пункт 8.3.3).

Примечание — Существует различие между функцией безопасности, от начала до конца реализуемой одним элементом или более, и функцией безопасности элемента (т. е. каждого элемента, участвующего в реализации). Если два элемента объединяются для достижения более высокой стойкости к систематическим отказам, то в такой комбинации каждый из данной пары элементов должен быть способен к предотвращению/смягчению опасного события. При этом функции безопасности каждого из этих элементов не обязательно должны быть идентичными, и не требуется, чтобы каждый из элементов комбинации был способен независимо обеспечивать функциональную безопасность, которая задана для всей комбинации.

Пример — В управлении скоростью наполнения емкости жидкостью функция безопасности предотвращения нежелательного ускорения наполнения полностью реализуется на двух процессорах. Функция безопасности элемента, реализуемая основным контроллером, управляет клапаном подачи воды в режиме запрос/ответ. Функция безопасности элемента, реализуемая другим процессором, выполняет разного рода контроль [см. ГОСТ 34332.5—2021 (приложение В, пункт В.3.4)] и аварийный останов подачи жидкости в случае необходимости во избежание переполнения емкости. Применение комбинации этих двух процессоров способствует тому, что выполнение в полном объеме функции безопасности «предотвращение нежелательного ускорения подачи жидкости» будет обеспечено.

7.4.2.12 Если для реализации всей или части функции безопасности повторно использован определенный элемент ПО, то этот элемент должен соответствовать следующим требованиям систематической полноты безопасности:

- требованиям одного из следующих способов обеспечения соответствия:

- способ 1с: разработка, отвечающая установленным требованиям, а именно требованиям настоящего стандарта для предотвращения и управления систематическими отказами в ПО;

- способ 2с: проверка при эксплуатации данного элемента. Необходимо предоставить свидетельства относительно проверки элемента при его эксплуатации [см. ГОСТ 34332.3—2021 (пункт 8.3.14)];

- способ 3с: оценка разработки, соответствующей требованиям. Должно быть документально подтверждено соблюдение требований 7.4.2.13.

Примечания

1 Способы 1с, 2с и 3с соответствуют способам, описанным в ГОСТ 34332.3—2021 (пункт 8.3.2.3) для элементов ПО. Они приведены в настоящем пункте исключительно для того, чтобы минимизировать обращение к ГОСТ 34332.3.

2 Действующее ПО может быть доступным коммерческим продуктом, или оно, возможно, разработано конкретной организацией для предыдущего изделия или системы. Однако данное ПО может как соответствовать, так и не соответствовать требованиям настоящего стандарта.

3 Требования к действующим элементам ПО применяют также к библиотеке времени выполнения операций или интерпретатору;

- должно быть представлено руководство по безопасности [см. ГОСТ 34332.3—2021 (приложение Г) и приложение Г], которое дает достаточно точное и полное описание элемента для обеспечения оценки полноты конкретной функции безопасности, полностью или частично зависящей от действующего элемента ПО.

Примечания

1 Руководство по безопасности может быть получено на основе собственной документации поставщика элемента и описания процесса разработки поставщика элемента, или создано, или расширено дополнительными действиями, квалифицированно выполненными разработчиком, отвечающим за безопасность системы, или третьей стороной. В некоторых случаях может понадобиться инженерный анализ для создания спецификации или разработки документации, соответствующей требованиям данного пункта с учетом сложившихся правовых условий (например, авторское право или права интеллектуальной собственности).

2 Обоснование элемента ПО может быть разработано при планировании безопасности (см. раздел 6).

7.4.2.13 Согласно способу 3с существующий ранее элемент ПО должен соответствовать следующим требованиям:

а) спецификация требований к СБ ПО для элемента в его новом применении должна быть подробно документально оформлена в соответствии с требованиями настоящего стандарта для любого элемента, связанного с безопасностью, с той же стойкостью к систематическим отказам. Спецификация требований к СБ ПО ПЭ СБЗС системы должна охватывать функциональное и безопасное поведение и применяться к элементу в его новом использовании, как определено в подразделе 7.2;

б) в обоснование для использования элемента ПО должны быть включены свидетельства о том, что были рассмотрены требуемые свойства системы безопасности, определенные в 7.2.2, 7.4.3—7.4.7, 7.5.2, 7.7.2, 7.8.2, 7.9.2 и разделе 8 с учетом требований приложения В;

в) в документально оформленном проекте элемента должны быть подробно представлены свидетельства соответствия спецификации требований и требуемой стойкости к систематическим отказам (см. 7.4.3, 7.4.5, 7.4.6 и таблицы А.2 и А.4 приложения А);

г) при соблюдении требований по перечислениям а), б) должна быть учтена интеграция ПО и АС (см. 7.5 и таблицу А.6 приложения А);

д) должно быть представлено доказательство того, что для элемента ПО выполнены процедуры проверки и подтверждения соответствия с использованием систематического подхода с документально оформленным тестированием и анализом всех частей проекта элемента и кода (см. 7.4.7, 7.4, 7.5, 7.7, 7.9 и таблицы А.5 — А.7 и А.9 приложения А, а также связанные с ними таблицы приложения Б).

Примечание — Для удовлетворения требованиям тестирования может быть использован положительный опыт применения вероятностных методов и метода «черного ящика» (см. таблицы А.7 приложения А и Б.3 приложения Б);

е) если элемент ПО выполняет функции, которые не требуются системе, связанной с безопасностью, то должно быть представлено свидетельство того, данные функции не влияют на Э/Э/ПЭ СБЗС систему относительно соответствия требованиям функциональной безопасности.

Примечание — Способы, соответствующие данному требованию, включают в себя:
- устранение таких функций из проекта;
- их отключение;
- использование соответствующей архитектуры системы (например, декомпозиция на части, упаковка в отдельный файл, разнообразие, проверка достоверности выходов);
- широкое тестирование;

ж) должно быть доказано, что идентифицированы все вероятностные механизмы отказа элемента ПО и реализованы соответствующие меры их ослабления.

Примечание — Соответствующие меры ослабления включают в себя:
- использование соответствующей архитектуры системы (например, декомпозиция на части, упаковка в отдельный файл, разнообразие, проверка достоверности выходов);
- широкое тестирование;

и) при планировании использования элемента должны быть идентифицированы конфигурация элемента ПО, среды выполнения ПО и АС, а также (при необходимости) конфигурация системы компиляции/редактирования связей;

к) обоснованием использования элемента ПО должно быть проведение для него процедуры подтверждения соответствия только для тех применений, которые отвечают установленным в руководстве для этого элемента по безопасности конкретных изделий [см. ГОСТ 34332.3—2021 (приложение Г) и приложение Г].

7.4.2.14 В уместных случаях к данным и языкам генерации данных необходимо применять нижеприведенные требования:

Примечание — Руководящие указания по системам, управляемым данными, приведены в приложении Ж;

- если ПЭ СБЗС система обладает действующей функциональностью, которая сконфигурирована данными и соответствует конкретным требованиям применения, то проект прикладного ПО должен соответствовать степени конфигурируемости использования по функциональности и сложности ПЭ СБЗС системы;
- если функциональность ПЭ СБЗС системы определена в значительной степени или в основном конфигурационными данными, то для предотвращения появления отказов во время проектирования, производства, установки (загрузки) и модификации данных конфигурации и гарантии того, что конфигурационные данные правильно формируют логику применения, следует использовать соответствующие методы и средства;
- спецификация структур данных должна быть выполнена:
  - не противоречащей функциональным требованиям системы, включая данные применения,
  - в полном объеме,
  - внутренне непротиворечивой,
  - такой, чтобы структуры данных были защищены от изменения или повреждения;
- если ПЭ система обладает необходимой функциональностью, которая сконфигурирована данными и соответствует установленным требованиям применения, то сам процесс конфигурации должен быть документально оформлен.

7.4.3 Требования к проектированию архитектуры программного обеспечения

Примечания

1 Архитектура ПО представляет основные элементы и подсистемы СБ ПО, их взаимосвязь, способ реализации необходимых характеристик, и в частности, полноты безопасности. Архитектура СБ ПО также определяет общее поведение ПО и то, как элементы ПО реализуют интерфейс и взаимодействуют между собой. Примеры основных компонентов ПО включают в себя операционные системы, базы данных, подсистемы ввода и вывода УО, коммуникационные подсистемы, прикладные программы, инструментальные средства программирования и диагностики и т. п.

2 В некоторых отраслях промышленности архитектура ПО может называться «описание функций или спецификация функций проекта» (хотя эти документы могут также включать в себя вопросы, относящиеся к АС).

3 В некоторых случаях пользовательского прикладного программирования, в частности в языках, используемых в ПЛК, архитектура определяется поставщиком как стандартная характеристика ПЛК. Однако в соответствии с требованиями настоящего стандарта к поставщику может быть предъявлено требование гарантировать пользователю соответствие поставляемого продукта требованиям 7.4. Пользователь приспосабливает ПЛК, используя стандартные возможности программирования, например многозвенные логические схемы. При этом требования 7.4.3—7.4.8 также действуют. Требование определения и документирования архитектуры следует рассматривать

как требование к той информации, которую пользователь может использовать при выборе ПЛК (или эквивалентного ему устройства) для применения.

4 С точки зрения системы безопасности стадия разработки архитектуры ПО соответствует периоду, когда для ПО разрабатывают базовую стратегию безопасности.

5 Хотя стандарты комплекса ГОСТ 34332 устанавливают численные целевые показатели отказов для функций безопасности, выполняемых Э/Э/ПЭ СБЗС системами, систематическая полнота безопасности, как правило, не определяется количественно. Полноту безопасности ПО определяют как стойкость к систематическим отказам со шкалой уверенности от 1 до 4. Для целей настоящего стандарта принято, что программная ошибка может быть безопасной или опасной в зависимости от специфики использования ПО. Архитектура системы/ПО должна быть выбрана такой, чтобы опасные отказы элемента были ограничены каким-либо архитектурным ограничением, а в выбранных методах разработки эти ограничения были учтены. Согласно требованиям настоящего стандарта методы разработки и подтверждения соответствия жестко регламентированы, что согласовано с требуемой стойкостью к систематическим отказам.

6 Для выбора соответствующих методов и средств (см. приложения А и Б), которые отвечают требованиям настоящего пункта, должны быть рассмотрены и предусмотрены следующие свойства [см. руководство по интерпретации свойств в приложении В и неформатные описания методов и средств в ГОСТ 34332.5—2021 (приложение Е)] архитектуры ПО:

- полнота спецификации требований к ПО Э/Э/ПЭ СБЗС системы;
- корректность спецификации требований к ПО Э/Э/ПЭ СБЗС системы;
- отсутствие собственных ошибок проекта;
- простота и ясность;
- предсказуемость поведения;
- верифицируемость и тестируемость проекта;
- отказоустойчивость;
- защита от отказов по общей причине, вызванной внешним событием.

7.4.3.1 В зависимости от характера разработки ПО ответственность за соответствие требованиям 7.4.4 могут нести несколько сторон. Распределение ответственности должно быть документально оформлено во время планирования СБЗС системы [см. ГОСТ 34332.2—2017 (раздел 6)].

7.4.3.2 Проект архитектуры ПО должен быть создан поставщиком ПО и/или разработчиком ПО. Описание архитектуры должно быть подробным и соответствовать следующим требованиям:

- содержать выбор и обоснование (см. 7.1.2.7) интегрированного набора методов и средств, которые будут необходимы в течение ЖЦ ПО ПЭ СБЗС системы для обеспечения соответствия требованиям к СБ ПО для заданного УПБ. Эти методы и средства включают в себя стратегию проектирования ПО для обеспечения устойчивости к отказам (совместимую с АС) и избегания отказов, в том числе (при необходимости) избыточность и разнообразие;
- основываться на разделении на элементы/подсистемы, для каждой из которых должна быть представлена следующая информация:
  а) проводилась ли верификация и если проводилась, то при каких условиях,
  б) связан ли каждый из этих компонентов/подсистем с безопасностью или не связан,
  в) существует ли стойкость к систематическим отказам для компонента/подсистемы ПО;
- определять все взаимодействия между ПО и АС, а также оценивать и детализировать их значение.

Примечание — Если взаимодействие между ПО и АС уже определено архитектурой системы, то достаточно сослаться на архитектуру системы;

- использовать для представления архитектуры нотацию, которая является однозначно определенной или ограничена до подмножества однозначно определенных характеристик;
- содержать набор проектных характеристик, которые должны быть использованы для поддержания полноты безопасности всех данных. В число таких данных допускается включать входные и выходные производственные, коммуникационные данные, данные интерфейса оператора, сопровождения и хранящиеся во внутренних базах данных;
- определять соответствующие тесты интеграции архитектуры ПО для обеспечения выполнения спецификации требований к ПО ПЭ СБЗС системы для заданного УПБ.

7.4.3.3 Любые изменения, которые может потребоваться внести в спецификацию требований к ПЭ СБЗС системе (см. 7.4.3.2), должны быть согласованы с разработчиком ПЭ СБЗС системы и документально оформлены.

Примечание — Итерационное взаимодействие между архитектурой АС и ПО является неизбежным (см. рисунки 5 и 6), поэтому существует необходимость в обсуждении с разработчиком АС таких вопросов, как спецификация тестирования интеграции программируемой электроники и ПО (см. 7.5).

7.4.4 Требования к инструментальным средствам поддержки, включая языки программирования

Примечание — При выборе соответствующих методов и средств (см. приложения А и Б) для обеспечения выполнения требований настоящего пункта должны быть рассмотрены и предусмотрены следующие свойства [см. руководство по интерпретации свойств в приложении В и неформальные описания методов и средств в ГОСТ 34332.5—2021 (приложение Е)] архитектуры ПО:

- уровень, до которого инструментальные средства поддерживают разработку ПО поддержки с требуемыми свойствами СБ ПО;
- четкость работы и функциональность инструментальных средств;
- корректность и воспроизводимость результата.

7.4.4.1 ПО инструментальных средств поддержки, работающее в неавтономном режиме, должно быть рассмотрено как элемент ПО ПЭ СБЗС системы.

7.4.4.2 Действие по выбору ПО инструментальных средств поддержки, работающих в автономном режиме, должно быть тесно связанным с частью действий по разработке ПО.

Примечания

1 Требования к ЖЦ СБ ПО приведены в 7.1.2.

2 Для увеличения полноты безопасности СБ ПО за счет уменьшения вероятности появления или необнаружения отказов во время его разработки следует использовать соответствующие инструментальные средства, работающие в неавтономном режиме и поддерживающие разработку ПО. Примерами инструментальных средств, используемых на стадиях ЖЦ разработки ПО, являются:

- инструментальные средства преобразования или трансляции, которые преобразуют ПО или представление проекта (например, текст или схему) из одного уровня абстрактного представления в другой уровень — инструментальные средства усовершенствования проекта, компиляторы, ассемблеры, компоновщики, редакторы связей, загрузчики и средства генерации кода;
- средства оценки и подтверждения соответствия, такие как статические анализаторы кода, средства контроля тестового охвата, средства доказательства теорем и средства моделирования;
- инструментальные средства диагностики для контроля и модификации ПО в процессе эксплуатации;
- инструментальные средства инфраструктуры, такие как системы поддержки разработок;
- инструментальные средства управления конфигурацией, такие как средства управления версиями информации;
- инструментальные средства данных применения, которые создают или поддерживают данные, необходимые для определения параметров и создания экземпляров функций системы. Такие данные включают в себя параметры функций, диапазоны инструментальных средств, уровни срабатывания и отключения аварийных сигналов состояния выхода, которые будут восприняты как отказы.

3 Инструментальные средства поддержки, работающие в автономном режиме, должны быть выбраны как интегрируемые. Инструментальные средства интегрированы, если они работают совместно так, что выходы одного инструментального средства имеют соответствующее содержание и формат для автоматической передачи на вход следующего инструментального средства, минимизируя таким образом возможность ошибки при повторной обработке промежуточных результатов.

4 Инструментальные средства поддержки, работающие в автономном режиме, должны быть выбраны как совместимые с потребностями применения ПЭ СБЗС системы и интегрированного комплекса инструментальных средств.

5 Необходимо рассмотреть и предусмотреть наличие соответствующих инструментальных средств для предоставления сервисов, необходимых для всего ЖЦ Э/Э/ПЭ СБЗС системы (например, средства поддержки спецификаций, проектирования, реализации, документирования, модификации).

6 Необходимо уделить внимание компетентности пользователей выбранных инструментальных средств. Требования к их компетентности приведены в ГОСТ 34332.2—2017 (раздел 6).

7.4.4.3 Выбор инструментальных средств поддержки, работающих в автономном режиме, должен быть обоснован.

7.4.4.4 Все инструментальные средства поддержки классов Т2 и Т3, работающие в автономном режиме, должны сопровождаться спецификацией или документацией на выбранное средство, в которой четко определено поведение инструментального средства и представлены любые инструкции или ограничения при его использовании. Требования к ЖЦ разработки ПО приведены в 7.1.2, а для категорий ПО инструментальных средств поддержки, работающих в автономном режиме, — 7.4.4.2.

Примечание — Такая «спецификация или документация на инструментальное средство» не является руководством по безопасности применяемого элемента [см. ГОСТ 34332.3—2021 (приложение Г) и приложение Г] непосредственно для самого инструментального средства. Руководство по безопасности для применяемого элемента касается функционирующего элемента, который включен в исполняемую СБ систему. Если последний эле-

мент сгенерирован инструментальным средством класса Т3 и затем включен в исполняемую СБ систему, то любая соответствующая информация (например, если в документации для оптимизирующего компилятора указано, что порядок оценки параметров функции не гарантируется) из спецификации или документации на инструментальное средство должна быть включена в руководство по безопасности для применяемого элемента, что позволяет провести оценку полноты конкретной функции безопасности, которая полностью или частично зависит от элемента, включенного в исполняемую СБ систему.

7.4.4.5 Для определения уровня доверия к инструментальным средствам и возможных механизмов отказа инструментальных средств, которые могут повлиять на применяемое ПО, должна быть выполнена оценка инструментальных средств классов Т2 и Т3 поддержки ПО в автономном режиме. Если механизмы отказа идентифицированы, то должны быть использованы соответствующие меры по их ослаблению.

Примечания

1 ПО исследования опасности и работоспособности (HAZOP) реализует один из методов анализа последствий возможных отказов, который можно использовать для ПО инструментального средства.

2 Примерами мер по ослаблению являются предотвращение ошибок, ограниченное использование функциональности инструментального средства, проверка выходных результатов инструментального средства, применение разнообразных инструментальных средств для той же цели.

7.4.4.6 Для каждого инструментального средства в классе Т3 должно быть представлено свидетельство о том, что инструментальное средство соответствует спецификации или документации на него. Такое свидетельство может быть основано на определенной комбинации информации о предыдущем удовлетворительном использовании в подобных средах и для подобных применений (в данной организации или в других организациях) и на подтверждении соответствия инструментального средства, как определено в 7.4.4.7.

Примечания

1 Знание предыстории использования может способствовать подтверждению с высокой степенью доверия относительно готовности инструментального средства к работе. При этом также должны быть учтены записи ошибок/несоответствий, когда инструментальное средство используют для разработки в новой среде.

2 Свидетельство для инструментального средства класса Т3 может быть применено также к инструментальным средствам класса Т2 для оценки правильности их результатов.

7.4.4.7 Результаты подтверждения соответствия инструментальных средств должны быть документально оформлены и содержать:

- хронологическую запись действий по подтверждению соответствия;
- версию используемого руководства по инструментальному средству;
- функции инструментального средства, для которых проводится процедура подтверждения соответствия;
- используемые инструментальные средства и оборудование;
- результаты действия по подтверждению соответствия: документально оформленные результаты подтверждения соответствия, которые должны установить, что соответствие ПО подтверждено или существуют причины для отказа;
- контрольные примеры и их результаты для последующего анализа;
- несоответствия между ожидаемыми и фактическими результатами.

7.4.4.8 Если свидетельство соответствия по 7.4.4.6 недоступно, то должны быть приняты эффективные меры для управления отказами рассматриваемой ПЭ СБЗС системы, которые являются следствием ошибок при работе инструментального средства.

Примечание — Примером такой меры является применение средства генерации разнообразного избыточного кода, позволяющее обнаруживать и управлять отказами рассматриваемой ПЭ СБЗС системы, которые произошли в ней из-за ошибок транслятора.

7.4.4.9 Должна быть проверена совместимость инструментальных средств в интегрированном комплексе инструментальных средств.

Примечание — Инструментальные средства считаются интегрированными в том случае, если они работают совместно таким образом, что выходы одного инструментального средства, соответствующее содержание и формат пригодны для автоматической передачи на вход следующего инструментального средства, минимизируя тем самым возможность появления ошибки при повторной обработке промежуточных результатов [см. ГОСТ 34332.5—2021 (Б.3.5 приложения Б)].

7.4.4.10 В той степени, в которой этого требует УПБ, представление ПО или проекта (включая язык программирования) должно быть выбрано таким, чтобы оно:

- имело транслятор, пригодный для данного использования (если необходимо), включая подтверждение соответствия требованиям межгосударственных или международных стандартов;
- использовало свойства языка, определенные исключительно для него;
- соответствовало характеристикам применения;
- обладало свойствами, облегчающими обнаружение ошибок при проектировании или программировании;
- поддерживало характеристики, соответствующие методу проектирования.

Примечания

1 Языки программирования являются классом ПО и используются для представлений проекта. Транслятор преобразует ПО или представление проекта (например, текст или блок-схему) из одного уровня абстракции на другой уровень. Примерами трансляторов являются инструменты усовершенствования проекта, компиляторы, ассемблеры, компоновщики, редакторы связей, загрузчики и инструменты генерации кода.

2 Оценка транслятора может быть выполнена для конкретного проектного применения или класса применений. В последнем случае вся необходимая информация об инструментальном средстве (см. 7.4.4.4), его назначении и надлежащем использовании должна быть доступной пользователю инструментального средства. В таком случае оценка инструментального средства для конкретного проекта может быть сокращена до проверки общей пригодности инструментального средства для проекта и соответствия со спецификацией или руководством по инструментальному средству (т. е. анализируют правильное использование инструментального средства). Правильное использование инструментального средства может включать в себя дополнительные действия по проверке в рамках конкретного проекта.

3 Для оценки пригодности транслятора для выполнения своей цели согласно заданным критериям, которые должны включать в себя функциональные и нефункциональные требования, может быть использована процедура подтверждения соответствия (т. е. набор тестовых программ, корректная трансляция которых известна заранее). Основным методом подтверждения соответствия транслятора его функциональным требованиям может быть динамическое тестирование. По возможности следует использовать автоматическую процедуру тестирования.

7.4.4.11 Если требования по 7.4.4.10 не могут быть выполнены в полном объеме, то необходимо обосновать пригодность языка для реализации данной цели, а также использовать все доступные дополнительные меры, направленные на устранение любых идентифицированных недостатков языка.

7.4.4.12 Языки программирования для разработки всего ПО ПЭ СБЗС системы следует использовать в соответствии со стандартами составления программ для таких языков.

Примечание — Руководящие указания по использованию стандартов кодирования для ПО ПЭ СБЗС системы приведены в ГОСТ 34332.5.

7.4.4.13 В стандартах составления программ должны быть определены правильные методы программирования, запрещено использование небезопасных возможностей языка (например, неопределенных или непонятных особенностей языка, неструктурированных конструкций и т. п.), упрощены проверка и тестирование и определены процедуры для документирования исходного текста. В документацию, относящуюся к исходному тексту, следует включать, по меньшей мере, следующую информацию:

- юридическое лицо [например, компания, автор(ы) и т. д.];
- описание;
- входные и выходные данные;
- историю управления конфигурацией.

7.4.4.14 Если выполняется автоматическая генерация кода или применяется автоматический транслятор, то необходимо проводить оценку пригодности автоматического транслятора для разработки СБЗС системы для тех стадий ЖЦ разработки, на которых применяют средства поддержки ПО разработки.

