ФЕДЕРАЛЬНОЕ АГЕНТСТВО

ПО ТЕХНИЧЕСКОМУ РЕГУЛИРОВАНИЮ И МЕТРОЛОГИИ

НАЦИОНАЛЬНЫЙ СТАНДАРТ РОССИЙСКОЙ ФЕДЕРАЦИИ

ГОСТ Р 53195.4— 2010

БЕЗОПАСНОСТЬ ФУНКЦИОНАЛЬНАЯ СВЯЗАННЫХ С БЕЗОПАСНОСТЬЮ ЗДАНИЙ И СООРУЖЕНИЙ СИСТЕМ

Часть 4

Требования к программному обеспечению

(IEC 61508-3:2010, NEQ)
(IEC 61508-4:2010, NEQ)
(ISO/IEC Guide 51:1999, NEQ)

Издание официальное

Москва
Стандартинформ
2018

gost.e-ecolog.ru

Предисловие

1 РАЗРАБОТАН Университетом комплексных систем безопасности и инженерного обеспечения

2 ВНЕСЕН Техническим комитетом по стандартизации ТК 439 «Средства автоматизации и системы управления» при поддержке Технического комитета по стандартизации ТК 465 «Строительство»

3 УТВЕРЖДЕН И ВВЕДЕН В ДЕЙСТВИЕ Приказом Федерального агентства по техническому регулированию и метрологии от 21 декабря 2010 г. № 820-ст

4 В настоящем стандарте учтены основные нормативные положения следующих международных стандартов:

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

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

Руководство ИСО/МЭК 51:1999 «Аспекты безопасности. Руководящие указания по включению их в стандарты» (ISO/IEC Guide 51:1999 «Safety aspects — Guidelines for their inclusion in standards», NEQ)

5 ВВЕДЕН ВПЕРВЫЕ

6 ПЕРЕИЗДАНИЕ. Октябрь 2018 г.

*Правила применения настоящего стандарта установлены в статье 26 Федерального закона от 29 июня 2015 г. № 162-ФЗ «О стандартизации в Российской Федерации». Информация об изменениях к настоящему стандарту публикуется в ежегодном (по состоянию на 1 января текущего года) информационном указателе «Национальные стандарты», а официальный текст изменений и поправок — в ежемесячном информационном указателе «Национальные стандарты». В случае пересмотра (замены) или отмены настоящего стандарта соответствующее уведомление будет опубликовано в ближайшем выпуске ежемесячного информационного указателя «Национальные стандарты». Соответствующая информация, уведомление и тексты размещаются также в информационной системе общего пользования — на официальном сайте Федерального агентства по техническому регулированию и метрологии в сети Интернет (*gost.e-ecolog.ru

© Стандартинформ, оформление, 2018

Настоящий стандарт не может быть полностью или частично воспроизведен, тиражирован и распространен в качестве официального издания без разрешения Федерального агентства по техническому регулированию и метрологии

Содержание

1 Область применения 1
2 Нормативные ссылки 2
3 Термины и определения 3
4 Обозначения и сокращения 4
5 Требования 4
5.1 Соответствие требованиям стандарта 4
5.2 Требования к документации 4
5.3 Требования к управлению функциональной безопасностью 4
5.4 Общие требования к СБЗС ПО 4
5.5 Требования к жизненному циклу СБЗС ПО 5
5.6 Требования к обеспечению безопасности ПО 7
5.7 Ввод в действие, эксплуатация и модификация ПО 18
5.8 Верификация ПО 21
6 Оценка функциональной безопасности 24
Приложение А (справочное) Руководство по выбору методов и средств 25
Приложение Б (справочное) Подробные таблицы 31
Библиография 35

Введение

Современные здания и сооружения — объекты капитального строительства — представляют собой сложные системы, включающие в свой состав систему конструкций и ряд систем в разных сочетаниях, в том числе инженерные системы жизнеобеспечения, реализации технологических процессов, энерго-, ресурсосбережения, безопасности и другие системы. Эти системы взаимодействуют друг с другом, с внешней и внутренней средами.