7.4.4.15 Если средства поддержки ПО в автономном режиме классов Т2 и Т3 генерируют элементы для базовой конфигурации, то управление конфигурацией должно быть таким, чтобы гарантировать, что информация об инструментальных средствах записана в базовой конфигурации. В частности, в состав информации о средствах поддержки ПО включают:

- идентификацию средства поддержки ПО и его версии;
- идентификацию элементов базовой конфигурации, для которых использована данная версия средства поддержки ПО;

- последовательность использования средства поддержки ПО (включая параметры средства поддержки, опции и выбранные сценарии) для каждого базового элемента конфигурации.

Примечание — Цель настоящего подпункта состоит в обеспечении возможности по видоизменению базовой конфигурации.

7.4.4.16 Управление конфигурацией должно быть осуществлено таким образом, чтобы гарантировать, что инструментальные средства в классах Т2 и Т3 использованы исключительно квалифицированным специалистом.

7.4.4.17 Управление конфигурацией должно быть таким, чтобы гарантировать, что использованы только инструментальные средства, совместимые друг с другом и с ПЭ СБЗС системой.

Примечание — АС ПЭ СБЗС системы могут также наложить ограничения на совместимость ПО инструментальных средств, например: эмулятор процессора должен быть точной моделью реальной электроники процессора.

7.4.4.18 Каждая новая версия средства поддержки ПО в автономном режиме должна быть квалифицирована. Эта квалификация может опираться на доказательства, представленные для более ранней версии, при наличии достаточных доказательств того, что:

- функциональные различия (при их наличии) не будут влиять на совместимость средства поддержки ПО с остальной частью набора инструментальных средств, и новая версия может не содержать принципиально новых неизвестных отказов;
- новая версия может не содержать значительных новых неизвестных ошибок.

Примечание — Доказательство того, что новая версия может не содержать принципиально новых неизвестных отказов, может быть основано на четкой идентификации выполненных изменений, анализе действий по проверке и подтверждению соответствия, выполняемых для новой версии, и на любом актуальном опыте работы других пользователей с новой версией.

7.4.4.19 В зависимости от характера разработки ПО ответственность за соответствие требованиям 7.4.4 могут нести несколько сторон. Распределение ответственности должно быть документально оформлено во время планирования системы безопасности [см. ГОСТ 34332.2—2017 (раздел 6)].

7.4.5 Требования к детальному проектированию и разработке системы ПО

Примечания

1 Под детальным проектированием понимается разделение основных элементов архитектуры на систему программных модулей, проектирование отдельных программных модулей и их программирование. В небольших по объему приложениях проектирование программных систем и архитектуры может быть объединено.

2 Характер детального проектирования и разработки может изменяться в зависимости от характера процессов разработки программ и архитектуры ПО (см. 7.4.3). Если прикладное программирование выполняется, например, с помощью языков многозвенных логических схем и функциональных блоков, то детальное проектирование может быть рассмотрено, скорее, как конфигурирование, а не программирование. Тем не менее правильный стиль программирования состоит:

- в структурировании ПО, включая организацию модульной структуры, в которой выделяют (насколько это возможно) блоки, связанные с безопасностью;
- использовании проверок на попадание в интервал допустимых значений и других возможностей защиты от ошибок при вводе исходных данных;
- использовании ранее верифицированных программных модулей;
- применении проектных решений, которые способствуют выполнению будущих модификаций ПО.

3 При выборе соответствующих методов и средств (см. приложения А и Б) для выполнения требований настоящего пункта должны быть рассмотрены и учтены следующие свойства (см. руководство по интерпретации свойств в приложении В) и неформальные описания методов и средств в ГОСТ 34332.5—2021 (приложение Е) проектирования и разработки:

- полнота спецификации требований к ПО Э/Э/ПЭ СБЗС системы;
- корректность спецификации требований к ПО Э/Э/ПЭ СБЗС системы;
- отсутствие ошибок, допущенных в проекте;
- простота и четкость;
- предсказуемость поведения;
- поддающийся проверке и тестированию проект;
- отказоустойчивость/обнаружение неисправностей;
- отсутствие отказов по общей причине.

7.4.5.1 В зависимости от характера разработки СБ ПО ответственность за соответствие требованиям 7.4.4 могут нести несколько сторон. Распределение ответственности должно быть документально оформлено во время планирования системы безопасности [см. ГОСТ 34332.2—2017 (раздел 6)].

7.4.5.2 До начала детального проектирования должна быть подготовлена следующая информация: спецификация требований к Э/Э/ПЭ СБЗС системе, описание проекта архитектуры ПО, план подтверждения соответствия аспектов ПО Э/Э/ПЭ безопасности СБЗС системы.

7.4.5.3 СБ ПО следует разрабатывать таким образом, чтобы достигались модульность, тестируемость и способность к модификации СБЗС системы.

7.4.5.4 Дальнейшее уточнение характеристик проекта для КСБ, Э/Э/ПЭ СБЗС системы, каждого главного элемента/подсистемы в описании проекта архитектуры ПО должно быть основано на декомпозиции системы на программные модули (т. е. на спецификации проекта программной системы). Необходимо специфицировать проект каждого программного модуля и проверки этих модулей.

Примечания

1 Информация о действующих элементах ПО приведена в 7.4.2.

2 Верификация элементов ПО состоит из тестирования и анализа.

7.4.5.5 Должны быть определены соответствующие проверки интеграции программной системы, фиксирующие, что программная система соответствует требованиям к СБ ПО для Э/Э/ПЭ СБЗС системы с заданным УПБ.

7.4.6 Требования к реализации исходных текстов программ

Примечание — В соответствии с необходимым УПБ исходный код должен обладать следующими свойствами (конкретные методы представлены в приложениях А и Б; руководящие указания по интерпретации свойств — в приложении В):

- должен быть читаемым, понятным и пригодным к проверке;
- соответствовать специфицированным требованиям к проекту программного модуля (см. 7.4.5):
- отвечать специфицированным требованиям к стандартам составления программ (см. 7.4.4):
- соответствовать требованиям, определенным при планировании системы безопасности (см. раздел 6).

7.4.6.1 Каждый модуль ПО должен быть просмотрен (проверен). Если код создан с помощью автоматических средств, то он должен соответствовать требованиям 7.4.4. Если исходный код состоит из повторно используемого действующего ПО, то он должен соответствовать требованиям 7.4.2.

Примечание — Просмотр кода относится к процессам верификации (см. 7.9). Просмотр кода может быть выполнен с помощью следующего контроля кода (в порядке возрастания строгости): пользователем ПО; сквозным контролем ПО [см. ГОСТ 34332.5—2021 (пункт В.5.15 приложения В)] или формальной проверкой [см. ГОСТ 34332.5—2021 (пункт В.5.14 приложения В)].

7.4.7 Требования к тестированию программных модулей

Примечания

1 Процесс проверки того, что программный модуль корректно выполняет все требования, содержащиеся в спецификации тестирования, относится к процессам верификации (см. 7.9). Сочетание просмотра исходных текстов и тестирования программных модулей дает гарантию того, что программный модуль соответствует требованиям своей спецификации, т. е. верифицирует модуль.

При выборе соответствующих методов и средств (см. приложения А и Б) для выполнения требований настоящего пункта должны быть рассмотрены и предусмотрены следующие свойства (см. руководство по интерпретации свойств в приложении В) и неформальные описания методов и средств в ГОСТ 34332.5—2021 (приложение Е) тестирования программных модулей:

- полнота тестирования в соответствии со спецификацией проекта ПО;
- корректность тестирования в соответствии со спецификацией проекта ПО (удовлетворительное выполнение);
- воспроизводимость;
- точно определенная конфигурация тестирования.

7.4.7.1 Каждый программный модуль должен быть проверен (протестирован) в соответствии со спецификацией, разработанной при проектировании ПО (см. 7.4.5).

Примечание — Верификация состоит из тестирования и анализа.

7.4.7.2 Эти проверки должны продемонстрировать, что каждый программный модуль выполняет функции, для которых он предназначен, и не выполняет функции, которые не были для него предусмотрены.

Примечания

1 Указанное выше не означает тестирования всех комбинаций входных и выходных данных. Достаточным может быть тестирование всех классов эквивалентности или структурное тестирование. Анализ граничных значений или потоков управления может сократить количество проверок до приемлемого уровня. Программы, используемые для анализа, могут способствовать достижению более эффективного выполнения требований. Перечисленные методы приведены в ГОСТ 34332.5—2021 (приложение В).

2 Если при разработке используют формальные методы, формальные доказательства или операторы проверки условий, то область применения подобных проверок может быть сокращена. Перечисленные методы/средства приведены в ГОСТ 34332.5—2021 (приложение Б).

3 Хотя систематическая полнота безопасности обычно количественно не определяется, некоторые количественные статистические данные (например, статистическое тестирование, повышение надежности) являются приемлемыми, если удовлетворены все соответствующие условия для статистически достоверного доказательства [например, примеры в ГОСТ 34332.5—2021 (приложение Г)].

4 Если модуль настолько прост, что для него можно провести исчерпывающий тест, то это является наиболее эффективным способом демонстрации соответствия.

7.4.7.3 Результаты тестирования программных модулей должны быть документально оформлены.

7.4.7.4 Должны быть определены процедуры для коррекции при непрохождении теста.

7.4.8 Требования к тестированию интеграции программного обеспечения

Примечание — Проверка того, что интеграция СБ ПО является корректной, относится к процессам верификации (см. 7.9).

7.4.8.1 Проверки интеграции СБ ПО следует разрабатывать на этапе проектирования и разработки (см. 7.4.5).

7.4.8.2 Проверки интеграции системы ПО следует осуществлять таким образом, чтобы определять:

- разделение СБ ПО на контролируемые интегрируемые подмножества;
- контрольные примеры и контрольные данные;
- типы проверок, которые должны быть проведены;
- условия тестирования, используемые инструменты, конфигурацию и программы;
- условия, при которых проверка считается выполненной;
- процедуры, которые необходимо выполнить, если проверка дала отрицательный результат.

7.4.8.3 СБ ПО должно быть проверено в соответствии с тестами интеграции программ, определенными в спецификации тестирования интеграции системы ПО. Эти тесты должны быть выполнены таким образом, чтобы они позволили продемонстрировать, что все программные модули и программные элементы/подсистемы корректно взаимодействуют для выполнения тех функций, для которых они предназначены, и не выполняют непредусмотренных функций.

Примечания

1 Указанное выше не означает тестирования всех комбинаций входных и выходных данных. Достаточным может быть тестирование всех классов эквивалентности или структурное тестирование. Анализ граничных значений или потоков управления может сократить число проверок до приемлемого уровня. Программы, пригодные для анализа, могут помочь в достижении более быстрого выполнения требований. Перечисленные методы приведены в ГОСТ 34332.5—2021 (приложение В).

2 Если при разработке используют формальные методы, формальные доказательства или операторы проверки условий, то область применения подобных проверок может быть сокращена. Перечисленные методы приведены в ГОСТ 34332.5—2021 (приложение В).

3 Хотя систематическая полнота безопасности обычно количественно не определяется, некоторые количественные статистические данные, например статистическое тестирование, повышение надежности, являются приемлемыми, если удовлетворены все соответствующие условия для статистически достоверного доказательства [например, см. ГОСТ 34332.5—2021 (приложение Г)].

7.4.8.4 Результаты проверки интеграции СБ ПО должны быть документально оформлены; в документации должны быть сформулированы результаты проверки и указано, были ли выполнены цели и критерии проверки. Если тестирование признано неудовлетворительным, должны быть определены причины этого.

7.4.8.5 При интеграции СБ ПО все модификации должны быть объектом анализа влияния, по результатам которого должно быть определено, какие программные модули были изменены, и должна быть установлена необходимость проведения повторной верификации и проектирования.

7.5 Интеграция аппаратных средств и программного обеспечения

Примечания

1 Разработку интеграции АС и ПО осуществляют на стадиях разработки и проектирования (см. блоки 6 и 7 на рисунке 3).

2 Проверку корректности интеграции осуществляют на стадиях разработки и проектирования, а окончательную оценку и подтверждение соответствия — на стадии реализации (см. блок 13 на рисунке 3, а также см. рисунки 5 и 6).

7.5.1 Цели

7.5.1.1 Первой целью является обеспечение интеграции ПО с используемой ПЭ.

7.5.1.2 Вторая цель состоит в обеспечении объединения ПО и АС в Э/Э/ПЭ СБЗС систему, осуществления проверки их совместимости и выполнения требований назначенного УПБ Э/Э/ПЭ СБЗС системы.

Примечания

1 Проверка корректности интеграции ПО с АС в Э/Э/ПЭ СБЗС систему относится к процессам верификации (см. 7.9).

2 В зависимости от характера применения эти проверки могут быть объединены с проверками по 7.4.8.

3 Окончательную оценку и подтверждение соответствия корректности интеграции ПО с АС в Э/Э/ПЭ СБЗС систему осуществляют на объекте (см. блок 13 на рисунке 3, а также см. рисунок 6). При этом оценку и подтверждение соответствия допускается проводить автономно и/или в составе КСБ (см. рисунок 5), в зависимости от сложности системы обеспечения безопасности объекта.

7.5.1.3 Третья цель заключается в обеспечении объединения Э/Э/ПЭ СБЗС в КСБ, проверки их совместимости и выполнении требований по защите объекта (здания или сооружения).

Примечания

1 Проверка корректности интеграции Э/Э/ПЭ СБЗС систем в Э/Э/ПЭ КСБ относится к процессам верификации (см. 7.9).

2 В зависимости от характера применения эти проверки могут быть объединены с проверками по 7.4.8.

3 Окончательную оценку и подтверждение соответствия корректности интеграции Э/Э/ПЭ СБЗС систем в КСБ осуществляют на объекте (см. блок 13 на рисунке 3, а также см. рисунок 5).

7.5.2 Требования

Примечание — При выборе соответствующих методов и средств (см. приложения А и Б) для выполнения требований настоящего пункта должны быть рассмотрены и предусмотрены следующие свойства (см. руководство по интерпретации свойств в приложении В) и неформальные описания методов и средств интеграции в ГОСТ 34332.5—2021 (приложение Е):

- полнота интеграции в соответствии со спецификациями проекта;
- корректность интеграции в соответствии со спецификациями проекта (удовлетворительное выполнение);
- воспроизводимость;
- точно определенная конфигурация интеграции.

7.5.2.1 Проверки интеграции должны быть определены на этапе проектирования и разработки (см. 7.4.3). Их целью является проверка совместимости ПО и АС в ПЭ, связанной с безопасностью.

Примечание — При разработке проверок интеграции может потребоваться тесная кооперация разработчика СБ ПО с разработчиком Э/Э/ПЭ СБЗС системы.

7.5.2.2 Спецификация тестов интеграции для программируемой электроники (ПО и АС в Э/Э/ПЭ СБЗС систему и Э/Э/ПЭ СБЗС систем в КСБ) должна быть выбрана такой, чтобы определять:

- разбиение системы на уровни интеграции;
- тестовые примеры и тестовые данные;
- типы выполняемых проверок;
- условия тестирования, включая инструменты, программы поддержки и описание конфигурации;
- условия, при которых проверка считается выполненной.

7.5.2.3 В тестах интеграции программируемой электроники (АС и ПО) следует различать операции, выполняемые разработчиком на его оборудовании, и операции, требующие доступа к пользовательскому оборудованию.

7.5.2.4 В тестах интеграции программируемой электроники (АС и ПО) различают следующие процессы:

а) включение АС и ПО в целевое ПЭ оборудование;

б) интеграцию ПО в Э/Э/ПЭ СБЗС системы, т. е. добавление интерфейсов таких средств, как датчики и исполнительные устройства;

в) интеграцию УО и Э/Э/ПЭ СБЗС системы;

г) интеграцию Э/Э/ПЭ СБЗС систем в КСБ.

Примечание — Процессы по перечислениям б) — г) рассмотрены в ГОСТ 34332.2 и ГОСТ 34332.3.

7.5.2.5 ПО должно быть интегрировано с ПЭ аппаратурой, связанной с безопасностью, в соответствии со специфицированными тестами интеграции для программируемой электроники (АС и ПО).

7.5.2.6 При тестировании интеграции программируемой электроники, связанной с безопасностью (АС и ПО), все изменения в интегрированной системе должны стать объектом анализа влияния, по результатам которого должно быть определено, какие программные модули были изменены, и должна быть установлена необходимость проведения повторной верификации.

7.5.2.7 Тестовые примеры и результаты их выполнения должны быть документально оформлены для последующего анализа.

7.5.2.8 Результаты проверки интеграции программируемой электроники (АС и ПО) должны быть документально оформлены. В документации должны быть сформулированы результаты проверки, а также указано, были ли выполнены цели и критерии проверки. Если тестирование было неудовлетворительным, должны быть описаны причины произошедшего. Все модификации или изменения, являющиеся результатом тестирования, должны быть объектом анализа влияния, по результатам которого должно быть определено, какие программные элементы/модули были изменены, и должна быть установлена необходимость повторной верификации и проектирования.

7.6 Процедуры эксплуатации и модификации программного обеспечения

Примечание — В настоящем стандарте процедуры эксплуатации и модификации ПО не рассматриваются.

7.6.1 Цели

Целью требований настоящего подраздела является представление информации и процедур, касающихся ПО, необходимых для того, чтобы убедиться в том, что функциональная безопасность Э/Э/ПЭ СБЗС системы (а также КСБ) сохраняется при эксплуатации и модификациях.

7.6.2 Требования

Требования приведены в ГОСТ 34332.3—2021 (пункт 8.7.1) и в 7.8.

Примечание — В настоящем стандарте ПО (в отличие от АС) не может подвергаться техническому обслуживанию, оно модифицируется.

7.7 Подтверждение соответствия программных аспектов безопасности СБЗС системы

Примечания

1 Данная стадия относится к стадии разработки и проектирования ПО и Э/Э/ПЭ СБЗС систем (см. блоки 6 и 7 на рисунке 3).

2 Как правило, подтверждение соответствия ПО не может быть проведено отдельно от его АС и системного окружения.

7.7.1 Цели

Целью требований настоящего подраздела является гарантирование соответствия интегрированной системы специфицированным требованиям к ПО Э/Э/ПЭ СБЗС системы для заданного УПБ.

7.7.2 Требования

Примечание — При выборе соответствующих методов и средств (см. приложения А и Б) для выполнения требований настоящего подраздела должны быть рассмотрены и предусмотрены следующие свойства [см. руководство по интерпретации свойств в приложении В и неформальные описания методов и средств в ГОСТ 34332.5—2021 (приложение Е)] подтверждения соответствия функциональной безопасности Э/Э/ПЭ СБЗС системы:

- полнота подтверждения соответствия системы согласно спецификации проекта ПО;
- корректность подтверждения соответствия согласно спецификации проекта ПО (удовлетворительное выполнение);
- воспроизводимость;
- подтверждение соответствия точно определенной конфигурации.

7.7.2.1 Если для СБ ПО соответствие с требованиями установлено при планировании подтверждения соответствия для Э/Э/ПЭ СБЗС системы [см. ГОСТ 34332.5—2021 (подраздел 8.6)], то проводить повторное подтверждение соответствия не требуется.

7.7.2.2 Операции подтверждения соответствия следует проводить согласно спецификациям, разработанным при планировании подтверждения соответствия программных аспектов безопасности Э/Э/ПЭ СБЗС системы.

7.7.2.3 В зависимости от характера разработки ПО ответственность за соответствие требованиям подраздела 7.7 могут нести несколько сторон. Распределение ответственности должно быть документально оформлено во время планирования безопасности системы [см. ГОСТ 34332.2—2017 (раздел 6)].

7.7.2.4 Результаты подтверждения соответствия программных аспектов безопасности Э/Э/ПЭ СБЗС системы должны быть документально оформлены.

7.7.2.5 При проведении подтверждения соответствия программных аспектов безопасности Э/Э/ПЭ СБЗС системы для каждой функции безопасности должны быть документально оформлены следующие результаты:

- хронологический перечень операций подтверждения соответствия, который позволит восстановить последовательность действий.

Примечание — При записи результатов испытаний важно, чтобы последовательность действий могла быть восстановлена. Главное в этом требовании — восстановить последовательность действий, а не создать упорядоченный по времени/дате список документов;

- используемая версия плана подтверждения соответствия программных аспектов безопасности Э/Э/ПЭ СБЗС системы (см. подраздел 7.3);
- подтверждаемые функции безопасности (с использованием тестирования или анализа) со ссылками на план подтверждения соответствия программных аспектов безопасности Э/Э/ПЭ СБЗС системы (см. подраздел 7.3);
- используемые инструменты и оборудование, а также данные калибровки;
- результаты операций подтверждения соответствия;
- расхождения между ожидаемыми и фактическими результатами.

7.7.2.6 При наличии расхождений между ожидаемыми и фактическими результатами проводят анализ и принимают решение о продолжении подтверждения соответствия или о подготовке запроса на изменение и о возвращении к более ранней стадии ЖЦ разработки. Это решение должно быть документально оформлено как часть результатов подтверждения соответствия программных аспектов безопасности Э/Э/ПЭ СБЗС системы.

Примечание — Требования 7.7.2.2—7.7.2.5 основаны на общих требованиях ГОСТ 34332.2—2017 (подраздел 7.15).

7.7.2.7 При подтверждении соответствия программных аспектов безопасности Э/Э/ПЭ СБЗС системы необходимо выполнять следующие требования:

- основным методом подтверждения соответствия для ПО должно быть тестирование; анимацию и моделирование допускается использовать как дополнительные методы:
- прогон ПО следует проводить путем имитации:
  - входных сигналов в нормальном режиме работы,
  - предполагаемых случаев,
  - нежелательных условий, требующих вмешательства системы;
- поставщик и/или разработчик ПО (либо несколько сторон, ответственных за соответствие) должны предоставить документально оформленные результаты подтверждения соответствия программных аспектов безопасности Э/Э/ПЭ СБЗС системы и всю имеющую отношение к этой операции документацию в распоряжение разработчика системы, чтобы дать ему возможность выполнить требования ГОСТ 34332.2 и ГОСТ 34332.3.

7.7.2.8 Инструментальные средства ПО следует выбирать в соответствии с 7.4.4.

7.7.2.9 К результатам подтверждения соответствия программных аспектов безопасности Э/Э/ПЭ СБЗС системы предъявляют следующие требования:

- проверки должны показать, что все заданные требования, предъявляемые к СБ ПО (см. подраздел 7.2), выполняются правильно и что программная система не выполняет непредусмотренных функций;

- тестовые примеры и их результаты должны быть документально оформлены для последующего анализа и независимой оценки соответствия согласно требованиям УПБ [см. ГОСТ 34332.2—2017 (раздел 8)];

- в документально оформленные результаты подтверждения соответствия программных аспектов безопасности Э/Э/ПЭ СБЗС системы должны быть включены либо утверждение о том, что программа получила подтверждение соответствия, либо причины, по которым она его не получила.

7.8 Модификация программного обеспечения

7.8.1 Цель

Целью требований настоящего подраздела является внесение корректировок, усовершенствований или изменений в принятое ПО, гарантирующих сохранение стойкости к систематическим отказам ПО.

Примечание — В настоящем стандарте ПО (в отличие от АС) не может быть поддержано, так как его модифицируют.

7.8.2 Требования

Примечание — При выборе соответствующих методов и средств (см. приложения А и Б) для выполнения требований настоящего подраздела должны быть рассмотрены и предусмотрены следующие свойства [см. руководство по интерпретации свойств в приложении В и неформальные описания методов и средств в ГОСТ 34332.5—2021 (приложение Е)] модификации ПО:

- полнота модификации в соответствии с установленными требованиями;
- корректность модификации согласно требованиям к ней:
- отсутствие ошибок в проекте;
- предотвращение нежелательного поведения ПО;
- верифицируемость и тестируемость проекта;
- регрессионное тестирование и проведение проверок.

7.8.2.1 Перед выполнением какой-либо модификации ПО должны быть подготовлены процедуры модификации [см. ГОСТ 34332.2—2017 (подраздел 7.17)].

Примечание — Требования, перечисленные в 7.8.2.1—7.8.2.9, относятся в первую очередь к изменениям, выполняемым на стадии эксплуатации ПО. Требования по 7.8.2.1—7.8.2.9 также могут быть применены на стадии интеграции программируемой электроники, а также на стадиях установки и ввода в эксплуатацию всей системы безопасности [см. ГОСТ 34332.2—2017 (подраздел 7.12)].

7.8.2.2 Процесс модификации может быть начат только после появления запроса на санкционированную модификацию ПО в рамках процедур, определенных на этапе планирования системы безопасности (см. раздел 6), в котором приведена подробная информация:

- об опасностях, на которые могут повлиять изменения;
- о предлагаемых изменениях;
- причинах изменений.

Примечание — Причины появления запроса на модификацию могут быть, например, связаны:

- с тем, что функциональная безопасность оказалась ниже той, которая определена в спецификациях;
- накопленными данными о систематических отказах;
- появлением нового или изменением действующего законодательства, относящегося к безопасности;
- модификацией УО или способа его использования;
- модификацией общих требований к системе безопасности;
- анализом характеристик эксплуатации и технического обслуживания, который показывает, что эти характеристики имеют значения ниже запланированных;
- текущим аудитом функциональной безопасности систем.

7.8.2.3 Должен быть выполнен анализ влияния предлагаемых модификаций ПО на функциональную безопасность Э/Э/ПЭ СБЗС системы для определения:

- необходимости анализа рисков;
- какие именно этапы ЖЦ ПО Э/Э/ПЭ СБЗС системы следует повторить.

7.8.2.4 Результаты анализа влияния, полученные в соответствии с 7.8.2.3, должны быть документально оформлены.

7.8.2.5 Все модификации, оказывающие влияние на функциональную безопасность Э/Э/ПЭ СБЗС системы (включая КСБ), следует осуществлять начиная с возврата на соответствующую(ий) стадию (этап) ЖЦ СБ ПО. Все последующие стадии (этапы) следует выполнять согласно процедурам, опре-

деленным для конкретных стадий (этапов) в соответствии с требованиями настоящего стандарта. При планировании системы безопасности (см. раздел 6) должны быть подробно описаны все последующие процессы.

Примечание — Может потребоваться проведение полного анализа опасностей и рисков, в результате которого может появиться потребность в иных уровнях полноты безопасности, чем те, которые определены для функций безопасности, реализуемых Э/Э/ПЭ СБЗС системами.

7.8.2.6 Планирование системы безопасности для модификации СБ ПО выполняют в соответствии с требованиями, представленными в ГОСТ 34332.2—2017 (раздел 6), в частности с требованиями:

- к идентификации персонала и определению требований к его квалификации;
- подробной спецификации модификации;
- планированию верификации;
- определению области применения процедур повторного подтверждения соответствия и тестирования модификации в той степени, в которой этого требует УПБ.

Примечание — В зависимости от характера применения может оказаться необходимым участие экспертов в области данного применения.

7.8.2.7 Модификация должна быть проведена в соответствии с разработанным планом.

7.8.2.8 Все модификации должны быть подробно задокументированы, включая:

- запрос на модификацию/корректировку;
- результаты анализа влияния на предлагаемые модификации ПО на функциональную безопасность и принятые решения с их обоснованием;
- сведения об изменениях конфигурации ПО;
- отклонения от нормальной работы и нормальных условий работы;
- документы, связанные с процессами модификации.

7.8.2.9 Информация о деталях всех проведенных модификаций должна быть документально оформлена. В документацию включают данные и результаты повторной верификации и повторного подтверждения соответствия.

7.8.2.10 Оценка необходимых модификаций или корректировок должна зависеть от результатов анализа влияния модификаций и стойкости к систематическим отказам ПО.

7.9 Верификация программного обеспечения

7.9.1 Цель

Целью требований настоящего подраздела являются проверка и оценка в соответствии с требуемым УПБ результатов, полученных на заданной стадии ЖЦ ПО Э/Э/ПЭ СБЗС системы, а также гарантирование того, что эти результаты признаны корректными и соответствуют исходной информации для определенной стадии.

Примечания

1 Настоящий подраздел учитывает базовые аспекты верификации, которые являются общими для нескольких стадий ЖЦ системы безопасности. Настоящий подраздел не предъявляет дополнительных требований к элементам проверки при верификации в 7.4.7 (проверка программных модулей), 7.4.8 (интеграция ПО) и подразделе 7.5 (интеграция программируемой электроники), которые представляют собой процессы верификации. Данный подраздел не требует также дополнительной верификации для процессов подтверждения соответствия ПО (см. подраздел 7.7), так как подтверждение соответствия в настоящем стандарте определено как демонстрация соответствия спецификации требований к системе безопасности. Проверку того, является ли корректной спецификация, проводят специалисты по предметным областям.

2 В зависимости от архитектуры ПО ответственность за проведение верификации ПО может быть распределена между всеми организациями, вовлеченными в разработку и модификацию ПО.

7.9.2 Требования

Примечание — При выборе соответствующих методов и средств (см. приложения А и Б) для выполнения требований настоящего подраздела должны быть рассмотрены и предусмотрены следующие свойства [см. руководство по интерпретации свойств в приложении В и неформальные описания методов и средств в ГОСТ 34332.5—2021 (приложение Е)] верификации данных:

- полнота верификации в соответствии с предыдущей стадией;
- корректность верификации в соответствии с предыдущей стадией (удовлетворительное выполнение);
- воспроизводимость;
- верификация точно определенной конфигурации.

7.9.2.1 Верификацию ПО для каждой стадии ЖЦ ПО Э/Э/ПЭ СБЗС системы следует планировать (см. подраздел 7.4) одновременно с разработкой; вся информация, относящаяся к этому вопросу, должна быть документально оформлена.

7.9.2.2 Планирование верификации ПО должно касаться критериев, методов и инструментария, используемого при верификации. В ходе планирования должны быть рассмотрены и предусмотрены:

- оценка требований полноты безопасности;
- выбор и документирование стратегии, процессов и методов верификации;
- выбор и использование инструментов верификации (тестовая программа, специальные программные средства для тестирования, имитаторы ввода/вывода и т. п.);
- оценка результатов верификации;
- исправления, которые должны быть внесены.

7.9.2.3 Верификация ПО должна быть проведена в соответствии с планом.

Примечание — Выбор методов и средств, предназначенных для верификации, а также степень независимости процессов верификации определены рядом факторов и могут быть указаны в стандартах для прикладных отраслей. К числу таких факторов относятся, например:

- размер проекта;
- степень сложности;
- степень новизны проекта;
- степень новизны технологии.

7.9.2.4 Должны быть документально оформлены свидетельства того, что верифицируемая стадия завершена удовлетворительно во всех отношениях.

7.9.2.5 В документацию, составляемую после каждой верификации, включают идентификацию параметров, подлежащих верификации, и информацию, необходимую для выполнения верификации.

Примечание — Информация, необходимая для проведения верификации, включает в себя (но не ограничивается): входную информацию, полученную от предыдущей стадии ЖЦ, стандарты проектирования, стандарты кодирования и используемые инструментальные средства;

а также перечень несоответствий.

Примечание — Например, несоответствия возможны в программных модулях, структурах данных и алгоритмах, неудовлетворительно адаптированных к поставленной задаче.

7.9.2.6 Вся существенная информация, относящаяся к стадии *N* ЖЦ ПО системы безопасности, необходимая для правильного выполнения следующей стадии *N* + 1, должна быть доступна и верифицирована. К выходной информации стадии *N* относятся:

а) информация об адекватности спецификации, описания проекта либо исходного текста программ, разработанных в процессе стадии *N*:

1) функциональность,

2) полнота безопасности, характеристики и другие требования к планированию системы безопасности (см. раздел 6),

3) требования понятности для коллектива разработчиков,

4) тестируемость для дальнейшей верификации,

5) безопасность модификации, допускающей дальнейшее развитие;

б) информация об адекватности планирования подтверждения соответствия и/или тестов, установленных для стадии *N,* определению и описанию проекта стадии *N*;

в) результаты проверки несовместимости между:

1) тестами, определенными для стадии *N*, и тестами для предыдущей стадии *N* – 1,

2) выходными данными стадии *N.*

7.9.2.7 В соответствии с выбором ЖЦ разработки ПО (см. подраздел 7.1) должна быть выполнена верификация:

а) требований к ПО системы безопасности;

б) архитектуры ПО;

в) проекта системы ПО;

г) проектов программных модулей;

д) исходных текстов программ;

е) данных:

1) тестирование программных модулей (см. 7.4 7),

2) тестирование интеграции ПО (см. 7.4.8),

3) тестирование интеграции программируемой электроники (см. подраздел 7.5),

4) подтверждение соответствия аспектов ПО системы безопасности (см. подраздел 7.7).

Примечание — Требования по перечислениям а) — д) представлены ниже.

7.9.2.8 При верификации требований к ПО системы безопасности (после определения требований к ПО системы безопасности и перед началом следующей стадии проектирования и разработки ПО) следует проверить:

а) соответствие спецификации требований к ПО системы безопасности относительно спецификации требований к Э/ЭУПЭ СБЗС системе [см. ГОСТ 34332.3—2017 (подраздел 8.2)] в отношении функциональности, полноты безопасности, характеристик и других требований к планированию системы безопасности;

б) соответствие плана подтверждения соответствия программных аспектов безопасности Э/Э/ПЭ СБЗС системы спецификации требований к безопасности ПО;

в) наличие несовместимости между:

1) спецификацией требований к безопасности ПО и спецификацией требований к безопасности Э/Э/ПЭ СБЗС системы [см. ГОСТ 34332.3—2017 (подраздел 8.2)];

2) спецификацией требований к ПО системы безопасности и планом подтверждения соответствия программных аспектов безопасности Э/Э/ПЭ СБЗС системы.

7.9.2.9 При верификации архитектуры ПО (после того, как выполнен проект архитектуры ПО) следует проверить:

а) соответствие проекта архитектуры ПО спецификации требований к безопасности ПО;

б) адекватность проверок интеграции, установленных в проекте архитектуры ПО;

в) адекватность атрибутов каждого основного элемента/подсистемы по отношению:

1) к реализуемости требуемых характеристик безопасности,

2) возможности проверки при последующей верификации,

3) пониманию задач персоналом, выполняющим разработку и верификацию,

4) безопасной модификации при дальнейшем развитии ПО;

г) наличие несовместимости между:

1) описанием проекта архитектуры ПО и спецификацией требований к ПО,

2) описанием проекта архитектуры ПО и тестами интеграции архитектуры ПО,

3) тестами интеграции проекта архитектуры ПО и планом подтверждения соответствия программных аспектов безопасности Э/Э/ПЭ СБЗС системы.

7.9.2.10 При верификации проекта системы ПО (после завершения проектирования системы ПО) следует проверить:

а) соответствие проекта системы ПО (см. 7.4.5) проекту архитектуры ПО;

б) соответствие тестов интеграции системы ПО (см. 7.4.5) проекту системы ПО (см. 7.4.5);

в) адекватность спецификации атрибутов каждого основного элемента проекта системы ПО (см. 7.4.5) по отношению:

1) к реализуемости требуемых характеристик системы безопасности,

2) возможности проверки при последующей верификации,

3) пониманию задач персоналом, выполняющим разработку и верификацию,

4) модификации системы безопасности, позволяющей выполнять дальнейшее развитие программы.

7.9.2.11 При верификации проекта модулей ПО (после того, как выполнен проект каждого программного модуля) следует проверить:

а) соответствие спецификации проекта программного модуля (см. 7.4.5) относительно спецификации проекта системы ПО (см. 7.4.5);

б) адекватность спецификации проверок каждого программного модуля (см. 7.4.5) относительно спецификации проекта программного модуля (см. 7.4.5);

в) адекватность атрибутов каждого программного модуля по отношению:

1) к реализуемости требуемых характеристик системы безопасности (см. спецификацию требований к ПО системы безопасности),

2) возможности проверки при последующей верификации,

3) пониманию задач персоналом, выполняющим разработку и верификацию,

4) модификации системы безопасности, позволяющей выполнять дальнейшее развитие программы;

г) наличие несовместимости между:

1) спецификацией проекта программного модуля (см. 7.4.5) и спецификацией проекта системы ПО (см. 7.4.5),

2) спецификацией проекта каждого программного модуля (см. 7.4.5) и спецификацией проверок этого программного модуля (см. 7.4.5),

3) спецификацией проверок программных модулей (см. 7.4.5) и спецификацией проверок интеграции системы ПО (см. 7.4.5).

7.9.2.12 В процессе верификации исходный текст должен пройти проверку статическими методами для того, чтобы гарантировать соответствие спецификации проекта программных модулей (см. 7.4.5) необходимым стандартам кодирования (см. 7.4.4) и плану подтверждения соответствия аспектов ПО безопасности Э/Э/ПЭ СБЗС системы.

Примечание — На ранних стадиях ЖЦ ПО Э/Э/ПЭ СБЗС системы верификация является статической (например, изучение, просмотр, формальная проверка и т. п.). При верификации исходного текста используют такие методы, как просмотр и прогон ПО. Сочетание результатов верификации исходных текстов и проверок ПО гарантирует, что каждый программный модуль будет соответствовать своей спецификации. С этого момента тестирование становится основным средством проверки.

7.9.2.13 При верификации данных должны быть проверены:

а) структуры данных;

б) прикладные данные:

1) на соответствие структурам данных,

2) полноту в отношении требований применения,

3) совместимость с системным ПО (например, для организации последовательности, управления во время выполнения и др.),

4) правильность значений данных;

в) все эксплуатационные параметры по отношению к требованиям применения;

г) все промышленные интерфейсы и соответствующее ПО [датчики и исполнительные устройства, а также автономные интерфейсы (см. 7.2.2.12)]:

1) на выявление предполагаемых отказов интерфейса,

2) устойчивость по отношению к предполагаемым отказам интерфейса;

д) все коммуникационные интерфейсы и соответствующее ПО на наличие адекватного уровня:

1) обнаружения ошибок,

2) защиты от повреждения,

3) подтверждения соответствия данных.

7.9.2.14 Верификация временных характеристик должна быть проверена на предсказуемость поведения во времени.

Примечание — Поведение во времени может быть определено производительностью средств, наличием ресурсов, временем реакции, временем выполнения в наихудшем случае, случаями переполнения памяти, случайными зависаниями, временем работы системы.

8 Оценка функциональной безопасности

Примечание — При выборе соответствующих методов и средств (см. приложения А и Б) для выполнения требований настоящего раздела должны быть рассмотрены и предусмотрены следующие свойства (см. руководство по интерпретации свойств в приложении В) и неформальные описания методов и средств в ГОСТ 34332.5 (приложение Е) оценки функциональной безопасности:

- полнота оценки функциональной безопасности в соответствии с требованиями настоящего стандарта:
- корректность оценки функциональной безопасности в соответствии с проектными спецификациями (удовлетворительное завершение);
- доступное для анализа решение выявленных проблем;
- возможность модификации оценки функциональной безопасности после изменения проекта без необходимости проведения тщательной переработки оценки;
- воспроизводимость;
- своевременность;
- точно определенная конфигурация.

8.1 Цели и требования к оценке функциональной безопасности СБ ПО — по ГОСТ 34332.2—2017 (раздел 8).

8.2 Если иное не установлено в стандартах на область применения, то минимальный уровень независимости для лиц, выполняющих оценку функциональной безопасности, должен быть определен по ГОСТ 34332.2—2017 (раздел 8).

8.3 При оценке функциональной безопасности могут быть использованы результаты процессов, приведенных в таблице А.10 приложения А.

Примечание — Выбор методов, приведенных в приложениях А и Б, не гарантирует, что будет достигнута необходимая полнота безопасности (см. 7.1.2.7). Лицо, проводящее оценку функциональной безопасности, должно также рассмотреть и оценить:

- совместимость и взаимное дополнение выбранных методов, языков и инструментальных средств для всего цикла разработки:
- полноту понимания разработчиками методов, языков и инструментальных средств, которые они используют;
- степень адаптированности методов, языков и инструментальных средств к тем проблемам, с которыми приходится сталкиваться при разработке.

Приложение А
(обязательное)

Руководство по выбору методов и средств

Некоторые из подразделов настоящего стандарта имеют ассоциированные с ними таблицы, например: подраздел 7.2 (спецификация требований к ПО системы безопасности) соотносится с данными таблицы А.1. Более подробные таблицы, содержащиеся в приложении Б, раскрывают содержание некоторых элементов таблиц приложения А (в частности, таблица Б.2 детализирует содержание динамического анализа и тестирования на основе таблицы А.5).

Обзор методов и средств, упоминаемых в приложениях А и Б, приведен в ГОСТ 34332.5. Для каждого из них представлены рекомендации по УПБ, изменяющемуся от 1 до 4, которые обозначены следующим образом:

ОР — метод или средство, особо рекомендованный(ое) к применению для данного УПБ. Если метод или средство не используют, то на стадии (этапе) планирования СБ системы этому должно быть дано подробное объяснение со ссылкой на приложение В, и это объяснение должно быть согласовано с техническим экспертом;

Р — метод или средство, рекомендованный(ое) к применению для данного УПБ, но степень обязательности рекомендации ниже, чем в случае рекомендации ОР;

«- -» — для данного метода или средства отсутствует рекомендация относительно его применения;

НР — метод или средство, не рекомендованный(ое) к применению для данного УПБ. Если метод или средство применяют, то на стадии (этапе) планирования системы безопасности этому должно быть дано подробное обоснование со ссылкой на приложение В, которое следует согласовать с техническим экспертом.

Методы и средства следует выбирать в соответствии с установленным УПБ. Альтернативные или эквивалентные методы и средства обозначают буквой, следующей за номером. Следует применять только один(но) из альтернативных или эквивалентных методов и средств.

Примечание — Данный подход действует в отношении методов/средств, приведенных в таблицах А.1 — А.4.

Могут быть применены другие методы и средства при условии, что они соответствуют требованиям и целям стадии разработки ПО. Руководящие указания по выбору методов приведены в приложении В.

Ранжирование методов и средств связано с концепцией эффективности, используемой в ГОСТ 34332.3. При равных условиях методы, имеющие ранг «ОР», будут более эффективными в предотвращении внесения систематических ошибок в процессе разработки ПО либо архитектуры программ при выявлении ошибок, оставшихся необнаруженными на стадии выполнения, по сравнению с методами, имеющими ранг «Р».

При большом числе факторов, влияющих на стойкость к систематическим отказам ПО, невозможно представить алгоритм, определяющий такую комбинацию методов и средств, которая была бы корректной для любого заданного применения. Тем не менее в приложении В приведено руководство по выбору конкретных методов для достижения стойкости к систематическим отказам ПО.

В случае конкретного применения соответствующая комбинация методов или средств должна быть сформулирована при планировании системы безопасности, при этом методы и средства должны быть применены, если примечания к таблицам не содержат иных требований.

Таблица А.1 — Спецификация требований к связанному с безопасностью программному обеспечению (см. подраздел 7.2)

Метод/средство*	Структурный элемент ГОСТ 34332.5—2021	УПБ 1	УПБ 2	УПБ 3	УПБ 4
1 Альтернативные методы					
1а Полуформальные методы	Таблица Б.7	Р	Р	ОР	ОР
1б Формальные методы	Б.2.2, В.2.4	- -	Р	Р	ОР
2 Прямая прослеживаемость между требованиями к системе безопасности и требованиями к программному обеспечению системы безопасности	В.2.11	Р	Р	ОР	ОР
3 Обратная прослеживаемость между требованиями к системе безопасности и предполагаемыми потребностями безопасности	В.2.11	Р	Р	ОР	ОР

Окончание таблицы А.1

Метод/средство*	Структурный элемент ГОСТ 34332.5—2021	УПБ 1	УПБ 2	УПБ 3	УПБ 4
4 Компьютерные средства разработки спецификаций для поддержки перечисленных выше подходящих методов/средств	Б.2.4	Р	Р	ОР	ОР
* Альтернативные или эквивалентные методы/средства обозначаются буквами, следующими за номером. Следует применять только один (одно) из альтернативных или эквивалентных методов/средств. Выбор альтернативных методов/средств должен быть обоснован в соответствии со свойствами, приведенными в приложении В, предпочтительно для каждого применения. Примечание — Согласно спецификации требований к ПО Э/Э/ПЭ СБЗС системы на постоянной основе должно быть приведено описание задачи на естественном языке и охарактеризовано использование необходимой системы математических обозначений, отражающих содержание соответствующего приложения.					

Таблица А.2 — Проектирование и разработка программного обеспечения. Проектирование архитектуры ПО (см. 7.4.3)

Метод/средство*	Структурный элемент ГОСТ 34332.5—2021	УПБ 1	УПБ 2	УПБ 3	УПБ 4
1 Обнаружение ошибок	В.3.1	- -	Р	ОР	ОР
2 Коды обнаружения ошибок	В.3.2	Р	Р	Р	ОР
3 Альтернативные методы					
3а Программирование с проверкой ошибок	В.3.3	Р	Р	Р	ОР
3б Методы контроля (при реализации процесса контроля и контролируемой функции на одном компьютере обеспечивается их независимость)	В.3.4	- -	Р	Р	- -
3в Методы контроля (реализация процесса контроля и контролируемой функции на разных компьютерах)	В.3.4	- -	Р	Р	ОР
3г Многовариантное программирование, реализующее одну спецификацию требований к ПО Э/Э/ПЭ СБЗС системы	В.3.5	- -	- -	- -	Р
3д Функционально многовариантное программирование, реализующее различные спецификации требований к ПО Э/Э/ПЭ СБЗС системы	В.3.5	- -	- -	Р	ОР
3е Восстановление предыдущего состояния	В.3.6	Р	Р	- -	ОР
3ж Проектирование ПО, не сохраняющего состояние (или проектирование ПО, сохраняющего ограниченное описание состояния)	В.2.12	- -	- -	Р	ОР
4 Альтернативные методы					
4а Механизмы повторных попыток парирования сбоя	В.3.7	Р	Р	- -	- -
4б Постепенное отключение функций	В.3.8	Р	Р	ОР	ОР
5 Исправление ошибок методами искусственного интеллекта	В.3.9	- -	ОР	ОР	ОР
6 Динамическая реконфигурация	В.3.10	- -	ОР	ОР	ОР
7 Модульный подход	Таблица Б.9	ОР	ОР	ОР	ОР
8 Использование доверительных/проверенных элементов ПО (при наличии)	В.2.10	Р	ОР	ОР	ОР

Окончание таблицы А.2

Метод/средство*	Структурный элемент ГОСТ 34332.5—2021	УПБ 1	УПБ 2	УПБ 3	УПБ 4
9 Прямая прослеживаемость между спецификацией требований к ПО системы безопасности и архитектурой ПО	В.2.11	Р	Р	ОР	ОР
10 Обратная прослеживаемость между спецификацией требований к ПО Э/Э/ПЭ СБЗС системы и архитектурой ПО	В.2.11	Р	Р	ОР	ОР
11 Альтернативные методы					
11а Структурные методы**	В.2.1	ОР	ОР	ОР	ОР
11б Полуформальные методы**	Таблица Б.7	Р	Р	ОР	ОР
11в Формальные методы проектирования и усовершенствования**	Б.2.2, В.2.4	- -	Р	Р	ОР
11г Автоматическая генерация ПО	В.4.6	Р	Р	Р	Р
12 Автоматизированные средства разработки спецификаций и проектирования	Б.2.4	Р	Р	ОР	ОР
13 Альтернативные методы					
13а Циклическое поведение с гарантированным максимальным временем цикла	В.3.11	Р	ОР	ОР	ОР
13б Архитектура с временным распределением	В.3.11	Р	ОР	ОР	ОР
13в Управление событиями с гарантированным максимальным временем реакции	В.3.11	Р	ОР	ОР	- -
14 Статическое выделение ресурсов	В.2.6.3	- -	Р	ОР	ОР
15 Статическая синхронизация доступа к разделяемым ресурсам	В.2.6.3	- -	- -	Р	ОР

* Альтернативные или эквивалентные методы/средства обозначаются буквами, следующими за номером. Следует применять только один (одно) из альтернативных или эквивалентных методов/средств. Выбор альтернативных методов/средств должен быть обоснован в соответствии со свойствами, приведенными в приложении В, предпочтительно для каждого применения.