Объекты капитального строительства жестко привязаны к 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 уровня приемлемого риска и поддержания его в течение периода эксплуатации или использования этих объектов. К средствам снижения риска относятся системы, связанные с безопасностью зданий и сооружений. Эти системы, состоящие из электрических, и/или электронных компонентов, и/или программируемых электронных компонентов, в течение многих лет используются для выполнения функций безопасности. Для решения задач безопасности зданий и сооружений во все бóльших объемах используются программируемые электронные (в том числе компьютерные) системы.

Настоящий стандарт устанавливает основные требования к функциональной безопасности программного обеспечения программируемых электронных систем, связанных с безопасностью зданий и сооружений, и к программному обеспечению, используемому для разработки таких систем в рамках области применения ГОСТ Р 53195.1, ГОСТ Р 53195.2 и ГОСТ Р 53195.3.

Стандарт устанавливает требования к действиям и процедурам, которые должны быть выполнены на стадиях жизненного цикла программного обеспечения этих систем для достижения и поддержания их функциональной безопасности.

Настоящий стандарт входит в комплекс стандартов с наименованием «Безопасность функциональная связанных с безопасностью зданий и сооружений систем» 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;

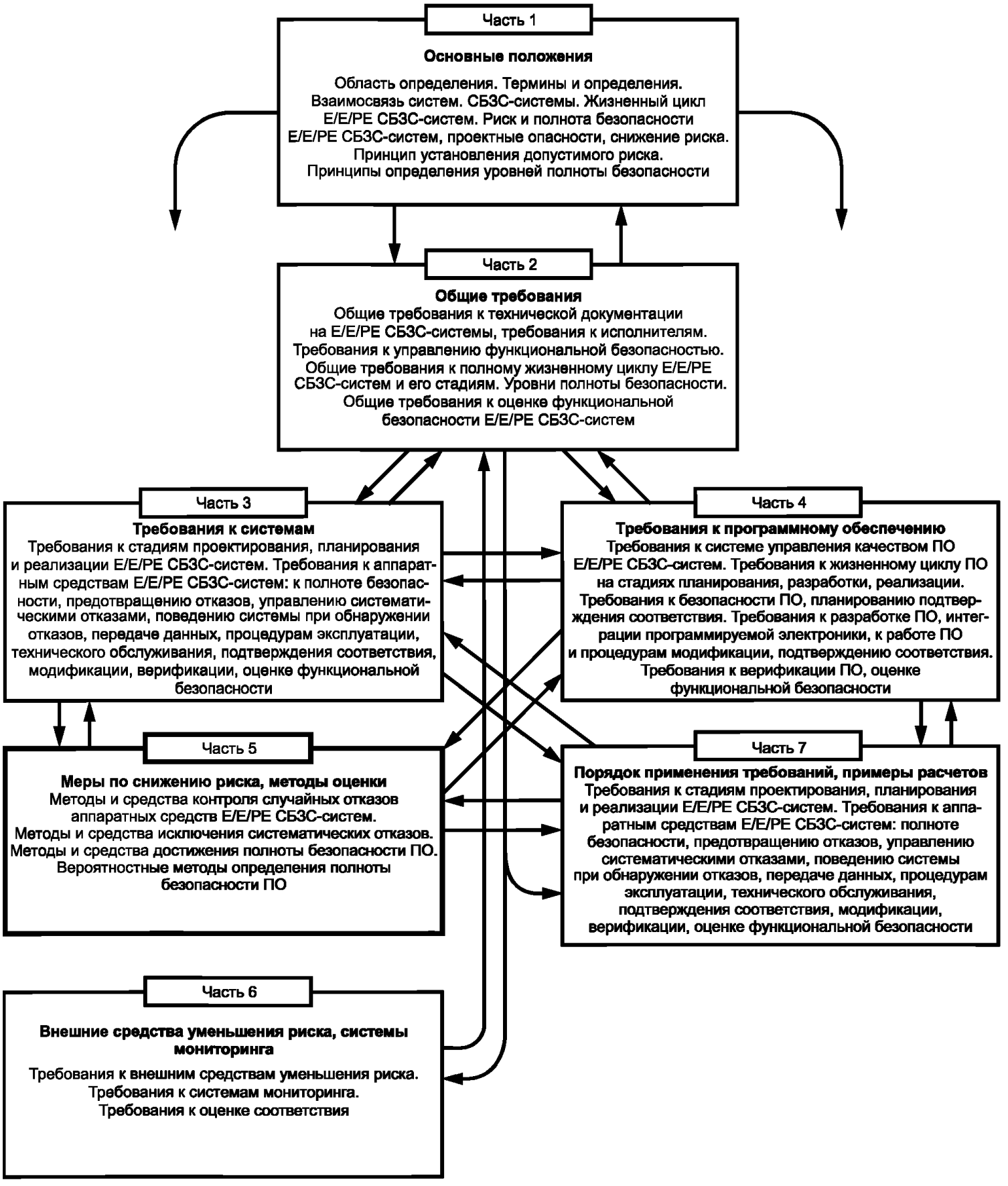
Часть 3. Требования к системам;