** Из группы методов 11 «Структурные методы» следует применять метод 11а в том случае, если метод 11б не подходит для предметной области с УПБ 3 + УПБ 4.

Примечания

1 Одни методы в данной таблице посвящены концепциям проектирования, другие — тому, как проект представляется.

2 Приведенные в настоящей таблице средства, касающиеся устойчивости к ошибкам (контроль ошибок), должны быть рассмотрены совместно с требованиями по ГОСТ 34332.3 к архитектуре и контролю ошибок для АС ПЭ устройств.

3 См. таблицу В.2 приложения В.

4 Группу методов/средств 13а — 13в применяют только к системам и ПО с требованиями к синхронизации Э/Э/ПЭ СБЗС системы.

5 Метод/средство 14: использование динамических объектов (например, при работе со стеком или с неупорядоченным массивом) может быть обязательным в отношении доступной памяти и времени выполнения. Метод 14 не должен быть применен, если используется компилятор, который гарантирует, что:

- перед выполнением будет выделено достаточно памяти для всех динамических переменных и объектов или в случае ошибки при выделении памяти будет достигнуто безопасное состояние;
- время отклика соответствует заданным требованиям.

6 Метод/средство 4а: устранение неисправностей с помощью повторных попыток часто подходит при любом УПБ, но число попыток должно быть ограничено.

Таблица А.3 — Проектирование и разработка программного обеспечения. Инструменты поддержки и языки программирования (см. 7.4.4)

Метод/средство*	Структурный элемент ГОСТ 34332.5—2021	УПБ 1	УПБ 2	УПБ 3	УПБ 4
1 Соответствующий язык программирования	В.4.5	ОР	ОР	ОР	ОР
2 Сильно типизированный язык программирования	В.4.1	ОР	ОР	ОР	ОР
3 Подмножество языков	В.4.2	- -	- -	ОР	ОР
4 Альтернативные методы					
4а Сертифицированные инструменты и сертифицированные переводчики	В.4.3	Р	ОР	ОР	ОР
4б Инструменты и переводчики: высокая степень исполнения в использовании	В.4.4	ОР	ОР	ОР	ОР

* Альтернативные или эквивалентные методы/средства обозначаются буквами, следующими за номером. Следует применять только один (одно) из альтернативных или эквивалентных методов/средств. Выбор альтернативных методов/средств должен быть обоснован в соответствии со свойствами, приведенными в приложении В, предпочтительно для каждого применения.

Примечание — См. таблицу В.3 приложения В.

Таблица А.4 — Проектирование и разработка ПО: детальное проектирование (см. 7.4.5 и 7.4.6) (включает в себя проектирование системы ПО, проектирование модуля ПО и кодирование)

Метод/средство	Структурный элемент ГОСТ 34332.5—2021	УПБ 1	УПБ 2	УПБ 3	УПБ 4
1 Альтернативные методы					
1а Методы структурных диаграмм*	В.2.1	ОР	ОР	ОР	ОР
1б Полуформальные методы*	Таблица Б.7	Р	Р	ОР	ОР
1в Формальные методы проектирования и усовершенствования*	Б.2.2, В.2.4	- -	Р	Р	ОР
2 Средства автоматизированного проектирования	Б.3.5	Р	Р	ОР	ОР
3 Программирование с защитой	В.2.5	- -	Р	ОР	ОР
4 Модульный подход	Таблица Б.9	ОР	ОР	ОР	ОР
5 Стандарты проектирования и кодирования	В.2.6, таблица Б.1	Р	ОР	ОР	ОР
6 Структурное программирование	В.2.7	ОР	ОР	ОР	ОР
7 Использование доверительных/проверенных программных модулей и компонентов (по возможности)	В.2.10	Р	ОР	ОР	ОР
8 Прямая прослеживаемость между спецификацией требований к ПО системы безопасности и проектом ПО	В.2.11	Р	Р	ОР	ОР

* Альтернативные или эквивалентные методы/средства обозначаются буквами, следующими за номером. Следует применять только один (одно) из альтернативных или эквивалентных методов/средств. Выбор альтернативных методов/средств должен быть обоснован в соответствии со свойствами, приведенными в приложении В, предпочтительно для каждого применения. Из группы 1 «Альтернативные методы» следует использовать метод 1а, только если метод 1б не подходит для предметной области с УПБ 3 и УПБ 4.

Примечания

1 См. таблицу В.4 приложения В.

2 Пригодность разработки объектно-ориентированного ПО для Э/Э/ПЭ систем находится на стадии обсуждения. Руководящие указания по использованию объектно-ориентированного подхода при разработке архитектуры ПО и при проектировании приведены в ГОСТ 34332.5—2021 (приложение Ж).

Таблица А.5 — Разработка и проектирование ПО (тестирование и интеграция программного модуля)

Метод/средство*	Структурный элемент ГОСТ 34332.5—2021	УПБ 1	УПБ 2	УПБ 3	УПБ 4
1 Вероятностное тестирование	В.5.1	- -	Р	Р	ОР
2 Динамический анализ и тестирование	Б.6.5, таблица Б.2	Р	ОР	ОР	ОР
3 Регистрация и анализ данных	В.5.2	ОР	ОР	ОР	ОР
4 Функциональное тестирование и тестирование методом «черного ящика»	Б.5.1, Б.5.2, таблица Б.3	ОР	ОР	ОР	ОР
5 Тестирование рабочих характеристик	Таблица Б.6	Р	Р	ОР	ОР
6 Тестирование, основанное на модели	В.5.27	Р	Р	ОР	ОР
7 Тестирование интерфейса	В.5.3	Р	Р	ОР	ОР
8 Управление тестированием и средства автоматизации	В.4.7	Р	ОР	ОР	ОР
9 Прямая прослеживаемость между спецификацией проекта ПО и спецификациями тестирования модуля и интеграции	В.2.11	Р	Р	ОР	ОР
10 Формальная верификация	В.5.12	- -	- -	Р	Р

* Методы/средства следует выбирать в соответствии с УПБ.

Примечания

1 Тестирование программных модулей и интеграции относится к процессам верификации (см. таблицу А.9).

2 См. таблицу В.5 приложения В.

3 Формальная проверка может сократить размер и объем занимаемой памяти модуля, поэтому необходимо тестирование интеграции.

Таблица А.6 — Интеграция программируемых электронных устройств (программное обеспечение и аппаратные средства) (см. 7.5)

Метод/средство*	Структурный элемент ГОСТ 34332.5—2021	УПБ 1	УПБ 2	УПБ 3	УПБ 4
1 Функциональное тестирование и тестирование методом «черного ящика»	Б.5.1, Б.5.2, таблица Б.3	ОР	ОР	ОР	ОР
2 Тестирование рабочих характеристик	Таблица Б.6	Р	Р	ОР	ОР
3 Прямая прослеживаемость между требованиями проекта системы и ПО к интеграции ПО и АС и спецификациями тестирований интеграции ПО и АС	В.2.11	Р	Р	ОР	ОР

* Методы/средства следует выбирать в соответствии с УПБ.

Примечания

1 Интеграция ПЭ устройств относится к процессам верификации (см. таблицу А.9).

2 См. таблицу В.6 приложения В.

Таблица А.7 — Аспекты валидации программного обеспечения (см. 7.7)

Метод/средство*	Структурный элемент ГОСТ 34332.5—2021	УПБ 1	УПБ 2	УПБ 3	УПБ 4
1 Вероятностное тестирование	В.5.1	- -	Р	Р	ОР
2 Моделирование процесса	В.5.18	Р	Р	ОР	ОР
3 Моделирование	Таблица Б.5	Р	Р	ОР	ОР

Окончание таблицы А.7

Метод/средство*	Структурный элемент ГОСТ 34332.5—2021	УПБ 1	УПБ 2	УПБ 3	УПБ 4
4 Функциональное тестирование и тестирование методом «черного ящика»	Б.5.1. Б.5.2, таблица Б.3	ОР	ОР	ОР	ОР
5 Прямая прослеживаемость между спецификацией требований к ПО системы безопасности и планом подтверждения соответствия ПО системы безопасности	В.2.11	Р	Р	ОР	ОР
6 Обратная прослеживаемость между планом подтверждения соответствия ПО Э/Э/ПЭ СБЗС системы и спецификацией требований к ПО системы	В.2.11	Р	Р	ОР	ОР
* Методы/средства следует выбирать в соответствии с УПБ. Примечание — См. таблицу В.6 приложения В.					

Таблица А.8 — Модификация программного обеспечения (см. 7.8)

Метод/средство*	Структурный элемент ГОСТ 34332.5—2021	УПБ 1	УПБ 2	УПБ 3	УПБ 4
1 Анализ влияния	В.5.23	ОР	ОР	ОР	ОР
2 Повторная верификация измененных программных модулей	В.5.23	ОР	ОР	ОР	ОР
3 Повторная верификация программных модулей, на которые оказывают влияние изменения в других модулях	В.5.23	Р	ОР	ОР	ОР
4 Альтернативные методы					
4а Повторное подтверждение соответствия системы в целом	Таблица А.7	- -	Р	ОР	ОР
4б Регрессионное подтверждение соответствия	В.5.25	Р	ОР	ОР	ОР
5 Управление конфигурацией ПО	В.5.24	ОР	ОР	ОР	ОР
6 Регистрация и анализ данных	В.5.2	ОР	ОР	ОР	ОР
7 Прямая прослеживаемость между спецификацией требований к ПО Э/Э/ПЭ СБЗС системы и планом модификации ПО (включая повторные верификацию и подтверждение соответствия)	В.2.11	Р	Р	ОР	ОР
8 Обратная прослеживаемость между планом модификации ПО (включая повторную верификацию и подтверждение соответствия) и спецификацией требований к ПО Э/Э/ПЭ СБЗС системы	В.2.11	Р	Р	ОР	ОР
* Выбор альтернативных методов/средств должен быть обоснован в соответствии со свойствами, приведенными в приложении В, предпочтительно для каждого применения. Альтернативные или эквивалентные методы/средства обозначаются буквами, следующими за номером. Следует применять только один (одно) из альтернативных или эквивалентных методов/средств. Примечания 1 См. таблицу В.7 приложения В. 2 Методы группы 1 «Анализ влияния» являются необходимой частью метода 4б «Регрессионное подтверждение соответствия» (см. ГОСТ 34332.5).					

Таблица А.9 — Верификация программного обеспечения (см. 7.9)

Метод/средство*	Структурный элемент ГОСТ 34332.5—2021	УПБ 1	УПБ 2	УПБ 3	УПБ 4
1 Формальное доказательство	В.5.12	- -	Р	Р	ОР
2 Анимация спецификации и тестирования	В.5.26	Р	Р	Р	Р
3 Статический анализ	Б.6.4, таблица Б.8	Р	ОР	ОР	ОР
4 Динамический анализ и тестирование	Б.6.5, таблица Б.2	Р	ОР	ОР	ОР
5 Прямая прослеживаемость между спецификацией проекта ПО и планом верификации (включая верификацию данных) ПО	В.2.11	Р	Р	ОР	ОР
6 Обратная прослеживаемость между планом верификации (включая верификацию данных) ПО и спецификацией проекта ПО	В.2.11	Р	Р	ОР	ОР
7 Численный анализ в автономном режиме	В.2.13	Р	Р	ОР	ОР
8 Тестирование и интеграция программного модуля	См. таблицу А.5				
9 Проверка интеграции программируемых электронных устройств	См. таблицу А.6				
10 Тестирование программной системы (подтверждение соответствия)	См. таблицу А.7				

* Методы/средства следует выбирать в соответствии с УПБ.

Примечания

1 Для удобства восприятия материала все процессы, связанные с верификацией, объединены в настоящей таблице, что, однако, не является дополнительными требованиями к элементам верификации, связанными с динамическим тестированием и приведенными в таблицах А.5 и А.6, которые относятся к процессам верификации. Настоящая таблица также не требует проведения верификационного тестирования в дополнение к подтверждению соответствия ПО (см. таблицу Б.7 приложения Б), которая в настоящем стандарте отображает соответствие спецификации требований к Э/Э/ПЭ СБЗС системе (конечную верификацию).

2 Проверка включает положения ГОСТ 34332.1 — ГОСТ 34332.3, поэтому первая проверка системы безопасности связана с более ранними спецификациями уровня системы.

3 Требования к верификации включены в ГОСТ 34332.2 — ГОСТ 34332.4. Следовательно, первая верификация СБЗС системы относится к ранним спецификациям системного уровня. На ранних стадиях ЖЦ СБ ПО верификация является статической, она может включать в себя, например, исследование, просмотр, формальную проверку. Когда программа полностью готова, становится возможным проведение динамического тестирования. Для верификации требуется объединение информации обоих типов. Например, верификация программного модуля статическими средствами включает в себя такие методы, как просмотр программ, прогон, статический анализ, формальная проверка. Верификация программ динамическими средствами включает в себя функциональное тестирование, тестирование методом «белого ящика», статистическое тестирование. Использование проверок обоих типов позволяет утверждать, что каждый программный модуль удовлетворяет соответствующей спецификации.

4 См. таблицу В.8 приложения В.

Таблица А.10 — Оценка функциональной безопасности (см. раздел 8)

Метод/средство*	Структурный элемент ГОСТ 34332.5—2021	УПБ 1	УПБ 2	УПБ 3	УПБ 4
1 Таблица контрольных проверок	Б.2.5	Р	Р	Р	Р
2 Таблицы решений (таблицы истинности)	В.6.1	Р	Р	Р	Р
3 Анализ отказов	Таблица Б.4	Р	Р	ОР	ОР

Окончание таблицы А.10

Метод/средство*	Структурный элемент ГОСТ 34332.5—2021	УПБ 1	УПБ 2	УПБ 3	УПБ 4
4 Анализ отказов по общей причине различного ПО (если оно используется)	В.6.3	- -	Р	ОР	ОР
5 Структурные схемы надежности	В.6.4	Р	Р	Р	Р
6 Прямая прослеживаемость между требованиями раздела 6 и планом оценки функциональной безопасности ПО	В.2.11	Р	Р	ОР	ОР
* Соответствующие методы/средства следует выбирать в соответствии с УПБ. Примечание — См. таблицу В.9 приложения В.					

Приложение Б
(обязательное)

Подробные таблицы

Таблица Б.1 — Стандарты для проектирования и кодирования (см. таблицу А.4 приложения А)

Метод/средство*	Структурный элемент ГОСТ 34332.5—2021	УПБ 1	УПБ 2	УПБ 3	УПБ 4
1 Использование стандартов кодирования для сокращения вероятности ошибок	В.2.6.2	ОР	ОР	ОР	ОР
2 Неприменение динамических объектов	В.2.6.3	ОР	ОР	ОР	ОР
3 Альтернативные методы					
3а Неприменение динамических переменных	В.2.6.3	- -	ОР	ОР	ОР
3б Проверка создания динамических переменных в неавтономном режиме	В.2.6.4	- -	ОР	ОР	ОР
4 Ограниченное использование прерываний	В.2.6.5	ОР	ОР	ОР	ОР
5 Ограниченное использование указателей	В.2.6.6	- -	ОР	ОР	ОР
6 Ограниченное использование рекурсий	В.2.6.7	- -	ОР	ОР	ОР
7 Не использовать неструктурированное управление в программах, написанных на языках высокого уровня	В.2.6.2	ОР	ОР	ОР	ОР
8 Не использовать автоматическое преобразование типов	В.2.6.2	ОР	ОР	ОР	ОР

* Методы/средства следует выбирать в соответствии с УПБ. Альтернативные или эквивалентные методы/средства обозначают буквами, следующими за номером. Следует применять только один (одно) из альтернативных или эквивалентных методов/средств. Выбор альтернативных методов/средств должен быть обоснован в соответствии со свойствами, приведенными в приложении Б, предпочтительно для каждого применения.

Примечания

1 Методы 2, 3а и 5: использование динамических объектов (например, при реализации стека или динамически распределяемой области памяти) может наложить ограничения на объем доступной памяти и время выполнения. Методы 2, 3а и 5 не следует применять, если используется компилятор, который обеспечивает, что:

- для всех динамических переменных и объектов перед выполнением будет выделено достаточно памяти, а в случае ошибки выделения памяти система перейдет в безопасное состояние;
- время реакции системы соответствует заданным требованиям.

2 См. таблицу В.10 приложения В.

Таблица Б.2 — Динамический анализ и тестирование (см. таблицы А.5 и А.9 приложения А)

Метод/средство*	Структурный элемент ГОСТ 34332.5—2021	УПБ 1	УПБ 2	УПБ 3	УПБ 4
1 Выполнение тестового примера с анализом граничных значений	В.5.4	Р	ОР	ОР	ОР
2 Выполнение тестового примера с предполагаемыми ошибками	В.5.5	Р	Р	Р	Р
3 Выполнение тестового примера с введением («засевом») ошибок	В.5.6	- -	Р	Р	Р
4 Выполнение тестового примера, сгенерированного на основе модели	В.5.27	Р	Р	ОР	ОР
5 Моделирование реализации	В.5.20	Р	Р	Р	ОР

Окончание таблицы Б.2

Метод/средство*	Структурный элемент ГОСТ 34332.5—2021	УПБ 1	УПБ 2	УПБ 3	УПБ 4
6 Тестирование с разделением входных данных на классы эквивалентности	В.5.7	Р	Р	Р	ОР
7 Альтернативные методы					
7а Структурный тест с охватом 100 % (точки входа)	В.5.8	ОР	ОР	ОР	ОР
7б Структурный тест с охватом 100 % (операторы)**	В.5.8	Р	ОР	ОР	ОР
7в Структурный тест с охватом 100 % (условные переходы)**	В.5.8	Р	Р	ОР	ОР
7г Структурный тест с охватом 100 % (составные условия, MC/DC)**	В.5.8	Р	Р	Р	ОР

* Методы/средства следует выбирать в соответствии с УПБ. Альтернативные или эквивалентные методы/средства обозначают буквами, следующими за номером. Следует применять только один (одно) из альтернативных или эквивалентных методов/средств. Выбор альтернативных методов/средств должен быть обоснован в соответствии со свойствами, приведенными в приложении Б, предпочтительно для каждого применения.

** Если охват 100 % не может быть достигнут (например, заявленный охват защитным кодом), то должно быть дано соответствующее объяснение.

Примечания

1 Анализ с использованием тестовых примеров проводят на уровне подсистем, он основан на спецификациях и/или спецификациях и текстах программ.

2 См. таблицу В.11 приложения В.

Таблица Б.3 — Функциональное тестирование и проверка методом «черного ящика» (см. таблицы А.5 — А.7 приложения А)

Метод/средство*	Структурный элемент ГОСТ 34332.5—2021	УПБ 1	УПБ 2	УПБ 3	УПБ 4
1 Выполнение тестового примера на основе анализа граничных значений	Б.6.6.2	- -	- -	Р	Р
2 Выполнение тестового примера, сгенерированного на основе модели	В.5.27	Р	Р	ОР	ОР
3 Макетирование/анимация	В.5.17	- -	- -	Р	Р
4 Разделение входных данных на классы эквивалентности, включая анализ граничных значений	В.5.7, В.5.4	Р	ОР	ОР	ОР
5 Моделирование процесса	В.5.18	Р	Р	Р	Р

* Методы/средства следует выбирать в соответствии с УПБ.

Примечания

1 Анализ с использованием тестовых примеров выполняют на уровне систем ПО, и он основан только на спецификациях.

2 Полнота моделирования будет зависеть от УПБ, сложности и применения.

3 См. таблицу В.12 приложения В.

Таблица Б.4 — Анализ отказов (см. таблицу А.10 приложения А)

Метод/средство*	Структурный элемент ГОСТ 34332.5—2021	УПБ 1	УПБ 2	УПБ 3	УПБ 4
1 Альтернативные методы					
1а Причинно-следственные диаграммы	Б.6.6.2	Р	Р	Р	Р

Окончание таблицы Б.4

Метод/средство*	Структурный элемент ГОСТ 34332.5—2021	УПБ 1	УПБ 2	УПБ 3	УПБ 4
1б Анализ методом дерева событий	Б.6.6.3	Р	Р	Р	Р
2 Анализ методом дерева отказов	Б.6.6.5	Р	Р	Р	Р
3 Анализ функциональных отказов программного обеспечения	Б.6.6.4	Р	Р	Р	Р

* Методы/средства следует выбирать в соответствии с УПБ. Альтернативные или эквивалентные методы/средства обозначают буквами, следующими за номером. Следует применять только один (одно) из альтернативных или эквивалентных методов/средств. Выбор альтернативных методов/средств должен быть обоснован свойствами, приведенными в приложении В, предпочтительно для каждого применения.

Примечания

1 Предварительно должен быть проведен анализ рисков для определения того, к какому УПБ следует отнести ПО.

2 См. таблицу В.13 приложения В.

Таблица Б.5 — Моделирование (см. таблицу А.7 приложения А)

Метод/средство*	Структурный элемент ГОСТ 34332.5—2021	УПБ 1	УПБ 2	УПБ 3	УПБ 4
1 Диаграммы потоков данных	В.2.2	Р	Р	Р	Р
2 Альтернативные методы					
2а Метод конечных автоматов	Б.2.3.2	- -	Р	ОР	ОР
2б Формальные методы	Б.2.2, В.2.4	- -	Р	Р	ОР
2в Моделирование во времени сетями Петри	Б.2.3	- -	Р	ОР	ОР
3 Моделирование реализации	В.5.20	Р	ОР	ОР	ОР
4 Макетирование/анимация	В.5.17	Р	Р	Р	Р
5 Структурные диаграммы	В.2.3	Р	Р	Р	ОР

* Методы/средства следует выбирать в соответствии с УПБ. Альтернативные или эквивалентные методы/средства обозначают буквами, следующими за номером. Следует выполнять только один (одно) из альтернативных или эквивалентных методов/средств. Выбор альтернативных методов/средств должен быть обоснован свойствами, приведенными в приложении В, предпочтительно для каждого применения.

Примечания

1 Если конкретный метод не перечислен в таблице, не следует считать, что он исключен из рассмотрения. Такой метод должен соответствовать требованиям настоящего стандарта.

2 Количественное значение вероятностей не требуется.

3 См. таблицу В.14 приложения В.

4 Ссылки (являющиеся справочными, а не обязательными) во второй графе таблицы указывают на подробные описания методов/средств, изложенных в приложениях В и С [5].

Таблица Б.6 — Тестирование рабочих характеристик (см. таблицы А.5 и А.6 приложения А)

Метод/средство*	Структурный элемент ГОСТ 34332.5—2021	УПБ 1	УПБ 2	УПБ 3	УПБ 4
1 Проверка на критические нагрузки и стресс-тестирование	В.5.21	Р	Р	ОР	ОР
2 Ограничения на время ответа и объем памяти	В.5.22	ОР	ОР	ОР	ОР
3 Требования к реализации	В.5.19	ОР	ОР	ОР	ОР

* Методы/средства следует выбирать в соответствии с УПБ.

Примечание — См. таблицу В.15 приложения В.

Таблица Б.7 — Полуформальные методы (см. таблицы А.1, А.2 и А.4 приложения А)

Метод/средство*	Структурный элемент ГОСТ 34332.5—2021	УПБ 1	УПБ 2	УПБ 3	УПБ 4
1 Логические/функциональные блок-схемы	См. примечание 1	Р	Р	ОР	ОР
2 Диаграммы последовательности действий	См. примечание 1	Р	Р	ОР	ОР
3 Диаграммы потоков данных	В.2.2	Р	Р	Р	Р
4 Альтернативные методы					
4а Конечные автоматы/диаграммы переходов	Б.2.3.2	Р	Р	ОР	ОР
4б Моделирование во времени сетями Петри	Б.2.3.3	Р	Р	ОР	ОР
5 Модели данных «сущность—связь—атрибут»	Б.2.4.4	Р	Р	Р	Р
6 Диаграммы последовательности сообщений	В.2.14	Р	Р	Р	Р
7 Таблицы решений и таблицы истинности	В.6.1	Р	Р	ОР	ОР
8 UML	В.3.12	Р	Р	Р	Р

* Методы/средства следует выбирать в соответствии с УПБ. Альтернативные или эквивалентные методы/средства обозначают буквами, следующими за номером. Следует выполнять только один (одно) из альтернативных или эквивалентных методов/средств. Выбор альтернативных методов/средств должен быть обоснован свойствами, приведенными в приложении В, предпочтительно для каждого применения.