Часть 5. Меры по снижению риска, методы оценки;

Часть 6. Внешние средства уменьшения риска, системы мониторинга;

Часть 7. Порядок применения требований, примеры расчетов.

Структура комплекса стандартов приведена ниже.


Часть 1
Основные положения
Область определения. Термины и определения. Взаимосвязь систем. СБЗС-системы. Жизненный цикл E/E/PE СБЗС-систем. Риск и полнота безопасности E/E/PE СБЗС-систем, проектные опасности, снижение риска. Принцип установления допустимого риска. Принципы определения уровней полноты безопасности
Часть 2
Общие требования
Общие требования к технической документации на E/E/PE СБЗС-системы, требования к исполнителям. Требования к управлению функциональной безопасностью. Общие требования к полному жизненному циклу E/E/PE СБЗС-систем и его стадиям. Уровни полноты безопасности. Общие требования к оценке функциональной безопасности E/E/PE СБЗС-систем
Часть 3
Требования к системам
Требования к стадиям проектирования, планирования и реализации E/E/PE СБЗС-систем. Требования к аппаратным средствам E/E/PE СБЗС-систем: к полноте безопасности, предотвращению отказов, управлению систематическими отказами, поведению системы при обнаружении отказов, передаче данных, процедурам эксплуатации, технического обслуживания, подтверждения соответствия, модификации, верификации, оценке функциональной безопасности
Часть 4
Требования к программному обеспечению
Требования к системе управления качеством ПО E/E/PE СБЗС-систем. Требования к жизненному циклу ПО на стадиях планирования, разработки, реализации. Требования к безопасности ПО, планированию подтверждения соответствия. Требования к разработке ПО, интеграции программируемой электроники, к работе ПО и процедурам модификации, подтверждению соответствия. Требования к верификации ПО, оценке функциональной безопасности
Часть 5
Меры по снижению риска, методы оценки
Методы и средства контроля случайных отказов аппаратных средств E/E/PE СБЗС-систем. Методы и средства исключения систематических отказов. Методы и средства достижения полноты безопасности ПО. Вероятностные методы определения полноты безопасности ПО
Часть 7
Порядок применения требований, примеры расчетов
Требования к стадиям проектирования, планирования и реализации E/E/PE СБЗС-систем. Требования к аппаратным средствам E/E/PE СБЗС-систем: полноте безопасности, предотвращению отказов, управлению систематическими отказами, поведению системы при обнаружении отказов, передаче данных, процедурам эксплуатации, технического обслуживания, подтверждения соответствия, модификации, верификации, оценке функциональной безопасности
Часть 6
Внешние средства уменьшения риска, системы мониторинга
Требования к внешним средствам уменьшения риска. Требования к системам мониторинга. Требования к оценке соответствия

НАЦИОНАЛЬНЫЙ СТАНДАРТ РОССИЙСКОЙ ФЕДЕРАЦИИ

БЕЗОПАСНОСТЬ ФУНКЦИОНАЛЬНАЯ СВЯЗАННЫХ С БЕЗОПАСНОСТЬЮ ЗДАНИЙ И СООРУЖЕНИЙ СИСТЕМ

Часть 4

Требования к программному обеспечению

Functional safety of building/erection safety-related systems. Part 4. Software requirements

Дата введения — 2012—01—01

1 Область применения

Настоящий стандарт распространяется на:

- программное обеспечение (далее — ПО) программируемых электронных связанных с безопасностью зданий и сооружений систем (далее — E/E/PE СБЗС-систем), в дальнейшем именуемое СБЗС ПО, а также на системы, подсистемы и компоненты внутри E/E/PE СБЗС-систем, которые содержат хотя бы один программируемый электронный компонент;