Примечание — См. таблицу В.16 приложения В.

Таблица Б.8 — Статический анализ (см. таблицу А.9 приложения А)

Метод/средство*	Структурный элемент ГОСТ 34332.5—2021	УПБ 1	УПБ 2	УПБ 3	УПБ 4
1 Анализ граничных значений	В.5.4	Р	Р	НР	НР
2 Таблица контрольных проверок	В.2.5	Р	Р	Р	Р
3 Анализ потоков управления	В.5.9	Р	ОР	ОР	ОР
4 Анализ потоков данных	В.5.10	Р	ОР	ОР	ОР
5 Предположение ошибок	В.5.5	Р	Р	Р	Р
6 Альтернативные методы					
6а Формальные проверки, включая конкретные критерии	В.5.14	Р	Р	ОР	ОР
6б Сквозной контроль (программного обеспечения)	В.5.15	Р	Р	Р	Р
7 Тестирование на символьном уровне	В.5.11	- -	- -	Р	Р
8 Анализ проекта	В.5.16	ОР	ОР	ОР	ОР
9 Статический анализ выполнения программы с ошибкой	Б.2.2, В.2.4	Р	Р	Р	ОР
10 Временной анализ выполнения при наихудших условиях	В.5.20	Р	Р	Р	Р

* Методы/средства следует выбирать согласно с УПБ. Альтернативные или эквивалентные методы/средства обозначают буквами, следующими за номером. Следует выполнять только один (одно) из альтернативных или эквивалентных методов/средств. Выбор альтернативных методов/средств должен быть обоснован свойствами, приведенными в приложении В, предпочтительно для каждого применения.

Примечание — См. таблицу В.17 приложения В.

Таблица Б.9 — Модульный подход (см. таблицу А. 4 приложения А)

Метод/средство*	Структурный элемент ГОСТ 34332.5—2021	УПБ 1	УПБ 2	УПБ 3	УПБ 4
1 Ограничение размера программного модуля	В.2.9	ОР	ОР	ОР	ОР
2 Управление сложностью программного обеспечения	В.5.13	Р	Р	ОР	ОР
3 Ограничение доступа/инкапсуляция информации	В.2.8	Р	ОР	ОР	ОР
4 Ограниченное число параметров/фиксированное число параметров подпрограммы	В.2.9	Р	Р	Р	Р
5 Одна точка входа и одна точка выхода в каждой подпрограмме и функции	В.2.9	ОР	ОР	ОР	ОР
6 Полностью определенный интерфейс	В.2.9	ОР	ОР	ОР	ОР

* Методы/средства следует выбирать согласно УПБ. Использование одного метода является, по-видимому, недостаточным. Следует рассматривать все соответствующие методы.

Примечание — См. таблицу В.18 приложения В.

Приложение В
(справочное)

Свойства стойкости к систематическим отказам программного обеспечения

В.1 Введение

Учитывая большое число факторов, влияющих на стойкость к систематическим отказам ПО, невозможно создать алгоритм, включающий в себя методы и средства и обеспечивающий необходимый результат для любого заданного применения. Целями настоящего приложения являются:

- предоставление руководства по выбору конкретных методов из приложений А и Б для обеспечения на систематической основе возможностей ПО;
- оказание помощи в обосновании использования методов, которые не перечислены в приложениях А и Б.

Настоящее приложение является дополнением к приложениям А и Б.

В.1.1 Структура приложения В, связанная с приложениями А и Б

Выходные данные каждой стадии ЖЦ СБ ПО определены в таблице А.1 приложения А. Например, рассмотрена спецификация требований к безопасности ПО.

В таблице А.1 приложения А рекомендованы конкретные методы разработки спецификации требований к СБ ПО:

Метод/средство*	Структурный элемент ГОСТ 34332.5—2021	УПБ 1	УПБ 2	УПБ 3	УПБ 4
1 Альтернативные методы					
1а Полуформальные методы	Таблица Б.7	Р	Р	ОР	ОР
1б Формальные методы	Б.2.2, В.2.4	- -	Р	Р	ОР
2 Прямая прослеживаемость между требованиями к системе безопасности и требованиями к ПО системы безопасности	В.2.11	Р	Р	ОР	ОР
3 Обратная прослеживаемость между требованиями к системе безопасности и предполагаемыми потребностями безопасности	В.2.11	Р	Р	ОР	ОР
4 Компьютерные средства разработки спецификаций для поддержки перечисленных выше соответствующих методов/средств	Б.2.4	Р	Р	ОР	ОР

В таблице В.1 отмечено, что спецификация требований к безопасности ПО характеризуется свойствами, приведенными в ГОСТ 34332.5—2021 (приложение Е) и представленными ниже.

В.1.2 Свойства

В.1.2.1 Полнота охвата ПО потребностей безопасности.

В.1.2.2 Корректность охвата ПО потребностей безопасности.

В.1.2.3 Отсутствие ошибок в спецификации, включая отсутствие неоднозначности.

В.1.2.4 Четкость требований к системе безопасности.

В.1.2.5 Отсутствие неблагоприятного взаимовлияния функций, не связанных с безопасностью, и функций безопасности, реализуемых ПО Э/Э/ПЭ СБЗС системы.

В.1.2.6 Способность обеспечения проведения оценки и подтверждения соответствия.

В таблице В.1 также ранжированы неформальные шкалы эффективности *R*1/*R*2/*R*3 конкретных методов для достижения вышеперечисленных свойств.

По методу/средству 1а «Полуформальные методы» учитывают свойства следующим образом:

- В.1.2.1 — *R*1 «Дружественный или зависящий от предметной области метод спецификации и нотация, используемые специалистами в проблемной области»;
- В.1.2.2 — *R*1 «Дружественный или зависящий от предметной области метод спецификации и нотация, используемые специалистами в проблемной области»; *R*2 «Верификация спецификации согласно критериям охвата»;
- В.1.2.3 — *R*1 «Метод и нотация, которые помогают предотвратить или обнаружить внутреннюю несогласованность, неверное поведение или математически несовместимые выражения»; *R*2 «Проверка спецификации согласно критериям охвата»; *R*3 «Проверка спецификации, основанная на систематическом анализе и/или систематическом предотвращении определенных типов отказов внутри спецификации»;

- В.1.2.4 — *R*1 «Определяемая нотация, ограничивающая возможность для непонимания»; *R*2 «Применение пределов сложности в спецификации»;

- В.1.2.5 — «- -»;

- В.1.2.6 — *R*1 «Определяемая нотация, снижающая неоднозначность в спецификации».

При ранжировании по шкалам *R*1—*R*3, а также по методу «- -» учитывают следующие свойства:

*R*1 — без объективных критериев приемки или с ограниченными объективными критериями приемки, например: тестирование методом «черного ящика», основанное на профессиональном суждении; полевые испытания;

*R*2 — с объективными критериями приемки, гарантирующими с высокой долей вероятности, что необходимое свойство достигнуто (исключения должны быть определены и обоснованы), например: тест или аналитические методы с метриками охвата, охват таблицами контрольных проверок;

*R*3 — с объективным систематическим обоснованием того, что необходимое свойство достигнуто, например: формальное доказательство, демонстрирующее соблюдение архитектурных ограничений, которые гарантируют свойство;

- - — данный метод не относится к этому свойству.

То, что можно связать со свойствами спецификации требований к ПО Э/Э/ПЭ СБЗС системы (как основание для обеспечения безопасности ПО), зависит от строгости методов, с помощью которых были достигнуты необходимые свойства спецификации требований к ПО системы. Строгость метода неформально ранжируется по шкалам *R*1, *R*2, *R*3, причем *R*1 — наименее строгий метод и *R*3 — наиболее строгий метод.

Для каждого метода, обеспечивающего конкретное свойство, определяют одно из ранжированных значений *R*1/*R*2/*R*3 в зависимости от уровня строгости этого метода.

Пример — Метод 1а «Полуформальные методы» определяют по шкалам R1 «Определяемая нотация, ограничивающая возможность для непонимания»; R2 «Применение пределов сложности в спецификации».

В данном примере применение полуформального метода со строгостью *R*1 обеспечивает ограниченную нотацию, которая улучшает точность выражений, и увеличение строгости до *R*2, дополнительно ограничивая сложность спецификации, что в противном случае могло бы привести к путанице.

Метод/средство	Свойства					
	1	2	3	4	5	6
1а Полуформальные методы				*R*1 Определяемая нотация, ограничивающая возможность для непонимания. *R*2 Применение пределов сложности в спецификации		

В.1.3 Используемый метод 1

Общие рекомендации

Если можно убедительно продемонстрировать, что необходимые свойства достигнуты при разработке спецификации требований к безопасности СБ ПО, то есть основания полагать, что спецификация требований к безопасности СБ ПО является адекватной основой для разработки ПО с достаточным уровнем систематической полноты безопасности.

Каждый из методов, представленных в таблице А.1 приложения А, в той или иной степени обеспечивает выполнение одного или большего числа вышеупомянутых свойств по таблице В.1, которые относятся к спецификации требований к СБ ПО.

Несмотря на то что в таблице А.1 приложения А рекомендованы конкретные методы, эти рекомендации не являются нормативными. Фактически из положений приложения А ясно следует, что «учитывая большое число факторов, влияющих на стойкость к систематическим отказам ПО, невозможно создать алгоритм, включающий методы и средства и обеспечивающий необходимый результат для любого заданного применения».

Выбор методов, используемых при разработке спецификации требований к ПО Э/Э/ПЭ СБЗС систем, обусловлен рядом практических ограничений (см. 7.1.2.7) в дополнение к возможностям методов. Такие ограничения могут включать в себя:

- непротиворечивость и взаимодополняющий характер выбранных методов, языков и инструментов для всего цикла разработки;

- неполное понимание разработчиками используемых ими методов, языков и инструментов;

- недостаточную адаптацию методов, языков и инструментов к тем проблемам, для решения которых они использованы.

Таблица В.1 может быть использована для сравнения относительной эффективности конкретных методов, представленных в таблице А.1 приложения А, с целью достижения требуемых свойств на стадиях (этапах) ЖЦ ПО СБЗС системы с учетом практических ограничений для конкретного разрабатываемого проекта.

Например, для верификации и подтверждения соответствия использование формального метода (*R*3) более обосновано, чем полуформального метода (*R*2), однако другие ограничения проекта (например, доступность сложных компьютерных инструментов поддержки или узкоспециализированное представление формальной нотации) могут повлиять на выбор полуформального подхода.

Таким образом, требуемые свойства, представленные в таблице В.1, могут служить основанием для аргументированного и практического сравнения альтернативных методов, которые рекомендованы в таблице А.1 приложения А и предназначены для разработки спецификации требований к ПО системы безопасности. В общем случае при рассмотрении требуемых свойств, перечисленных в таблице В, для конкретной стадии ЖЦ может быть сделан аргументированный выбор применения нескольких альтернативных методов, рекомендованных в приложении В.

Следует обратить внимание на то, что из-за систематического характера поведения свойства, представленные в приложении В, возможно, не будут достигнуты или строго доказаны. Скорее, свойства являются целью, к которой необходимо стремиться. Для достижения этой цели могут потребоваться компромиссы между различными свойствами, например между степенью защиты проекта и его простотой.

Наконец, в дополнение к определению критериев *R*1/*R*2/*R*3 полезно в качестве рекомендации сформировать неформальную связь между ростом уровня строгости от *R*1 к *R*3 и ростом уверенности в корректности ПО. Так как рекомендация является общей и неформальной, необходимо стремиться к следующим минимальным уровням строгости, когда согласно условиям приложения А требуется соответствующий ему УПБ:

- УПБ 1/2 — *R*1;
- УПБ 3 — *R*2 (если возможно);
- УПБ 4 — наиболее высокая возможная строгость.

В.1.4 Используемый метод 2

Несмотря на то что в приложении А рекомендованы конкретные методы, также допускается применять другие методы и средства при условии соблюдения требований и целей стадии (этапа) ЖЦ.

Ранее отмечено, что на систематическую способность ПО влияют многие факторы, и невозможно представить такой алгоритм для выбора и комбинирования методов, который гарантирует в любом конкретном приложении достижение определенных свойств.

Может существовать несколько эффективных способов обеспечения требуемых свойств, и разработчики системы могут быть в состоянии представить альтернативные доказательства. Информация в таблицах настоящего приложения может быть использована в качестве основы для аргументации обоснования выбора методов, отсутствующих в таблицах приложения А.

В.2 Свойства для систематической полноты безопасности

В соответствии с руководящими указаниями, представленными в настоящем стандарте и ГОСТ 34332.2, определяют конкретные методы обеспечения выполнения свойств систематической полноты безопасности и формирования убедительных доказательств. Если метод не способствует достижению свойства, то в таблицах настоящего приложения это показано как «- -». Влияние метода как отрицательное, так и положительное на свойства. Альтернативные методы обозначены буквами «а» и «б», следующими за порядковым номером данного метода в левой колонке таблиц.

Таблица В.1 — Свойства для обеспечения систематической полноты безопасности. Спецификация требований к СБ ПО (см. 7.2 и таблицу А.1 приложения А)

Метод/средство	Свойство					
	Полнота охвата потребностей безопасности программным обеспечением	Корректность охвата потребностей безопасности программным обеспечением	Отсутствие ошибок в спецификации, включая отсутствие неоднозначности	Четкость требований к системе безопасности	Отсутствие неблагоприятного взаимовлияния функций, не связанных с безопасностью, и функций безопасности, реализуемых ПО системы безопасности	Способность обеспечения проведения оценки и подтверждения соответствия
1а Полуформальные методы	*R*1 Дружественный или зависящий от предметной области метод спецификации и нотация, используемые специалистами в проблемной области	*R*1 Дружественный или зависящий от предметной области метод спецификации и нотация, используемые специалистами в предметной области. *R*2 Верификация спецификации согласно критерию охвата	*R*1 Метод и нотация, которые помогают предотвратить или обнаружить внутреннюю несогласованность, отсутствующее поведение или математически несовместимые выражения. *R*2 Проверка спецификации согласно критериям охвата. *R*3 Проверка спецификации, основанная на систематическом анализе и/или систематическом предотвращении определенных типов отказов внутри спецификации	*R*1 Определяемая нотация, ограничивающая возможность для непонимания. *R*2 Применение пределов сложности в спецификации	- -	*R*2 Определяемая нотация, уменьшающая неоднозначность в спецификации

Продолжение таблицы В.1

Метод/средство	Свойство					
	Полнота охвата потребностей безопасности программным обеспечением	Корректность охвата потребностей безопасности программным обеспечением	Отсутствие ошибок в спецификации, включая отсутствие неоднозначности	Четкость требований к системе безопасности	Отсутствие неблагоприятного взаимовлияния функций, не связанных с безопасностью, и функций безопасности, реализуемых ПО системы безопасности	Способность обеспечения проведения оценки и подтверждения соответствия
1б Формальные методы	*R*1 Дружественный или зависящий от предметной области метод спецификации и нотация, используемые специалистами в предметной области	*R*1 Дружественный или зависящий от предметной области метод спецификации и нотация, используемые специалистами в предметной области. *R*2 Верификация спецификации согласно критерию охвата. *R*3 Гарантия правильности на ограниченных аспектах поведения	*R*1 Метод и нотация, которые помогают предотвратить или обнаружить внутреннюю несогласованность, неверное поведение или математически несовместимые выражения. *R*2 Проверка спецификации согласно критериям охвата. *R*3 Проверка спецификации, основанная на систематическом анализе и/или систематическом предотвращении определенных типов отказов внутри спецификации	-- Примечание — Может усложнить достижение этого свойства, если метод не является дружественным для приложения или не зависящим от предметной области	--	*R*1 Уменьшение неоднозначности в спецификации
2 Прямая прослеживаемость между спецификацией требований к безопасности системы и требованиями к безопасности ПО	*R*1 Уверенность в том, что спецификация требований к безопасности программного обеспечения получает системные требования к безопасности	--	--	--	--	--

Окончание таблицы В.1

Метод/средство	Свойство					
	Полнота охвата потребностей безопасности программным обеспечением	Корректность охвата потребностей безопасности программным обеспечением	Отсутствие ошибок в спецификации, включая отсутствие неоднозначности	Четкость требований к системе безопасности	Отсутствие неблагоприятного взаимовлияния функций, не связанных с безопасностью, и функций безопасности, реализуемых ПО системы безопасности	Способность обеспечения проведения оценки и подтверждения соответствия
3 Обратная прослеживаемость между требованиями к системе безопасности и предполагаемыми потребностями в безопасности	- -	*R*1 Уверенность в том, что спецификация требований к ПО Э/Э/ПЭ СБЗС системы не содержит ненужной сложности	- -	*R*1 Прослеживаемость потребностей безопасности УО способствует его доступности	- -	- -
4 Компьютерные средства разработки спецификаций для поддержки соответствующих перечисленных выше методов/средств	*R*1 Инкапсуляция знаний в предметной области УО и программной среды. *R*2 Если перечень вопросов, которые необходимо учесть в таблице контрольных проверок, определен, обоснован и охватывает проблему	*R*1 Методы функционального моделирования. *R*2 Функциональное моделирование в соответствии с определенными и обоснованными критериями охвата	*R*2 Синтаксическая и семантическая проверки, чтобы убедиться в том, что соответствующие правила выполнены	*R*1 Анимация или просмотр спецификации	*R*1 Идентификация связанных и не связанных с безопасностью функций	*R*1 Помощь в прослеживаемости и в охвате. *R*2 Измерение прослеживаемости и охвата

Таблица B.2 — Свойства систематической полноты безопасности. Проектирование и разработка ПО. Разработка архитектуры ПО (см. 7.4.3, структурный элемент, приведенный в таблице А.2 приложения А)

Метод/средство	Свойство							
	Полнота спецификации требований к безопасности ПО	Корректность спецификации требований к ПО Э/Э/ПЭ СБЗС системы	Отсутствие ошибок проекта	Простота и понятность	Предсказуемость поведения	Верифицируемость и тестируемость проекта	Отказоустойчивость	Защита от отказов по общей причине, вызванной одним событием
1 Обнаружение ошибок	- -	- -	- -	- - Может быть осложнено достижение этого свойства	*R*1 Мониторинг потока логических программ обеспечивает предсказуемость	- -	*R*1 (*R*2, если цели покрытия определены, обоснованы и выполнены	*R*1 или - -
2 Коды обнаружения ошибок	- -	- -	- -	- - Достижение этого свойства может быть сложным	*R*1 Достижение этого свойства может быть сложным	- -	*R*1 (*R*2, если цели покрытия определены, обоснованы и выполнены). Эффективен для конкретных областей применения, например для передачи данных	*R*1 Эффективность для конкретных областей применения, например для передачи данных
3а Программирование с проверкой ошибок	- -	*R*2 Проверка соответствия с помощью постусловий детальным требованиям	- -	*R*2 Ограничения входного пространства с помощью предусловий	*R*2 Проверка выходов, которые должны быть ожидаемыми либо приемлемыми, с помощью постусловий	*R*2 Ограничение входного пространства и необходимого тестируемого пространства с помощью предусловий	*R*3 Эффективность для конкретных отказов	*R*3 Эффективность для конкретных отказов

Продолжение таблицы В.2

Метод/средство	Свойство							
	Полнота спецификации требований к безопасности ПО	Корректность спецификации требований к ПО Э/Э/ПЭ СБЗС системы	Отсутствие ошибок проекта	Простота и понятность	Предсказуемость поведения	Верифицируемость и тестируемость проекта	Отказоустойчивость	Защита от отказов по общей причине, вызванной одним событием
3б Разнообразные методы мониторинга (с независимостью между монитором и контролируемой функцией на одном компьютере)	- -	- -	*R*2 Реализация минимальных требований безопасности посредством независимого контроля	*R*2 Неявное разнообразие благодаря независимому контролю	*R*2 Реализация минимальных требований безопасности с использованием простого способа независимого контроля	*R*2 Реализация минимальных требований безопасности с использованием простого способа независимого контроля	*R*1 (*R*2, если цели охвата определены, обоснованы и выполнены)	*R*1 (*R*2, если цели охвата определены, обоснованы и выполнены)
3в Разнообразные методы мониторинга (с разделением между монитором и контролируемым компьютером)	- -	- -	*R*2 Реализация минимальных требований безопасности посредством независимого контроля	*R*2 Неявное разнообразие, обеспечиваемое независимым контролем	*R*2 Реализация минимальных требований безопасности посредством простого способа независимого контроля	*R*2 Реализация минимальных требований безопасности посредством независимого контроля	*R*1 (*R*2, если цели охвата определены, обоснованы и выполнены)	*R*1 (*R*2, если цели охвата определены, обоснованы и выполнены)
3г Разнообразная избыточность, реализующая те же требования к спецификации требований к безопасности ПО	- -	- -	- -	- - Примечание — Может осложнить достижение этого свойства, если оно выполнено в одном и том же исполняемом ПО	- -	- -	*R*1, если отказ одной программы не оказывает негативного влияния на другие. *R*2, если цели охвата определены, обоснованы и выполнены. Отсутствие защиты от ошибок в спецификации требований	*R*1 Если отказ одной программы не оказывает негативного влияния на другие. *R*2 Если цели охвата определены, обоснованы и выполнены. Не защищает от ошибок в спецификации требований

Продолжение таблицы В.2

Метод/средство	Свойство							
	Полнота спецификации требований к безопасности ПО	Корректность спецификации требований к ПО Э/Э/ПЭ СБЗС системы	Отсутствие ошибок проекта	Простота и понятность	Предсказуемость поведения	Верифицируемость и тестируемость проекта	Отказоустойчивость	Защита от отказов по общей причине, вызванной одним событием
3д Функционально разнообразная избыточность, реализующая различные требования к безопасности ПО (обычно с применением датчиков, работающих на разных физических принципах)	- -	- -	*R*1	- - Примечание — Достижение этого свойства может быть сложным, если реализуется в одной исполнимой программе	- -	- -	*R*1, если отказ одной программы не оказывает негативного влияния на другие	*R*1, если отказ одной программы не оказывает негативного влияния на другие. Защита от ошибок в спецификации требований
3е Восстановление предыдущего состояния	- -	- -	- - Примечание — Достижение этого свойства может быть сложным	- -	- - Примечание — Достижение этого свойства может быть сложным	- -	*R*2	*R*1 (*R*2, если цели охвата определены, обоснованы и выполнены)
3ж Проектирование ПО, не сохраняющего состояние (или проектирование ПО, сохраняющее ограниченное описание состояния)	*R*2, если несохранение или сохранение ограниченного описания состояния предусмотрено в требованиях к Э/Э/ПЭ СБЗС системе	*R*2, если несохранение или сохранение ограниченного описания состояния предусмотренно требованиями к Э/Э/ПЭ СБЗС системе	*R*2, если несохранение или сохранение ограниченного описания состояния предусмотрено в требованиях к Э/Э/ПЭ СБЗС системе	*R*1, *R*2, если определены, обоснованы и выполнены ограничения для определенного возможного числа состояний	*R*1, *R*2, если определены, обоснованы и выполнены ограничения для определенного возможного числа состояний	*R*1, *R*2, если определены, обоснованы и выполнены цели верификации/тестирования возможных состояний	*R*1, если это приводит к самовосстановлению проекта. *R*2, если определены, обоснованы и выполнены цели самовосстановления	*R*1, если это приводит к самовосстановлению проекта. *R*2, если определены, обоснованы и выполнены цели самовосстановления