- любое программное обеспечение, являющееся частью СБЗС-системы либо используемое для разработки системы, связанной с безопасностью, в рамках области применения ГОСТ Р 53195.1, ГОСТ Р 53195.2 и ГОСТ Р 53195.3. Такое программное обеспечение называется программным обеспечением систем, связанных с безопасностью зданий и сооружений (далее — СБЗС ПО). СБЗС ПО включает в себя операционные системы, системное программное обеспечение, программы, используемые в коммуникационных сетях, интерфейсы пользователей и обслуживающего персонала, инструментальные средства поддержки, встроенные программно-аппаратные средства, а также прикладные программы. Прикладные программы включают в себя программы высокого и низкого уровней, а также специальные программы на языках с ограниченной варьируемостью (см. 3.12) [1].

Настоящий стандарт устанавливает:

- требования к стадиям жизненного цикла СБЗС ПО и действиям, которые должны предприниматься на этих стадиях во избежание ошибок и отказов СБЗС ПО и для принятия необходимых мер при их возникновении;

- минимальный состав информации, относящейся к подтверждению безопасности СБЗС ПО, необходимой для установки, ввода в действие, интеграции и подтверждения соответствия E/E/PE СБЗС-систем требованиям безопасности;

- требования к подготовке информации и процедурам, относящимся к СБЗС ПО, необходимым пользователю для работы и поддержания E/E/PE СБЗС-систем в период эксплуатации;

- требования, предъявляемые к действиям при выполнении модификации СБЗС ПО;

- совместно с ГОСТ Р 53195.1, ГОСТ Р 53195.2, ГОСТ Р 53195.3 и ГОСТ Р 53195.5 требования к инструментальным средствам поддержки.

Примечание — Области применения ГОСТ Р 53195.3 и ГОСТ Р 53195.4 взаимосвязаны между собой (см. рисунок 1).

Настоящий стандарт не распространяется на ПО одиночных СБЗС-систем, способных осуществить необходимое снижение риска, и требуемая полнота безопасности которых ниже самого низкого уровня полноты безопасности (SIL1), определенного в таблицах 1 и 2 ГОСТ Р 53195.2—2008.

Настоящий стандарт должен применяться совместно с ГОСТ Р 53195.1, ГОСТ Р 53195.2, ГОСТ Р 53195.3 и ГОСТ Р 53195.5.