Продолжение таблицы В.2

Метод/средство	Свойство							
	Полнота спецификации требований к безопасности ПО	Корректность спецификации требований к ПО Э/Э/ПЭ СБЗС системы	Отсутствие ошибок проекта	Простота и понятность	Предсказуемость поведения	Верифицируемость и тестируемость проекта	Отказоустойчивость	Защита от отказов по общей причине, вызванной одним событием
4								
4а Механизмы повторных попыток парирования сбоя	- -	- -	- -	- -	Достижение этого свойства может быть сложным	- -	*R*1 (*R*2, если цели охвата определены, обоснованы и выполнены)	*R1* (*R*2, если цели охвата определены, обоснованы и выполнены)
4б Постепенное отклонение функций	- -	- -	- - Примечание — Достижение этого свойства может быть сложным	- -	- -	- -	*R*1 (*R*2, если цели охвата определены, обоснованы и выполнены)	*R*1 (*R*2, если цели охвата определены, обоснованы и выполнены)
5 Исправление ошибок методами искусственного интеллекта	- -	- - Примечание — Достижение этого свойства может быть сложным	- - Примечание — Достижение этого свойства может быть сложным	- - Примечание — Достижение этого свойства может быть сложным	- - Примечание — Достижение этого свойства может быть сложным	- - Примечание — Достижение этого свойства может быть сложным	- -	- -
6 Динамическая реконструкция	- -	- - Примечание — Достижение этого свойства может быть сложным	- - Примечание — Достижение этого свойства может быть сложным	- - Примечание — Достижение этого свойства может быть сложным	- - Примечание — Достижение этого свойства может быть сложным	- - Примечание — Достижение этого свойства может быть сложным	- -	- -

Продолжение таблицы В.2

Метод/средство	Свойство							
	Полнота спецификации требований к безопасности ПО	Корректность спецификации требований к ПО Э/Э/ПЭ СБЗС системы	Отсутствие ошибок проекта	Простота и понятность	Предсказуемость поведения	Верифицируемость и тестируемость проекта	Отказоустойчивость	Защита от отказов по общей причине, вызванной одним событием
7 Модульный подход	- -	*R*1 [*R*2, если цели декомпозиции (модульности) определены, обоснованы и выполнены]. В противном случае только *R*1	*R*1, если отсутствие конкретных типов ошибок в проекте может быть верифицировано независимо для каждого модуля. *R*3, если отсутствие конкретных типов ошибок в проекте может быть строго обосновано при проектировании модуля	*R*1 (*R*2, если цели декомпозиции определены, обоснованы и выполнены)	*R*1, (*R*2, если цели декомпозиции определены, обоснованы и выполнены)	*R*1, (*R*2, если цели декомпозиции определены, обоснованы и выполнены)	*R*1 Если модули, на которые не влияет отказ модуля, способствуют смягчению отказа/восстановлению отказавшего модуля. *R*3 Если допущение появления конкретных отказов может быть строго обосновано	*R*1, если модули, на которые могут влиять внешние события и которые могут одновременно повлиять на несколько каналов, выявлены и полностью проверены. *R*3, если появление конкретных внешних событий может быть строго обосновано
8 Использование доверительных/ проверенных элементов ПО (при наличии)	- -	*R*1, *R*2, *R*3, если элемент вносит существенный вклад в выполнение требований к системе безопасности и правильно используется	*R*1, *R*2, *R*3 Повторное использование проверенных на практике элементов. Такая возможность должна быть оправдана для элемента	*R*1 Модульный подход декомпозирует полную сложность на четко понятные блоки	*R*1, *R*2, *R*3 Элементы, проверенные на практике	- -	*R*1, *R*2 (*R*2, если возможности отказоустойчивости легко обеспечиваются элементом и правильно используются или вокруг элемента создается слой защиты)	*R*1, *R*2 (*R*2, если защита от внешних событий, которая может повлиять на одновременное использование нескольких каналов, легко обеспечивается элементом и правильно используется или если вокруг элемента создается слой защиты)

Продолжение таблицы В.2

Метод/средство	Свойство							
	Полнота спецификации требований к безопасности ПО	Корректность спецификации требований к ПО Э/Э/ПЭ СБЗС системы	Отсутствие ошибок проекта	Простота и понятность	Предсказуемость поведения	Верифицируемость и тестируемость проекта	Отказоустойчивость	Защита от отказов по общей причине, вызванной одним событием
9 Прямая прослеживаемость между спецификацией требований безопасности ПО и архитектурой ПО	*R*1 Архитектура соответствует требованиям безопасности ПО	- -	- -	- -	- -	- -	- -	- -
10 Обратная прослеживаемость между архитектурой ПО и спецификацией требований к ПО Э/Э/ПЭ СБЗС системы	- -	R1 Уверенность в том, что архитектура излишне не усложнена	- -	- -	- -	- -	- -	- -
11а Методы структурных диаграмм	- -	*R*1	- -	*R*1 (графические описания легче понять)	- -	*R*1 (структурируемые проекты легче верифицировать и тестировать)	- -	- -
11б Полуформальные методы	*R*1 Дружественные или зависящие от предметной области метод спецификации и нотация	*R*1 Дружественные или зависящие от предметной области метод спецификации и нотация	*R*2 Может выявить внутреннюю несогласованность, или неверное поведение, или математически некорректные выражения	- -	*R*2 (предоставление свидетельств по предсказуемости)	*R*2 (предоставление свидетельств по внутренней согласованности модели проекта)	- -	- -

Продолжение таблицы В.2

Метод/средство	Свойство							
	Полнота спецификации требований к безопасности ПО	Корректность спецификации требований к ПО Э/Э/ПЭ СБЗС системы	Отсутствие ошибок проекта	Простота и понятность	Предсказуемость поведения	Верифицируемость и тестируемость проекта	Отказоустойчивость	Защита от отказов по общей причине, вызванной одним событием
11в Методы формального проектирования и усовершенствования	*R*1 Дружественные и зависящие от предметной области методы спецификации и нотация	*R*1 Точное определение ограниченных аспектов поведения, которые должны соответствовать предметной области	*R*3 Возможность выявления внутренней несогласованности, или неверного поведения, или математически некорректных выражений	- - Примечание — Достижение этого свойства может быть сложным	*R*2 Предоставление доказательства по предсказуемости	*R*2	- -	- -
11г Автоматическая генерация ПО	*R*1, если исполняемое ПО автоматически генерируется из спецификации требований или из конструкции, которая была показана как полная. *R*2, если показано, что инструменты генерации имеют соответствующую предысторию	*R*1, если исполняемое ПО автоматически генерируется из спецификации требований или из конструкции, которая была показана как полная. *R*2, если показана соответствующая предыстория инструментов	*R*1, если средства генерации гарантируют исключение определенных внутренних ошибок конструкции. *R*2, если показано, что инструменты генерации имеют соответствующую предысторию	- -	- -	- -	*R*1, *R*2, *R*3, если способность к отказоустойчивости генерируется автоматически	- -

Продолжение таблицы В.2

Метод/средство	Свойство							
	Полнота спецификации требований к безопасности ПО	Корректность спецификации требований к ПО Э/Э/ПЭ СБЗС системы	Отсутствие ошибок проекта	Простота и понятность	Предсказуемость поведения	Верифицируемость и тестируемость проекта	Отказоустойчивость	Защита от отказов по общей причине, вызванной одним событием
12 Компьютерные средства разработки спецификаций и проектирования	*R*1 Инкапсуляция знаний в проблемной области УО и программной среды. *R*2, если определена, обоснована и охвачена таблица контрольных проверок, а также учтены ее проблемы	*R*1 Осуществление обратной прослеживаемости требований методов функционального моделирования. *R*2 Функциональное моделирование согласно определенным и обоснованным критериям охвата	*R*2 Семантические и синтаксические проверки для гарантирования того, что соответствующие правила выполнены	*R*1 Анимация и просмотр	- -	*R*2 Семантические и синтаксические проверки для гарантирования того, что соответствующие правила выполнены	- -	- -
13								
13а Циклическое поведение с гарантированным максимальным временем цикла	- -	*R*1 Решение вопросов синхронизации в спецификации. *R*3, если максимальное время цикла определено	*R*1 Решение вопросов синхронизации в спецификации. *R*3, если максимальное время цикла определено	- -	*R*1 Решение вопросов синхронизации в спецификации. *R*3, если максимальное время цикла определено	*R*1 Решение вопросов синхронизации в спецификации. *R*3, если максимальное время цикла определено строгим выводом	- -	- -

Окончание таблицы В.2

Метод/средство	Свойство							
	Полнота спецификации требований к безопасности ПО	Корректность спецификации требований к ПО Э/Э/ПЭ СБЗС системы	Отсутствие ошибок проекта	Простота и понятность	Предсказуемость поведения	Верифицируемость и тестируемость проекта	Отказоустойчивость	Защита от отказов по общей причине, вызванной одним событием
13б Архитектура с временным распределением	*R*3 Полнота гарантируется распределением (только для временных характеристик)	*R*3 Корректность гарантируется распределением (только для временных характеристик)	*R*3 Строгая гарантия от внутренних временных сбоев синхронизации	*R*1 Определенная нотация значительно снижает недоразумение (предсказуемость как подход)	*R*3 Неблагоприятное влияние: полное разделение во времени, отсутствие интерференции	*R*3 Существенное снижение усилий, необходимых для тестирования и сертификации системы	*R*2 Прозрачная реализация отказоустойчивости	*R*3 Отсутствие помех внешних прерываний расписанию, которое уделяет приоритетное внимание критическим значениям для безопасности
13в Управление событиями с гарантированным максимальным временем отклика	- -	- -	- -	*R*1 Архитектуры, управляемые событиями, могут препятствовать пониманию	*R*1 Архитектуры, управляемые событиями, могут препятствовать пониманию	*R*1 Более предсказуемое тестирование	- -	- -
14 Статическое выделение ресурсов	*R*1	*R*1	*R*1	*R*1 Более понятное проектирование	*R*2 С архитектурой, определяющей использование ресурсов	*R*1 Более предсказуемое тестирование	- -	- -
15 Статическая синхронизация доступа к разделяемым ресурсам	- -	*R*1 Обеспечение предсказуемости при доступе к ресурсам	*R*1 (*R*3, если поддерживается строгими выводами, обезличивающими корректность синхронизации) м	*R*1 Более понятное проектирование	*R*1, если поддерживается строгими выводами, обезличивающими корректность синхронизации	- -	- -	- -

Таблица В.3 — Свойства систематической полноты безопасности. Проектирование и разработка ПО. Инструментальные средства поддержки и языки программирования (см. 7.4.4 и таблицу А.3 приложения А)

Метод/средство	Свойство		
	Поддержка разработки с требуемыми свойствами	Четкость работы и функциональность инструментальных средств	Корректность и воспроизводимость результата
1 Выбор подходящего языка программирования	*R*2, если строгая типизация, ограниченное преобразование типов. *R*3, если определены семантики для формального вывода	- -	- -
2 Строго типизированные языки программирования	*R*2	- -	- -
3 Подмножество языка	*R*2 В зависимости от выбранного подмножества	*R*1	*R*2 В зависимости от выбранного подмножества
4а Сертифицированные средства и сертифицированные трансляторы	- -	*R*2	*R*2
4б Инструментальные средства, заслуживающие доверия на основании опыта использования	*R*1, если класс обнаруживаемых программных ошибок определяется систематически. *R*2, если существует объективное доказательство подтверждения соответствия эффективности инструментального средства	*R*1, если инструментальное средство поддержки не является специальным для данной предметной области. *R*2, если инструментальное средство поддержки создано специально для данной предметной области	*R*1, *R*2, если существует объективное доказательство подтверждения соответствия эффективности инструментального средства, например набор средств для подтверждения соответствия компиляторов

Таблица В.4 — Свойства систематической полноты безопасности. Проектирование и разработка программного обеспечения. Подробный проект (включает проектирование программной системы, проектирование программных модулей и кодирование) (см. 7.4.5 и 7.4.6)

Метод/средство	Свойств							
	Полнота спецификации требований к ПО Э/Э/ПЭ СБЗС системы	Корректность спецификации требований к безопасности ПО	Отсутствие ошибок проекта	Простота и четкость	Предсказуемость поведения	Верифицируемость и тестируемость проекта	Отказоустойчивость	Отсутствие отказов по общей причине
1в Формальный метод и методы доработки	- -	*R*3	*R*3	Примечание — Может усложнить достижение этого свойства	*R*3 Предоставление доказательств предсказуемости	*R*2	- -	- -
2 Средства автоматизированного проектирования	*R*2 Автоматизированное средство разработки спецификации, применяющее семантические и синтаксические проверки и гарантирующее, что соответствующие правила удовлетворены	*R*1	*R*2 Автоматизированное средство разработки спецификации, применяющее семантические и синтаксические проверки и гарантирующее, что соответствующие правила удовлетворены	- -	- -	*R*2 Основанные на CASE-технологии средства поддержки тестового охвата и статической проверки	- -	- -
3 Программирование с защитой	- -	- -	- -	- - Примечание — Достижение этого свойства может быть сложным	- - Примечание — Достижение этого свойства может быть сложным	- -	*R*1, (*R*2, если цели охвата определены, обоснованы и выполнены)	*R*1, (*R*2, если цели охвата определены, обоснованы и выполнены)
4 Прямая прослеживаемость между спецификацией требований к ПО системы безопасности и проектом ПО	*R*1 Соответствие проекта требованиям к ПО системы безопасности	- -	- -	- -	- -	- -	- -	- -

Таблица В.5 — Свойства систематической полноты безопасности. Проектирование и разработка ПО. Тестирование и интеграция программных модулей (см. 7.4.7, 7.4.8 и таблицу А.5 приложения А)

Метод/средство	Свойство			
	Полнота тестирования и интеграции в соответствии со спецификацией проекта программного обеспечения	Корректность тестирования и интеграции в соответствии со спецификацией проекта программного обеспечения (удовлетворительное выполнение)	Воспроизводимость	Точно определенная тестируемая конфигурация
1 Вероятностное тестирование	*R*1 (*R*2, если цели охвата набором тестовых данных, отражающих реальные условия функционирования программы, определены, обоснованы и выполнены)	*R*1 (*R*2, если необходимые результаты определены, обоснованы и выполнены)	- -	- -
2 Динамический анализ и тестирование	*R*1 (*R*2, если необходимые результаты определены, обоснованы и выполнены)	*R*1 (*R*2, если необходимые результаты определены, обоснованы и выполнены)	- -	- -
3 Регистрация и анализ данных	- -	*R*1	*R*1 Способствует согласованности процедур тестирования	*R*2, если журналы записей об отказах/тестах включают в себя подробную информацию о серии базовых программ
4 Функциональное тестирование и тестирование методом «черного ящика»	*R*1 (*R*2, если цели охвата набором тестовых данных, отражающих реальные условия функционирования программы, определены, обоснованы и выполнены)	*R*1 (*R*2, если необходимые результаты определены, обоснованы и выполнены)	- -	- -
5 Тестирование рабочих характеристик	- -	*R*1 (*R*2, если необходимые результаты определены, обоснованы и выполнены)	- -	- -

Окончание таблицы В.5

Метод/средство	Свойство			
	Полнота тестирования и интеграции в соответствии со спецификацией проекта программного обеспечения	Корректность тестирования и интеграции в соответствии со спецификацией проекта программного обеспечения (удовлетворительное выполнение)	Воспроизводимость	Точно определенная тестируемая конфигурация
6 Тестирование, основанное на модели (МВТ)	*R*2 МВТ позволяет на ранних стадиях выявлять неоднозначности в спецификации и проекте. МВТ используют, начиная с требований. *R*3, если при моделировании использованы формальные выводы и генерация тестовых примеров (TCG)	*R*2 Оценка результатов и комплектов регрессионных тестов — ключевое преимущество МВТ. *R*3, если применять формальные модели, то можно получить объективные данные по охвату тестами	*R*3 МВТ (с TCG) направлено на автоматическое выполнение сгенерированных тестов	*R*2 МВТ работает автоматически, тестируемая конфигурация должна быть точно определена; выполнение сгенерированных тестов подобно тестированию методом «черного ящика» с возможностью измерения уровня охвата исходного кода
7 Тестирование интерфейса	- -	*R*1 (*R*2, если необходимые выходы определены, обоснованы и выполнены)	- -	- -
8 Управление тестированием и средства автоматизации	*R*1 (*R*2, если цели охвата тестами определены, обоснованы и выполнены)	- -	*R*1 Автоматизация, способствующая согласованности	*R*2 Обеспечение воспроизводимости тестирования
9 Прямая прослеживаемость между спецификацией проекта ПО и спецификациями тестирования модуля и интеграции	*R*1 Соответствие спецификации тестов требованиям к ПО системы безопасности	- -	- -	*R*2 Гарантия четкой основы тестируемых требований
10 Формальная верификация	*R*3, если для создания тестовых примеров используется формальный вывод для того, чтобы показать, что все аспекты проекта реализованы	*R*3 Предоставление объективных данных о выполнении всех требований к ПО системы безопасности	*R*1, если средства поддержки недоступны. *R*2, если инструмент поддерживается	- -

Таблица В.6 — Свойства систематической полноты безопасности. Интеграция программируемых электронных устройств (ПО и АС) (см. 7.5 и таблицу А.6 приложения А)

Метод/средство	Свойство			
	Полнота интеграции в соответствии со спецификацией проекта	Корректность интеграции в соответствии со спецификацией проекта (удовлетворительное выполнение)	Воспроизводимость	Точно определенная конфигурация интеграции
1 Функциональное тестирование и тестирование методом «черного ящика»	*R*1 (*R2*, если цели охвата набором тестовых данных, отражающих реальные условия функционирования программы, определены, обоснованы и выполнены)	*R*1 (*R2*, если необходимые результаты определены, обоснованы и выполнены)	- -	- -
2 Тестирование выполнения	- -	*R*1 (*R2*, если необходимые результаты определены, обоснованы и выполнены)	- -	- -
3 Прямая прослеживаемость между требованиями проектирования системы и ПО к интеграции программных и аппаратных средств и спецификациями тестирования интеграции программных и аппаратных средств	*R*1 Соответствие спецификаций тестирования интеграции программных и аппаратных средств требованиям интеграции	- -	- -	*R*2 Гарантия четкой основы тестируемых требований

Таблица В.7 — Свойства систематической полноты безопасности. Программные аспекты подтверждения соответствия СБ ПО (см. 7.7 и таблицу А.7 приложения А)

Метод/средство	Свойство			
	Полнота подтверждения соответствия согласно спецификации проекта программного обеспечения	Корректность подтверждения соответствия согласно спецификации проекта программного обеспечения (удовлетворительное выполнение)	Воспроизводимость	Подтверждение соответствия точно определенной конфигурации
1 Вероятностное тестирование	*R*1 (*R2*, если цели охвата набором тестовых данных, отражающих реальные условия функционирования программы, определены, обоснованы и выполнены)	*R*1 (*R2*, если необходимые результаты определены, обоснованы и выполнены)	- -	- -

Окончание таблицы В.7

Метод/средство	Свойство			
	Полнота подтверждения соответствия согласно спецификации проекта программного обеспечения	Корректность подтверждения соответствия согласно спецификации проекта программного обеспечения (удовлетворительное выполнение)	Воспроизводимость	Подтверждение соответствия точно определенной конфигурации
2 Моделирование процесса	*R*1	*R*1 (*R*2, если необходимые результаты определены, обоснованы и выполнены)	- -	*R*2 Определение внешнего окружения
3 Функциональное тестирование и тестирование методом «черного ящика»	*R*1 (*R*2, если цели охвата набором тестовых данных, отражающих реальные условия функционирования программы, определены, обоснованы и выполнены)	*R*1 (*R*2, если необходимые результаты определены, обоснованы и выполнены)	- -	- -
4 Прямая прослеживаемость между спецификацией требований к ПО системы безопасности и планом подтверждения соответствия ПО системы безопасности	*R*1 План подтверждения соответствия ПО системы безопасности охватывает требования к ПО системы безопасности	- -	- -	*R*2 Гарантия четкой основы тестируемых требований
5 Обратная прослеживаемость между планом подтверждения соответствия ПО системы безопасности и спецификацией требований к ПО системы безопасности	- -	*R*1 Отсутствие излишней сложности в плане подтверждения соответствия ПО системы безопасности	- -	*R*2 Гарантия четкой основы тестируемых требований

Таблица В.8 — Свойства систематической полноты безопасности. Верификация ПО (см. 7.9 и таблицу А.9 приложения А)

Метод/средство	Свойство			
	Полнота верификации в соответствии с предыдущей стадией	Корректность верификации в соответствии с предыдущей стадией (удовлетворительное выполнение)	Воспроизводимость	Верификация точно определенной конфигурации
1 Формальное доказательство	- -	*R*3	- -	- -
2 Анимация спецификации и тестирования	*R*1	*R*1	- -	- -

Окончание таблицы В.8

Метод/средство	Свойство			
	Полнота верификации в соответствии с предыдущей стадией	Корректность верификации в соответствии с предыдущей стадией (удовлетворительное выполнение)	Воспроизводимость	Верификация точно определенной конфигурации
3 Статический анализ	- -	*R*1/*R*2/*R*3 (строгость может меняться от возможностей подмножества языка до формального анализа)	- -	- -
4 Динамический анализ и тестирование	*R*1 (*R*2, если цели охвата структурированием определены, обоснованы и выполнены)	*R*1 (*R*2, если необходимые результаты определены, обоснованы и выполнены)	- -	- -
5 Прямая прослеживаемость между спецификацией проекта ПО и планом верификации (включающим в себя верификацию данных) ПО	*R*1 Соответствие плана верификации ПО (включающего в себя верификацию данных) требованиям к ПО системы безопасности	- -	- -	*R*2 Гарантия четкой основы тестируемых требований
6 Обратная прослеживаемость между планом верификации (включающим верификацию данных) ПО и спецификацией проекта ПО	- -	*R*1 Отсутствие излишней сложности в плане верификации ПО (включающем в себя верификацию данных)	- -	*R*2 Гарантия четкой основы тестируемых требований
7 Численный анализ в автономном режиме	- -	*R*1 Высокая степень в ожидаемости точности вычислений. (*R*2 при объективных критериях приемки. *R*3, если используется совместно с объективным систематическим доказательством, обосновывающим критерии приемки)	- -	- -

Таблица В.9 — Свойства систематической полноты безопасности. Оценка функциональной безопасности (см. раздел 8 и таблицу А.10 приложения А)

Метод/средство	Свойство						
	Полнота оценки функциональной безопасности в соответствии с требованиями настоящего стандарта	Корректность оценки функциональной безопасности в соответствии с проектными спецификациями (удовлетворительное выполнение)	Доступное для анализа решение всех выявленных проблем	Возможность модификации оценки функциональной безопасности после изменения проекта без необходимости проведения тщательной переработки оценки	Воспроизводимость	Своевременность	Точно определенная конфигурация
1 Таблица контрольных проверок	*R*1	*R*1	*R*1	- -	*R*1	- -	- -
2 Таблицы решений (таблицы истинности)	*R*1	*R*2	- -	- -	*R*2	- -	- -
3 Анализ отказов	*R*2	*R*2	*R*1 (анализ отказов основан на согласованных списках отказов)	- -	*R*1 (анализ отказов основан на согласованных списках отказов)	- -	- -
4 Анализ отказов по общей причине различного ПО (если различное ПО используется)	*R*2	*R*2	*R*1 (анализ отказов по общей причине основан на согласованных списках источников общих причин)	- -	*R*1 (анализ отказов по общей причине основан на согласованных списках источников общих причин)	- -	- -
5 Структурные схемы надежности	*R*1	*R*1	- -	- -	- -	- -	- -
6 Прямая прослеживаемость между требованиями раздела 8 и планом оценки функциональной безопасности ПО	*R*1 Соответствие плана оценки функциональной безопасности ПО требованиям раздела 8	- -	- -	- -	- -	- -	- -

Таблица В.10 — Подробные свойства. Стандарты проектирования и кодирования (см. таблицу В.1)

Метод/средство	Свойство							
	Полнота спецификации требований к программному обеспечению системы безопасности	Корректность спецификации требований к программному обеспечению системы безопасности	Отсутствие ошибок проекта	Простота и четкость	Предсказуемость поведения	Верифицируемость и тестируемость проекта	Отказоустойчивость/обнаружение отказов	Отсутствие отказов по общей причине
1 Использование стандартов кодирования для сокращения вероятности ошибок	- -	- -	*R*1	*R*1 Запрет отдельных конструкций языка	*R*1	*R*1	- -	- -
2 Неиспользование динамических объектов	- -	- -	*R*1/*R*2/*R*3, в зависимости от используемого языка	- -	*R*1/*R*2, в зависимости от используемого языка	*R*1/*R*2, в зависимости от используемого языка	- -	- -
3а Неиспользование динамических переменных	- -	- -	*R*1/*R*2/*R*3, в зависимости от используемого языка	- -	*R*1/R2, в зависимости от используемого языка	*R*1/*R*2, в зависимости от используемого языка	- -	- -
3б Проверка создания динамических переменных при выполнении программы	- -	- -	*R*1/*R*2/*R*3, в зависимости от используемого языка	- -	*R*1/*R*2, в зависимости от используемого языка	*R*1/*R*2, в зависимости от используемого языка	- -	- -
4 Ограниченное использование прерываний	- -	- -	*R*1/*R*2, в зависимости от используемого языка	*R*1 Повышение четкости логики и последовательностей событий	*R*1/*R*2, в зависимости от используемого языка	*R*/R2, в зависимости от используемого языка	- -	- -
5 Ограниченное использование указателей	- -	- -	*R*1/*R*2, в зависимости от используемого языка	*R*1 Повышение четкости логики	*R*1/*R*2, в зависимости от используемого языка	*R*1/*R*2, в зависимости от используемого языка	- -	- -
6 Ограниченное использование рекурсий	- -	- -	*R*1/*R*2, в зависимости от используемого языка	- -	*R*1/*R*2, в зависимости от используемого языка	R1/*R*2, в зависимости от используемого языка	- -	- -

Окончание таблицы В.10

Метод/средство	Свойство							
	Полнота спецификации требований к программному обеспечению системы безопасности	Корректность спецификации требований к программному обеспечению системы безопасности	Отсутствие ошибок проекта	Простота и четкость	Предсказуемость поведения	Верифицируемость и тестируемость проекта	Отказоустойчивость/ обнаружение отказов	Отсутствие отказов по общей причине
7 Неиспользование неструктурированного управления в программах, написанных на языках высокого уровня	- -	- -	*R*1/*R*2, в зависимости от используемого языка	*R*1 Повышение четкости логики	*R*1/*R*2, в зависимости от используемого языка	*R*1/*R*2, в зависимости от используемого языка	- -	- -
8 Неиспользование автоматического преобразования типов	- -	*R*2 Предотвращение погрешностей округления	*R*2 Предотвращение погрешностей округления	*R*1	*R*1	- -	- -	- -

Таблица В.11 — Подробные свойства. Динамический анализ и тестирование (см. таблицу Б.2 приложения Б)

Метод/средство	Свойство			
	Полнота тестирования и верификации в соответствии со спецификацией проекта программного обеспечения	Корректность тестирования и верификации в соответствии со спецификацией проекта программного обеспечения (удовлетворительное выполнение)	Воспроизводимость	Точно определенная тестируемая и верифицируемая конфигурация
1 Выполнение тестового примера, связанного с анализом граничных значений	- -	*R*1 (*R*2, если заданы объективные критерии для граничных значений)	- -	- -
2 Выполнение тестового примера, связанного с предполагаемой ошибкой	- -	*R*1	- -	- -
3 Выполнение тестового примера, связанного с введением ошибки	- -	*R*1	- -	- -

Окончание таблицы В.11

Метод/средство	Свойство			
	Полнота тестирования и верификации в соответствии со спецификацией проекта программного обеспечения	Корректность тестирования и верификации в соответствии со спецификацией проекта программного обеспечения (удовлетворительное выполнение)	Воспроизводимость	Точно определенная тестируемая и верифицируемая конфигурация
4 Выполнение тестового примера, сгенерированного на основе модели	*R*2 Использование МВТ начиная с требований и выявление на ранних стадиях ошибок при проектировании и разработке ПО. *R*3, если при моделировании использованы формальные выводы и генерация тестовых примеров (TCG)	*R*2 Оценка результатов и комплектов регрессионных тестов — ключевое преимущество МВТ, что облегчает понимание последствий указанных требований. *R*3, если применены формальные модели, то можно получить объективные данные по охвату тестами	*R*3 Использование МВТ (с TCG) направлено на автоматическое выполнение сгенерированных тестов	*R*2 Работа МВТ в автоматическом режиме; тестируемая конфигурация должна быть точно определена; выполнение сгенерированных тестов подобно тестированию методом «черного ящика» с возможностью измерения уровня охвата исходного кода
5 Моделирование реализации	- -	*R*1 (*R*2, если цель — требования к производительности)	- -	- -
6 Разделение входных данных на классы эквивалентности	*R*1, если профиль входных данных четко определен и прост в управлении своей структурой	*R*1, если разбиения, вероятнее всего, не содержат нелинейности, т. е. все члены класса являются действительно эквивалентными	- -	- -
7 Структурное тестирование	- -	*R*1, (*R*2, если цель — требования к производительности)	- -	- -

Таблица В.12 — Подробные свойства. Функциональное тестирование и проверка методом «черного ящика» (см. таблицу В.3)

Метод/средство	Свойство			
	Полнота тестирования, интеграции и подтверждения соответствия согласно спецификациям проекта	Корректность тестирования интеграции и подтверждения соответствия согласно спецификациям проекта (удовлетворительное выполнение)	Воспроизводимость	Точно определенная конфигурация для тестирования, интеграции и подтверждения соответствия
1 Выполнение тестового примера на основе причинно-следственных диаграмм	*R*1	*R*1	- -	- -
2 Выполнение тестового примера, сгенерированного на основе модели (MBT)	*R*2 Тестирование, основанное на модели и обеспечивающее автоматическую генерацию эффективных контрольных примеров/процедур, путем использования модели системных требований и заданной функциональности. MBT способствует раннему выявлению ошибок и пониманию последствий указанных требований. *R*3, если при моделировании использованы формальные выводы и генерация тестовых примеров (TCG)	*R*2 MBT основано на моделях (в основном функциональных/поведенческих), формируемых на основании требований. *R*3, если применены формальные модели, то можно получить объективные данные по охвату тестами	*R*3 Использование MBT (с TCG) направлено на автоматическое выполнение сгенерированных тестов	*R*2 Работа MBT в автоматическом режиме, тестируемая конфигурация должна быть точно определена
3 Макетирование/анимация	- -	*R*1	- -	- -
4 Разделение входных данных на классы эквивалентности, включая анализ граничных значений	*R*1, если профиль входных данных четко определен и прост в управлении своей структурой	*R*1, если разбиения, как правило, не содержат нелинейности, т. е. все члены класса являются действительно эквивалентными	- -	- -
5 Моделирование процесса	- -	*R*1	- -	*R*2 Определение внешнего окружения

Таблица В.13 — Подробные свойства. Анализ отказов (см. таблицу В.4)

Метод/средство	Свойство						
	Полнота оценки функциональной безопасности в соответствии с требованиями настоящего стандарта	Корректность оценки функциональной безопасности в соответствии с проектными спецификациями (удовлетворительное выполнение)	Доступное для анализа решение всех выявленных проблем	Возможность модификации оценки функциональной безопасности после изменения проекта без необходимости проведения тщательной переработки оценки	Воспроизводимость	Своевременность	Точно определенная конфигурация
1а Причинно-следственные диаграммы	*R*2	*R*2	- -	- -	- -	- -	- -
1в Анализ методом дерева событий	*R*2	*R*2	- -	- -	- -	- -	- -
2 Анализ методом дерева отказов	*R*2	*R*2	- -	- -	- -	- -	- -
3 Анализ функциональных отказов ПО	*R*2	*R*2	- -	- -	- -	- -	- -

Таблица В.14 — Подробные свойства. Моделирование (см. таблицу В.5)

Метод/средство	Свойство			
	Полнота подтверждения соответствия согласно спецификации проекта программного обеспечения	Корректность подтверждения соответствия согласно спецификации проекта программного обеспечения (удовлетворительное выполнение)	Воспроизводимость	Точно определенная конфигурация подтверждения соответствия
1 Диаграммы потоков данных	- -	*R*1	- -	- -
2а Метод конечных автоматов	*R*3	*R*3	- -	- -
2б Формальные методы	*R*3	*R*3	- -	- -
2в Моделирование во времени сетями Петри	- -	*R*1	- -	- -
3 Моделирование реализации	- -	*R*1	- -	- -
4 Макетирование/анимация	- -	*R*1	- -	- -
5 Структурные диаграммы	- -	*R*1	- -	- -

Таблица В.15 — Подробные свойства. Тестирование характеристик реализации (см. таблицу В.6)

Метод/средство	Свойство			
	Полнота подтверждения соответствия согласно спецификации проекта программного обеспечения	Корректность подтверждения соответствия согласно спецификации проекта программного обеспечения (удовлетворительное выполнение)	Воспроизводимость	Точно определенная конфигурация подтверждения соответствия
1 Проверка на критические нагрузки и стресс-тестирование	- -	*R*1 (*R*2, если установлены объективные цели)	- -	- -
2 Ограничения на время ответа и объем памяти	- -	*R*1 (*R*2, если установлены объективные цели)	- -	- -
3 Требования к реализации	- -	*R*1 (*R*2, если установлены объективные цели)	- -	- -

Таблица В.16 — Подробные свойства. Полуформальные методы (см. таблицу В.7)

Метод/средство	Свойство									
	Полнота спецификации требований к ПО Э/Э/ПЭ СБЗС системы	Корректность спецификации требований к ПО Э/Э/ПЭ СБЗС системы	Отсутствие ошибок в проекте	Четкость требований к системе безопасности	Отсутствие неблагоприятного взаимовлияния функций, не связанных с безопасностью, с СБЗС системой, реализуемой ПО	Простота и четкость	Предсказуемость поведения	Верифицируемость и тестируемость проекта	Отказоустойчивость/обнаружение отказов	Отсутствие отказов по общей причине от внешних событий
1 Логические диаграммы/диаграммы функциональных блоков	*R*2	*R*2	*R*2	- -	- -	*R*1	*R*2	- -	- -	*R*1
2 Диаграммы последовательности	*R*2	*R*2	*R*2	- -	- -	*R*1	*R*2	- -	- -	*R*2
3 Диаграммы потоков данных	*R*1	*R*1	*R*1	- -	- -	*R*1 Использование при обработке транзакций	- -	- -	- -	*R*1
4а Конечные автоматы/диаграммы переходов	*R*2	*R*2	*R*2	- -	- -	*R*1 Математически полная спецификация последовательности событий	*R*2	- -	- -	*R*2

Окончание таблицы В.16

Метод/средство	Свойство									
	Полнота спецификации требований к ПО Э/Э/ПЭ СБЗС системы	Корректность спецификации требований к ПО Э/Э/ПЭ СБЗС системы	Отсутствие ошибок в проекте	Четкость требований к системе безопасности	Отсутствие неблагоприятного взаимовлияния функций, не связанных с безопасностью, с СБЗС системой, реализуемой ПО	Простота и четкость	Предсказуемость поведения	Верифицируемость и тестируемость проекта	Отказоустойчивость/обнаружение отказов	Отсутствие отказов по общей причине от внешних событий
4б Моделирование во времени сетями Петри	*R*2	*R*2	*R*2	- -	- -	*R*1 Определение взаимодействия в реальном времени	*R*2	- -	- -	*R*2
5 Модели данных «сущность — связь — атрибут»	*R*1	*R*1	*R*1	- -	- -	*R*1	- -	- -	- -	*R*1
6 Диаграммы последовательности сообщений	*R*2	*R*2	*R*2	- -	- -	*R*1	*R*2	- -	- -	*R*2
7 Таблицы решений и таблицы истинности	*R*2	*R*2	*R*2	- -	- -	*R*1 (для комбинационной логики)	*R*2	- -	- -	*R*2

Таблица В.17 — Свойства для систематической полноты безопасности. Статический анализ (см. таблицу В.8)

Метод/средство	Свойство			
	Полнота верификации в соответствии с предыдущей стадией	Корректность верификации в соответствии с предыдущей стадией (удовлетворительное выполнение)	Воспроизводимость	Точно определенная верифицируемая конфигурация
1 Анализ граничных значений	- -	*R*1 (*R*2, если заданы объективные критерии для граничных значений)	- -	- -
2 Таблица контрольных проверок	- -	*R*1	- -	*R*1

Окончание таблицы В.17

Метод/средство	Свойство			
	Полнота верификации в соответствии с предыдущей стадией	Корректность верификации в соответствии с предыдущей стадией (удовлетворительное выполнение)	Воспроизводимость	Точно определенная верифицируемая конфигурация
3 Анализ потоков управления	- -	*R*1	- -	- -
4 Анализ потоков данных	- -	*R*1	- -	- -
5 Предположение ошибок	- -	*R*1	- -	- -
6				
6а Формальные проверки, включая конкретные критерии	*R*2	*R*2	- -	*R*2
6б Сквозной контроль (программного обеспечения)	*R*1	*R*1	- -	*R*1
7 Тестирование на символьном уровне	- -	*R*2 (*R*3, если применяется для формально определенных предусловий и постусловий и выполняется инструментальным средством, использующим математически строгий алгоритм)	- -	- -
8 Анализ проекта	*R*2	*R*1 [*R*2 (с объективными критериями)]	- -	*R*2
9 Статический анализ выполнения программы с ошибкой	- -	*R*1 (*R*3 для определенных классов ошибок, если выполняется инструментальным средством, использующим математически строгий алгоритм)	- -	- -
10 Временной анализ выполнения при наихудших условиях	*R*1	*R*3	- - -	*R*2

Таблица В.18 — Подробные свойства. Модульный подход (см. таблицу В.8)

Метод/средство	Свойство							
	Полнота спецификации требований к программному обеспечению системы безопасности	Корректность спецификации требований к программному обеспечению системы безопасности	Отсутствие ошибок проекта	Простота и четкость	Предсказуемость поведения	Верифицируемость и тестируемость проекта	Отказоустойчивость/ обнаружение отказов	Отсутствие отказов по общей причине
1 Ограничение размера программного модуля	- -	- -	*R*1	*R*1	*R*1	*R*1	- -	- -
2 Управление сложностью ПО	- -	- -	*R*1	*R*1	*R*1	*R*1	- -	- -
3 Ограничение доступа/инкапсуляция информации	- -	- -	*R*1	*R*1	*R*1	*R*1	- -	- -
4 Ограниченное число параметров/фиксированное число параметров подпрограммы	- -	- -	*R*1	*R*1	*R*1	*R*1	- -	- -
5 Одна точка входа и одна точка выхода в каждой подпрограмме и функции	- -	- -	*R*1	*R*1	*R*1	*R*1	- -	- -
6 Полностью определенный интерфейс	- -	- -	*R*2	*R*1	*R*1	*R*1	- -	- -

Приложение Г
(обязательное)

Руководство по безопасности для применяемых изделий. Дополнительные требования к элементам программного обеспечения

Г.1 Цель руководства по безопасности

Г.1.1 Когда элемент системы (изделие) используют повторно или предполагается, что он будет снова использован в одной или нескольких других разрабатываемых системах, необходимо гарантировать, что этот элемент сопровождался достаточно точным и полным описанием (функций, ограничений и доказательств) для обеспечения возможности оценки полноты безопасности, заданной функции безопасности, которая полностью или частично реализуется этим элементом. Такие действия следует выполнять только с использованием руководства по безопасности.

Г.1.2 Руководство по безопасности может состоять из документации поставщика элемента, если она соответствует требованиям ГОСТ 34332.3—2021 (приложение Г) и настоящего приложения. В противном случае такая документация должна быть создана как часть проекта СБ системы.

Г.1.3 В руководстве по безопасности должны быть определены атрибуты элемента, которые могут включать в себя аппаратные и/или программные ограничения, которые интегратор должен знать и учитывать в процессе реализации применения. Руководство по безопасности служит источником информации для интегратора о свойствах элемента, а также для чего элемент был разработан, о его поведении и характеристиках.

Примечания

1 Область применения и время поставки руководства по безопасности зависят от того, кто его применяет, от типа интегратора, цели элемента и от того, кто его поставляет и поддерживает.

2 Физическое лицо, подразделение или организацию, которые интегрируют ПО, называют «интегратор».

Г.2 Содержание руководства по безопасности для элемента программного обеспечения

Г.2.1 В руководство по безопасности включают всю информацию, требуемую по ГОСТ 34332.3—2021 (приложение Г), которая относится к элементу системы. Например, пункты приложения Г ГОСТ 34332.3—2021, связанные с АС, не относятся непосредственно к элементу ПО.

Г.2.2 Элемент ПО должен быть идентифицирован, и все необходимые указания по его использованию должны быть доступны интегратору.

Примечание — Для ПО это может быть продемонстрировано с помощью четкого определения элемента и с указанием того, что содержание информации об элементе остается неизменным.

Г.2.3 Конфигурация элемента

Конфигурация элемента ПО, среды выполнения ПО и АС и, в случае необходимости, конфигурация системы компиляции/связей должны быть документально оформлены в руководстве по безопасности.

Рекомендуемая конфигурация элемента ПО должна быть документально оформлена в руководстве по безопасности, и эту конфигурацию следует использовать в применении, связанном с безопасностью.

Руководство по безопасности должно включать в себя все выдвинутые предположения, от которых зависит обоснование использования элемента.

Г.2.4 В руководство по безопасности должны быть включены следующие положения:

- компетентность. Должен быть определен минимальный уровень знаний, предполагаемый для интегратора элемента, т. е. знание конкретных инструментальных средств применения;

- степень доверия, отнесенная к элементу. Информация о любой сертификации элемента, прошедшего независимую оценку, значении полноты, которую интегратор может приписать уже существующему элементу. В эту информацию следует включать данные о полноте безопасности, для которой элемент разработан, о стандартах, использованных в процессе проектирования, и о любых ограничениях, которые интегратор должен реализовать для обеспечения требуемой стойкости к систематическим отказам. (В зависимости от функциональности элемента возможно, что некоторые требования могут быть удовлетворены только на стадии интеграции системы. В таких случаях эти требования должны быть идентифицированы для дальнейшего использования интегратором, например быстродействие и время отклика.)

Примечание — В отличие от ГОСТ 34332.3 в настоящем стандарте не требуется, чтобы в руководстве по безопасности для применяемых изделий были указаны режимы отказов ПО или значения интенсивностей отказов, так как причины отказов ПО существенно отличаются от причин случайных отказов АС (изделий) [см. ГОСТ 34332.3—2021 (приложение Г)];

- инструкции по установке. Подробное описание или ссылка на него относительно установки уже существующего элемента в интегрированную систему;

- причина выпуска новой версии элемента. Подробное описание того, что функционирующий элемент стал предметом выпуска очередной новой версии для устранения значительных отклонений или включения дополнительного функционала;

- существенные отклонения. Должно быть приведено подробное описание всех существенных отклонений с объяснением того, как происходит каждое отклонение, а также тех механизмов, которые должен использовать интегратор для ослабления отклонения, влияющего на конкретные функции;

- совместимость с предыдущими выпусками новых версий. Подробное описание наличия совместимости элемента с предыдущими версиями подсистемы, при отсутствии совместимости — подробное описание процедуры его обновления, которую необходимо выполнить;

- совместимость с другими системами. Функционирующий элемент может зависеть от специально разработанной операционной системы. В таких случаях должны быть подробно описаны детали версии специально разработанной операционной системы. Должен также быть определен стандарт на создание элемента, включающий в себя идентификацию и версию компилятора, инструменты, используемые для создания функционирующего элемента (идентификацию и версию), и применяемый тест для уже действующего элемента (идентификацию и версию);

- конфигурация элемента. Для функционирующего элемента должны быть даны имя (имена) и описание(я), включая версию элемента/номер элемента/состояние модификации элемента;

- управление изменениями. Механизм, с помощью которого интегратор может инициировать запрос на изменение разработчику ПО;

- невыполненные требования. Могут существовать требования, которые были определены, но не выполнены в текущей версии элемента. В таких случаях эти требования должны быть идентифицированы для того, чтобы их рассмотрел интегратор;

- предусмотренное проектом безопасное состояние. При определенных обстоятельствах в случае появления контролируемого отказа при применении системы элемент может перейти к своему безопасному состоянию, предусмотренному проектом. В таких случаях должно быть приведено точное определение предусмотренного проектом безопасного состояния, которое анализирует интегратор;

- ограничения интерфейса. Должны быть подробно описаны любые конкретные ограничения для заданных требований пользовательского интерфейса;

- подробное описание любых мер обеспечения безопасности, которые, возможно, были реализованы для предотвращения перечисленных угроз и уязвимостей;

- конфигурируемые элементы. Должны быть подробно описаны метод или методы конфигурации, используемые для элемента, их применение и любые ограничения на их применение.

Г.3 Обоснование требований для применяемых изделий в руководстве по безопасности

Г.3.1 В руководстве по безопасности должны быть обоснованы все требования для применяемых изделий приведением соответствующего доказательства [см. ГОСТ 34332.3—2021 (подпункт 8.3.13.7)].

Примечания

1 Важно, что требуемая характеристика элемента системы безопасности поддерживается обоснованными доказательствами. Неподдержанные требования не позволяют установить правильность и полноту функции безопасности, которую реализует элемент.

2 Доказательство поддержки может быть получено на основе документации поставщика элемента и разработчика процесса, выполняемого элементом, или может быть создано или расширено квалифицированными специалистами, разрабатывающими систему, связанную с безопасностью, или третьей стороной.

3 На доступность доказательства могут влиять коммерческие или юридические ограничения (например, авторское право или права интеллектуальной собственности). Эти ограничения выходят за рамки настоящего стандарта.

Г.3.2 Доказательство поддержки, которое в руководстве по безопасности обосновывает требования для применяемых изделий, отличается от аналогичного обоснования в руководстве по безопасности элемента ПО.

Г.3.3 Если такое доказательство, необходимое для оценки функциональной безопасности, не доступно, то элемент не подходит для использования в Э/Э/ПЭ СБЗС системах.

Приложение Д
(справочное)

Взаимосвязь между ГОСТ 34332.3 и настоящим стандартом

С помощью данных, приведенных в таблицах Д.1 и Д.2, можно определить, какие разделы ГОСТ 34332.3 необходимо использовать тем, кто решает вопросы исключительно с ПО. Большинство разделов ГОСТ 34332.3 связано с решением проблем АС. В настоящем стандарте рассмотрены наиболее значимые вопросы ПО, однако в ГОСТ 34332.3 сформулирован ряд требований в отношении ПО, которые, как правило, частично дублируют требования настоящего стандарта. Знание положений ГОСТ 34332.3 главным образом необходимо тем специалистам по ПО, которые занимаются совместимостью ПО и АС. Требования ГОСТ 34332.3 можно сгруппировать в определенные категории по таблице Д.1.

Таблица Д.1 — Категории требований ГОСТ 34332.3

Объект требований	Категория пользователей
Программное обеспечение	Для пользователей стандарта, работающих с АС, и для пользователей, работающих с ПО
Прикладное программное обеспечение	Для пользователей, работающих с ПО, которое непосредственно реализует соответствующую функцию безопасности, но не с ПО операционной системы или библиотечными функциями
Системное программное обеспечение	Для пользователей, работающих прежде всего с ПО операционной системы, библиотечными функциями и т.п.
Только аппаратные средства	Только для пользователей, не работающих с ПО
В основном аппаратные средства	Для пользователей, проявляющих только незначительный интерес к проблемам ПО

Требования ГОСТ 34332.3 к ПО и их взаимосвязь с определенными в таблице Д.1 категориями требований представлены в таблице Д.2.