Издание официальное

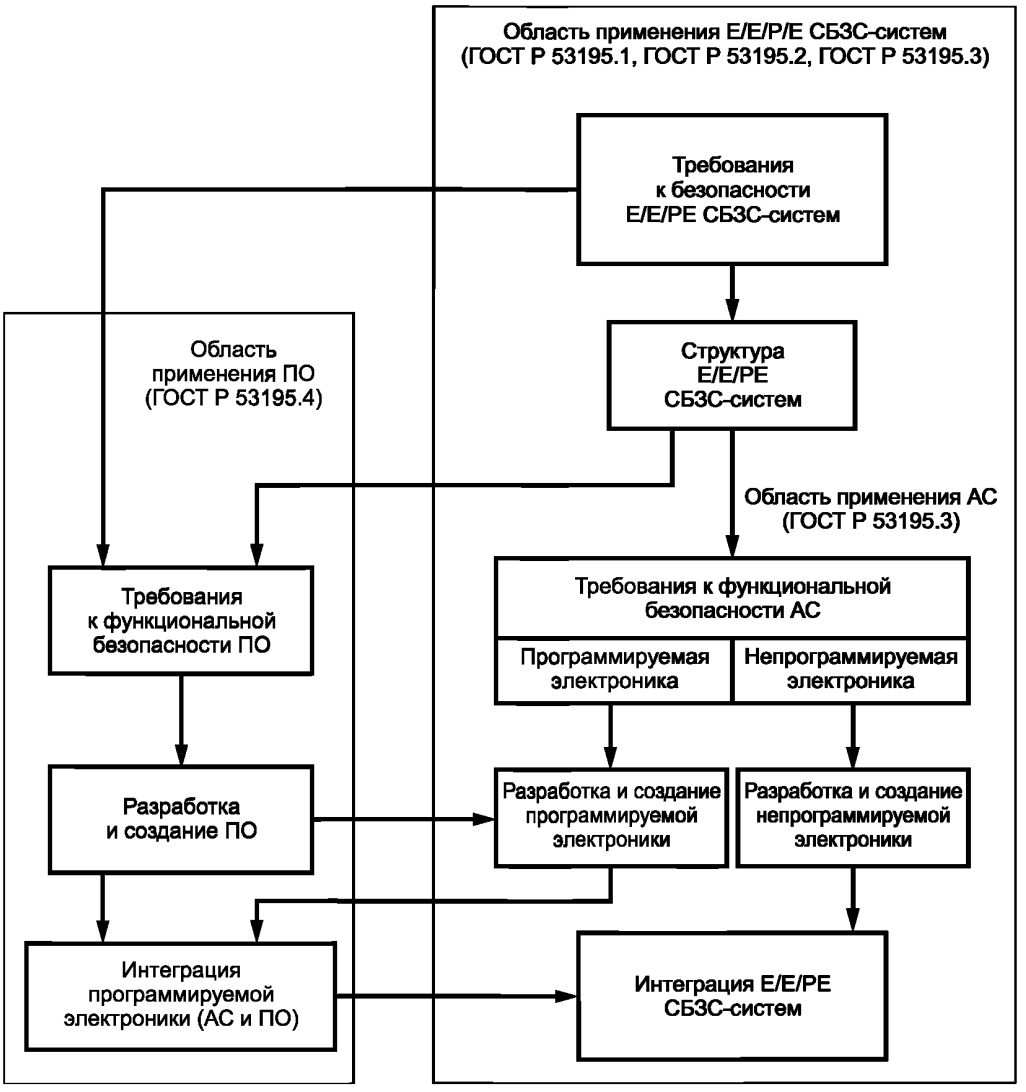


Рисунок 1 — Взаимосвязь областей применения АС и ПО

2 Нормативные ссылки

В настоящем стандарте использованы нормативные ссылки на следующие стандарты:

ГОСТ Р 53195.1 Безопасность функциональная связанных с безопасностью зданий и сооружений систем. Часть 1. Основные положения

ГОСТ Р 53195.2—2008 Безопасность функциональная связанных с безопасностью зданий и сооружений систем. Часть 2. Общие требования

ГОСТ Р 53195.3 Безопасность функциональная связанных с безопасностью зданий и сооружений систем. Часть 3. Требования к системам

ГОСТ Р 53195.5—2010 Безопасность функциональная связанных с безопасностью зданий и сооружений систем. Часть 5. Меры по снижению риска, методы оценки

Примечание — При пользовании настоящим стандартом целесообразно проверить действие ссылочных стандартов в информационной системе общего пользования — на официальном сайте Федерального агентства по техническому регулированию и метрологии в сети Интернет или по ежегодному информационному указателю «Национальные стандарты», который опубликован по состоянию на 1 января текущего года, и по выпускам ежемесячного информационного указателя «Национальные стандарты» за текущий год. Если заменен ссылочный стандарт, на который дана недатированная ссылка, то рекомендуется использовать действующую версию этого стандарта с учетом всех внесенных в данную версию изменений. Если заменен ссылочный стандарт, на который дана дати-

рованная ссылка, то рекомендуется использовать версию этого стандарта с указанным выше годом утверждения (принятия). Если после утверждения настоящего стандарта в ссылочный стандарт, на который дана датированная ссылка, внесено изменение, затрагивающее положение, на которое дана ссылка, то это положение рекомендуется применять без учета данного изменения. Если ссылочный стандарт отменен без замены, то положение, в котором дана ссылка на него, рекомендуется применять в части, не затрагивающей эту ссылку.

3 Термины и определения

В настоящем стандарте применены термины по ГОСТ Р 53195.1, ГОСТ Р 53195.2, ГОСТ Р 53195.3 и ГОСТ Р 53195.5, а также следующие термины с соответствующими определениями.

3.1 **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.

3.2 **динамическое тестирование** (dynamic testing): Работа программного обеспечения и/или работа аппаратного средства, выполняемая под контролем и планомерно для демонстрации наличия требуемого поведения и отсутствия нежелательного поведения.

Примечание — Динамическое тестирование представляет собой противоположность статическому анализу, при котором не требуется выполнение программ.

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

3.4 **избыточность** (redundancy): Наличие средств в дополнение к средствам, которые могут быть достаточны функциональному блоку, для выполнения требуемой операции или данным для представления информации.