Таблица Д.2 — Требования ГОСТ 34332.3 к ПО и их взаимосвязь с приведенными в таблице Е.1 приложения Е определениями терминов

Структурный элемент ГОСТ 34332.3—2021	Обеспечение/средства, значимые для пользователей	Примечание
6.2	ПО	—
6.2.3.1	Прикладное ПО	—
6.2.3.2—6.2.3.6	ПО	—
6.2.3.3	Только АС	—
6.3	ПО	7.3.2.2 — только АС
6.4	ПО	—
6.4.2.1—6.4.2.12	ПО	—
6.4.2.13, 6.4.2.14	Только АС	—
6.4.3.1—6.4.3.3	ПО	—
6.4.3.4	Только АС	—
6.4.4	Только АС	—
6.4.5	Только АС	—
6.4.6	ПО	7.4.7.6 — только АС

Окончание таблицы Д.2

Структурный элемент ГОСТ 34332.3—2021	Обеспечение/средства, значимые для пользователей	Примечание
6.4.7	ПО	7.4.7.1, 1-е, 2-е перечисления — только АС
6.4.8	Только АС	—
6.4.9.1—6.4.9.3	ПО	—
6.4.9.4, 6.4.9.5	Только АС	—
6.4.9.6, 6.4.9.7	ПО	—
6.4.10	ПО	—
6.4.11	Только АС	—
6.5	ПО	—
6.6	ПО	—
6.6.2.1, 1-е перечисление	АС	—
6.6.2.4	В основном АС	—
6.7	ПО	7.7.2.3, 7.7.2.4 — в основном прикладное ПО
6.8	ПО	—
6.9	В основном прикладное ПО	—
ПО	ПО	—
Приложение А (А.1)	В основном АС	—
Приложение А (А.2) и таблицы	В основном АС	Таблица А.10 — ПО
Приложение А (А.3)	В основном АС	Таблицы А.17, А.18 содержат некоторые аспекты ПО
Приложение Б, все таблицы	ПО	—
Приложение В	АС	—
Приложение Г	ПО	Г.2.3 — только АС

Приложение Е
(справочное)

Методы, не допускающие взаимодействия между элементами программного обеспечения на одном компьютере

Е.1 Введение

Независимое выполнение элементов ПО, работающих в одной компьютерной системе (состоящей из одного или более процессоров с памятью и другими устройствами, совместно используемыми этими процессорами), можно обеспечить и продемонстрировать с помощью различных методов. В настоящем приложении рассмотрены некоторые методы, не допускающие взаимодействия [между элементами ПО с различной стойкостью к систематическим отказам, между элементами, разработанными для реализации (или для принятия участия в реализации) одной и той же функции безопасности, или между элементами ПО, реализующими функции, связанные с безопасностью, и элементами ПО, реализующими функции, не связанные с безопасностью, на одном компьютере].

Примечание — Термин «независимость выполнения» означает, что элементы в процессе их выполнения не будут оказывать негативного влияния друг на друга, т. е. их выполнение не приведет к появлению опасного отказа. Этот термин используется для того, чтобы отличить другие аспекты независимости, которые могут потребоваться между элементами (в частности, «разнообразие») и которые соответствуют другим требованиям настоящего стандарта.

Е.2 Области поведения

Независимость выполнения должна быть обеспечена и продемонстрирована в областях пространства и времени.

Е.2.1 Пространственная область

Данные, используемые одним элементом ПО, не должны быть изменены другим элементом. В частности, данные не должны быть изменены элементом ПО, не связанным с безопасностью.

Е.2.2 Временная область

Функционирование одного элемента ПО не должно приводить к неправильному функционированию другого элемента, используя слишком большую часть общего времени процессора или блокируя работу другого элемента, запирая совместно используемый ресурс.

Е.3 Анализ причин

Для демонстрации независимости выполнения должна быть проведена для предложенного проекта идентификация всех возможных причин взаимовлияний между умозрительно независимыми (не взаимовлияющими) функционирующими элементами ПО в пространственной и временной областях. Анализ должен быть проведен при нормальных условиях функционирования и в условиях отказа. При анализе должны быть рассмотрены (как минимум):

- применение совместно используемой оперативной памяти;
- применение совместно используемых периферийных устройств;
- совместное использование времени процессора (где два элемента ПО или более выполняются на одном процессоре);
- коммуникации между элементами, необходимые для создания проекта всей системы;
- возможность того, что отказ в одном элементе (такой, как переполнение, или исключительная ситуация деления на ноль, или неправильное вычисление указателя) может вызвать последовательные отказы в других элементах.

Для обеспечения и обоснования независимости выполнения необходимо рассмотреть все эти идентифицированные источники взаимовлияний.

Е.4 Обеспечение пространственной независимости

Методы/средства для обеспечения и демонстрации пространственной независимости включают в себя использование:

- аппаратной защиты памяти между различными элементами ПО, включая элементы, различающиеся стойкостью к систематическим отказам;
- операционной системы, которая для каждого элемента ПО реализует отдельный процесс с его собственным пространством виртуальной памяти, поддерживаемым аппаратной защитой памяти;
- строгого анализа проекта, исходного кода и, возможно, объектного кода для демонстрации отсутствия любых явных или неявных обращений одного элемента ПО к памяти другого элемента, которые могут привести к искажению данных, принадлежащих этому элементу (для случая отсутствия аппаратной защиты памяти);
- программной защиты от недопустимой модификации элементом с более низким уровнем полноты данных элемента ПО с более высоким УПБ.

Передача данных от элемента ПО с более низким УПБ к элементу с более высоким уровнем УПБ не допустима в том случае, если элемент с более высоким УПБ не может проверить достаточную полноту получаемых данных.

Если необходимо передать данные между элементами ПО, которые должны быть выполнены независимо, то следует применять однонаправленные интерфейсы, такие, например, как сообщения или каналы, а не совместно используемая память.

Примечание — В идеальном случае независимые элементы не должны взаимодействовать друг с другом. Однако если проект системы требует, чтобы один элемент ПО передавал данные другому элементу, то проект коммуникационного механизма должен быть выбран таким, чтобы отправляющий и/или получающий сообщение элементы не находились в состоянии отказа, или их функционирование не будет заблокировано в случае прекращения или задержки передачи данных.

Кроме переменной информации в оперативной памяти при пространственном разделении должен быть учтен любой резидентный объект данных в постоянных запоминающих устройствах, таких как магнитные диски. Например, защита доступа к файлу, реализованная операционной системой, может быть использована для предотвращения записи одним элементом ПО в принадлежащие другому элементу области данных.

Е.5 Обеспечение временной независимости

Методы, гарантирующие временную независимость, включают в себя:

- детерминированные методы планирования, например применение:

  циклического алгоритма планирования, при котором каждому элементу задают определенный интервал времени, полученный для каждого элемента в результате анализа времени его работы в наихудшем случае, чтобы статически продемонстрировать выполнение требований синхронизации для каждого элемента,

  архитектуры с временным распределением;

- планирование, основанное на строгом приоритете, реализованное программой-диспетчером, работающей в реальном времени, со средствами, запрещающими смену приоритетов;

- временные заграждающие метки, которые завершают выполнение элемента в том случае, если происходит превышение выделенного для него времени выполнения или максимального времени (в этом случае должен быть проведен анализ риска, показывающий, что завершение выполнения элемента не приведет к опасному отказу и, таким образом, этот метод может быть эффективнее всего использован для элемента, не связанного с безопасностью);

- возможности операционной системы, которая запрещает какому-либо процессу применять полностью временные ресурсы процессора, например с помощью квантования времени. Такой подход применим, только если элементы, связанные с безопасностью, не устанавливают жестких требований к режиму реального времени и если показано, что алгоритм планирования не приведет к неоправданным задержкам обращения к какому-либо элементу.

Если элементы ПО совместно используют ресурс (например, периферийное устройство), то в проекте должно быть гарантировано, что элементы будут функционировать корректно, так как совместно используемый ресурс блокируется другим элементом. При определении отсутствия временного взаимовлияния необходимо учитывать время получения доступа к совместно используемому ресурсу.

Е.6 Требования к программному обеспечению поддержки

Если операционная система, программа-диспетчер, работающая в реальном времени, управление памятью, управление таймером или какое-либо подобное ПО должны быть использованы для обеспечения пространственной и/или временной независимости, то в таком ПО любые из его элементов, которые должны быть независимыми, должны обладать наиболее высокой стойкостью к систематическим отказам.

Примечание — Однако в таком ПО для независимых элементов существует возможность отказа по общей причине.

Е.7 Независимость программных модулей. Аспекты языка программирования

Таблица Е.1 содержит неформальные определения используемых терминов.

Таблица Е.1 — Связывание модулей. Определение терминов

Термин	Неформальное определение
Связность	Мера прочности связей между данными и подпрограммами внутри одного программного модуля
Связывание	Мера прочности связей между программными модулями
Инкапсуляция	Ограничение внешнего доступа к внутренним (личным) данным и подпрограммам; термин используют главным образом с объектно-ориентированными программами

Окончание таблицы Е.1

Термин	Неформальное определение
Независимость	Мера отсутствия связей между частями программы; антоним понятия «связывание»
Модуль	Выполняющая некоторую функцию ограниченная часть ПО, которая может иметь собственные данные: класс, иерархию классов, подпрограммы, блок, модуль, пакет и т. п. в соответствии с языком программирования
Интерфейс	Четко определенный набор заголовков подпрограмм, обеспечивающий доступ к программному модулю
Случайные данные («блуждающие данные»)	Полученные данные, не используемые в программном модуле, но передающиеся в другой программный модуль

Как правило, независимость программного модуля увеличивается, если между модулями существует слабое связывание и сильная связность внутри модулей. Сильная связность обеспечивает такую ситуацию, в которой идентифицируемые функциональные модули точно соответствуют идентифицируемым модулям реализации, а слабое связывание модулей обеспечивает незначительное взаимодействие и, следовательно, высокую степень независимости между функционально не связанными модулями.

Слабо связанный программный модуль обычно формируется в результате сильной связности внутри модуля, объединяя вместе код и используемые данные для выполнения одной конкретной функции. Слабая связность формируется в модулях, если код и данные объединены достаточно свободно или в результате некоторой временной последовательности либо некоторой последовательности потока управления.

Необходимо различать виды связывания модулей (см. таблицу Е.2).

При чтении или анализе кода (см. 7.9.2.12) проверяют, насколько слабо связаны программные модули. Такой анализ обычно требует понимания целей модулей и способа их выполнения. Поэтому истинное связывание может быть оценено только после прочтения кода и его документации.

Связывания по контенту необходимо избегать. Глобальное связывание может быть использовано только в исключительных случаях. Связывания по управлению и структурного связывания необходимо избегать. Если возможно, модули должны быть соединены связыванием с помощью интерфейса (инкапсуляцией) и/или связыванием с помощью данных.

Таблица Е.2 — Виды связывания модулей

Связывание	Определение	Объяснение	Обоснование	Примечание
Связывание с помощью интерфейсов, инкапсуляция	Связывание только для четко определенного множества подпрограмм	Доступ к модулю или к его данным только через подпрограммы; любое изменение величины переменной, любой вопрос о величине такой переменной или любой сервис, требуемый от модуля, выполняются через вызов подпрограммы	Заголовки подпрограмм (сигнатуры) модулей объясняют доступные сервисы. Если требуется изменить модуль, то большая часть этих изменений может быть выполнена непосредственно в модуле, не затрагивая другие модули. Поддерживает слабое связывание (в целом рекомендуется)	В основном для объектно-ориентированных программ, классов, иерархии классов, библиотек; не для подпрограмм
Связывание с помощью данных списка параметров	Передача только данных списка параметров или идентификаторов подпрограмм	Доступ к модулю или к его данным только через переменные или объекты, указанные в заголовке подпрограммы; любое изменение величины переменной, любой вопрос о величине такой переменной являются явными	Заголовок подпрограмм содержит данные или объекты, включенные в вызов подпрограммы. Поддерживает слабое связывание (в целом рекомендуется)	Внутри классов объектно-ориентированных программ этот принцип обычно не востребован. К локальным переменным можно получить прямой доступ. Строгое следование этому принципу может также привести к получению случайных данных. Для того чтобы этого избежать, данный принцип должен быть нарушен
Структурное связывание	Передаваемые данные содержат больше данных, чем необходимо	В получающую подпрограмму передается больше данных, чем это необходимо для выполнения требуемой функции	Лишние данные обеспечивают получающий модуль той информацией, которая не нужна для его выполнения. Эти данные могут привести к недоразумениям при взаимодействии модулей, что, однако, допускается	Как правило, этот недостаток может быть легко скорректирован
Связывание по управлению	Связывание, которое осуществляет непосредственное управление получающим модулем	Передача данных, которая может выполнить только передачу управления на другой модуль и во многих случаях характеризуется передачей одного бита	Более тесное связывание, чем предыдущее, так как требует немедленного действия, предписывая получающей подпрограмме что-либо выполнить. Необходимо проявлять осторожность. В целом не рекомендуется	Не всегда можно избежать. Например, передача может быть необходима при завершении действия или подтверждении соответствия значения

Окончание таблицы Е.2

Связывание	Определение	Объяснение	Обоснование	Примечание
Глобальное связывание	Связывание с помощью глобальных данных	Модули могут получить доступ к данным, к которым другие модули имеют прямой доступ, или один модуль может получить прямой доступ к данным, принадлежащим другому модулю	Заголовок подпрограмм не указывает на то, какие данные используются и откуда. Сложно понять функции подпрограмм и определить последствия любых изменений кода	В целом критикуемый вид связывания. Например, может быть востребован исключительно для того, чтобы избежать случайных данных. Необходимо использовать только в строго ограниченных рамках в соответствии с четко определенным и документально оформленным стандартом кодирования
Связывание по контенту	Прямой переход в другие модули, оказывающий влияние на условие в условных операторах в этих модулях, или прямой доступ к данным в других модулях	Реализуется в программах на языке ассемблера; не реализуется на всех языках высокого уровня. Может ускорить выполнение программы и уменьшить трудоемкость кодирования	Критикуемый вид связывания. Отдельный модуль можно понять только на том уровне, на котором понятны все соединенные с ним модули. Программа становится чрезвычайно сложной для понимания и изменения	В некоторых языках программирования связывание по контенту даже невозможно. Допускается всегда игнорировать

Приложение Ж
(справочное)

Руководство по адаптации жизненного цикла систем, управляемых данными

Ж.1 Система, управляемая данными. Системная и прикладная части

Многие системы состоят из двух частей. Одна часть является системной, другая часть — прикладной, которая адаптирует систему к конкретным требованиям необходимого применения. Прикладная часть может быть создана в виде данных, которые конфигурируют системную часть. В настоящем приложении такие системы именуют «управляемыми данными».

Конкретная прикладная часть такого ПО может быть разработана с использованием множества программных средств и языков программирования. Эти средства и языки могут ограничить способ создания прикладной программы.

Например, если язык программирования достаточно доступно описывает функциональность (например, язык многозвенных логических схем для простых систем блокировки), то прикладное ПО для запрограммированной задачи с высокой степенью вероятности будет простым. Однако если язык программирования описывает поведение приложения достаточно сложно, то прикладное ПО для запрограммированной задачи, вероятно, будет сложным. Если разработано очень простое прикладное ПО, то детальное проектирование можно выполнять как конфигурирование, а не как программирование.

Степень строгости, необходимой для достижения требуемой полноты безопасности, зависит от степени сложности конфигурации, поддерживаемой средствами разработки/конфигурирования, и сложности представляемого поведения применения (см. рисунок Ж.1).

Рисунок Ж.1 — Изменчивость языка и сложность систем, управляемых данными

На рисунке Ж.1 по осям откладывают классы сложности:

- для изменчивости языка:
  - фиксированная программа,
  - ограниченная изменчивость (в некоторых отраслях прикладную программу рассматривают как данные, которые интерпретируются системной частью),
  - полная изменчивость (обычно не рассматриваемый как управляемый данными такой тип систем может быть также использован для разработки применений и включен в настоящее приложение для полноты картины);
- возможной конфигурируемости применения:
  - ограниченная конфигурируемость,
  - полная конфигурируемость.

В действительности конкретная система может включать в себя разные уровни сложности и конфигурируемости. Кроме того, сложность можно представить как плавную шкалу с непрерывно изменяющимися значениями по осям на рисунке Ж.1. Если необходимо адаптировать ЖЦ ПО, то должен быть идентифицирован соответствующий уровень сложности и должна быть обоснована степень адаптации.

Ниже приведено описание различных типов систем для каждого уровня сложности. Указания по предполагаемым методам для реализации каждого типа системы приведены в ГОСТ 34332.3.

Типичные системы для каждого класса сложности описаны в Ж.2.

Ж.2 Конфигурация с ограниченной изменчивостью, ограниченная конфигурируемость применения

Для систем, соответствующих требованиям ГОСТ 34332, используют патентованный язык конфигурации с предварительно установленной поставляемой функциональностью.

Такой язык конфигурации не дает возможности программисту изменять функцию системы, а конфигурирование ограничено настройкой нескольких параметров, позволяющей системе соответствовать ее применению. Например, интеллектуальные датчики и приводы, сетевые контроллеры, контроллеры последовательности, небольшие системы регистрации данных и интеллектуальные инструменты настраиваются введением в них конкретных значений параметров.

В обоснование адаптации ЖЦ Э/Э/ПЭ СБЗС системы включают (как минимум):

- спецификацию входных параметров для данного применения;
- верификацию правильности реализации параметров в действующей системе;
- подтверждение соответствия всех комбинаций входных параметров;
- рассмотрение специальных и конкретных режимов работы в процессе конфигурирования;
- человеческий фактор/эргономику;
- блокировки, например обеспечение гарантии того, что блокировки выполнения прошли подтверждение соответствия во время процесса конфигурации;
- непреднамеренное реконфигурирование, например, доступа к ключу переключения устройств защиты.

Ж.3 Конфигурация на языке с ограниченной изменчивостью, полная конфигурируемость применения

Для систем, соответствующих требованиям ГОСТ 34332, используют патентованный язык конфигурации с предварительно установленной поставляемой функциональностью.

Такой язык конфигурации не дает возможности программисту изменять функцию системы, а конфигурирование ограничено созданием обширных данных статических параметров, позволяющих системе соответствовать ее применению. Примером может служить система авиадиспетчерской службы, состоящая из данных с большим числом сущностей данных, каждая из которых имеет один атрибут или более. Существенной характеристикой этих данных является то, что они не содержат явных последовательностей, упорядочиваний или ветвлений и представления комбинаторных состояний применения.

В обоснование адаптации ЖЦ Э/Э/ПЭ СБЗС системы кроме рассмотренного в Ж.2 следует включать (как минимум):

- средства автоматизации для создания данных;
- проверку непротиворечивости, например на самосовместимость данных;
- проверку правил, например для гарантирования генерации данных, удовлетворяющих определенным ограничениям;
- подтверждение соответствия интерфейсов системам подготовки данных.

Ж.4 Конфигурация на языке с ограниченной изменчивостью, ограниченная конфигурируемость применения

Для систем, соответствующих требованиям ГОСТ 34332, используют проблемно-ориентированные языки с ограниченной, предварительно установленной, поставляемой функциональностью, в которых операторы языка содержат или напоминают терминологию пользователя применения.

Эти языки позволяют пользователям с ограниченной изменчивостью настраивать функции системы в соответствии с собственными конкретными требованиями, используя ряд аппаратных и программных элементов.

Существенной характеристикой языка с ограниченной изменчивостью является то, что данные могут содержать явные последовательности, упорядочивания или ветвления и вызывать комбинаторные состояния применения, например: программирование на основе функциональных блоков, многозвенные логические схемы, системы, основанные на крупноформатной таблице, и графические системы.

В обоснование адаптации ЖЦ Э/Э/ПЭ СБЗС системы кроме рассмотренного в подразделе Ж.3 включают (как минимум):

- спецификацию основных требований применения;
- допускаемые подмножества языка для этого применения;
- методы разработки для объединения подмножеств языка;
- критерии охвата проверкой решений для комбинаций возможных системных состояний.

Ж.5 Конфигурация на языке с ограниченной изменчивостью, полная конфигурируемость применения

Для систем, соответствующих требованиям ГОСТ 34332, используют проблемно-ориентированный язык с ограниченной, предварительно установленной, поставляемой функциональностью, в котором операторы языка содержат или напоминают терминологию пользователя применения.

Существенным отличием применения языка с ограниченной изменчивостью при полной конфигурируемости применения от применения языка с ограниченной изменчивостью при ограниченной конфигурируемости применения является сложность конфигурации применения. Примерами таких применений могут служить графические системы и системы управления серийного производства на основе SCADA-систем.

Примечание — SCADA-система (от английского supervisory control and data acquisition system) — это система диспетчерского управления и сбора данных.

В обоснование адаптации ЖЦ Э/Э/ПЭ СБЗС системы кроме рассмотренного в подразделе Ж.4 включают (как минимум):

- проект архитектуры применения;
- обеспечение шаблонов;
- верификацию отдельных шаблонов;
- верификацию и подтверждение соответствия применения.

При реализации и тестировании модулей наиболее низкого уровня проблема ЖЦ, рассматриваемая в настоящем стандарте, вероятно, будет неактуальной (в зависимости от используемого языка).

Ж.6 Конфигурация на языке с полной изменчивостью, полная конфигурируемость применения

См. раздел Ж.7.

Ж.7 Конфигурация на языке с полной изменчивостью, полная конфигурируемость применения

Для систем с такой конфигурацией применяют требования настоящего стандарта в течение всего ЖЦ.

Части систем с полной изменчивостью разрабатывают на универсальных языках программирования или на универсальных языках базы данных либо с использованием универсальных научных пакетов и пакетов моделирования. Как правило, эти части выполняют в компьютерной системе под управлением операционной системы, которая управляет выделением системных ресурсов и средой мультипрограммирования реального времени. Например, системы, написанные на языках с полной изменчивостью, могут включать в себя специализированные системы управления оборудованием, специально разработанные системы управления полетом или веб-сервисы для управления службами, связанными с безопасностью.

Библиография

[1]	Технический регламент Таможенного союза ТР ТС 002/2011	О безопасности высокоскоростного железнодорожного транспорта
[2]	Технический регламент Таможенного союза ТР ТС 003/2011	О безопасности инфраструктуры железнодорожного транспорта
[3]	Технический регламент Таможенного союза ТР ТС 014/2011	О безопасности дорог
[4]	МЭК 61508-3	Функциональная безопасность электрических, электронных, программируемых электронных систем, связанных с безопасностью. Часть 3. Требования к программному обеспечению (Functional safety of electrical/electronic/programmable electronic safety-related systems — Part 3: Requirements for software)
[5]	Руководство ИСО/МЭК 51:2014	Аспекты безопасности. Руководящие указания по их включению в стандарты (Safety aspects: Guidelines for their inclusion in standards)

УДК 621.5:814.8:006.354

МКС 13.100
13.110
13.200
13.220
13.310
13.320
91.120.99

NEQ

Ключевые слова: системы, связанные с безопасностью зданий и сооружений; программное обеспечение; функциональная безопасность систем, связанных с безопасностью зданий и сооружений; программное обеспечение, связанное с безопасностью зданий и сооружений; требования к программному обеспечению

Редактор *Л.С. Зимилова*
Технический редактор *И.Е. Черепкова*
Корректор *Л.С. Лысенко*
Компьютерная верстка *Л.А. Круговой*

Сдано в набор 28.05.2021. Подписано в печать 24.06.2021. Формат 60×84⅛. Гарнитура Ариал.
Усл. печ. л. 11,63. Уч.-изд. л. 10,47.

Подготовлено на основе электронной версии, предоставленной разработчиком стандарта

Создано в единичном исполнении во ФГУП «СТАНДАРТИНФОРМ»
для комплектования Федерального информационного фонда стандартов,
117418 Москва, Нахимовский пр-т, д. 31, к. 2.
www.gostinfo.ru info@gostinfo.ru

Поправка к ГОСТ 34332.4—2021 Безопасность функциональная систем, связанных с безопасностью зданий и сооружений. Часть 4. Требования к программному обеспечению

В каком месте	Напечатано	Должно быть		
Предисловие. Таблица согласования	—	Казахстан	KZ	Госстандарт Республики Казахстан

(ИУС № 4 2022 г.)