Пример — Примерами избыточности являются дублирование функциональных компонентов и добавление битов четности.

3.5 **инспекция программы по Файгану** (Fagan inspection): Один из методов обнаружения ошибок на ранних этапах разработки программного обеспечения рабочей группой (при подготовке требований, проектировании, начальных этапах кодирования, планировании тестов), основанный на тщательном анализе первичных документов и проверке соответствия им вторичных документов.

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

3.7 **полнота безопасности программного обеспечения** (software safety integrity): Количественная характеристика, которая означает вероятность того, что программное обеспечение программируемой электронной системы будет выполнять заданные функции безопасности при всех оговоренных условиях в течение установленного периода времени.

3.8 **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.

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

3.10 **уровень полноты безопасности программного обеспечения** (software safety integrity level): Дискретный уровень (принимающий одно из четырех возможных значений от 1 до 4), определяющий полноту безопасности программного обеспечения в связанной с безопасностью системе.

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

3.12 **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 изменчивостью, которые используются 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, чтение 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.

4 Обозначения и сокращения

В настоящем стандарте приняты следующие обозначения и сокращения:

АРМ — автоматизированное рабочее место;

АС — аппаратное(ые) средство(а);

ПЗУ — постоянное запоминающее устройство;

ПЛК — программируемый логический контроллер;

ПО — программное обеспечение;

СБЗС ПО — программное обеспечение системы, связанной с безопасностью зданий и сооружений;

СБЗС-система — система, связанная с безопасностью зданий и сооружений;

УО — управляемое оборудование;

E/E/PE — электрическая, и/или электронная, и/или программируемая электронная (в отношении системы);

PE — программируемая электроника (программируемое электронное средство, программируемая электронная система);

SIL — обозначение уровня полноты безопасности.

5 Требования

5.1 Соответствие требованиям стандарта

Признание соответствия СБЗС ПО требованиям настоящего стандарта — по ГОСТ Р 53195.2—2008 (см. 5.1).

5.2 Требования к документации

Требования к документации E/E/PE СБЗС-систем — по ГОСТ Р 53195.2—2008 (см. 5.2).

5.3 Требования к управлению функциональной безопасностью

Требования к управлению функциональной безопасностью E/E/PE СБЗС-систем — по ГОСТ Р 53195.2—2008 (см. раздел 6).

5.4 Общие требования к СБЗС ПО

5.4.1 Основные цели и требования для полного жизненного цикла СБЗС ПО установлены в ГОСТ Р 53195.2. Дополнительно к ним к СБЗС ПО предъявляются следующие требования.

5.4.2 Жизненный цикл систем безопасности при разработке ПО должен быть выбран и специфицирован при планировании безопасности в соответствии с ГОСТ Р 53195.2—2008 (раздел 6). На стадии

планирования функциональной безопасности СБЗС ПО (см. блок 1.2 на рисунке 3) должны быть определены стратегия поставки, разработки, интеграции, верификации, приемки и модификации ПО в той мере, в какой этого требует уровень полноты безопасности E/E/PE СБЗС-системы.

5.4.3 При совместном использовании компонентов E/E/PE СБЗС-систем, имеющих разные уровни полноты безопасности, следует учитывать требования, установленные в 5.6.4.9.

5.4.4 Система управления конфигурацией СБЗС ПО должна:

- использовать административные и технические средства контроля на протяжении всего жизненного цикла ПО для управления изменениями в программах и гарантирования выполнения указанных в спецификациях требований к безопасности СБЗС ПО;
- гарантировать выполнение всех операций, необходимых для достижения заданной полноты безопасности СБЗС ПО;
- обеспечивать точную поддержку E/E/PE СБЗС-систем с использованием уникальной идентификации всех элементов конфигурации, необходимых для обеспечения полноты безопасности. Элементы конфигурации должны включать в себя, как минимум, следующее:
- анализ безопасности и требования к безопасности;
- спецификацию ПО и проектные документы;
- исходный текст программ;
- план и результаты тестирования;
- ранее разработанные программные компоненты и пакеты, которые должны быть включены в E/E/PE СБЗС-систему;
- все инструментальные средства и системы разработки, используемые при создании, тестировании или выполнении иных действий с СБЗС ПО;
- использовать процедуры контроля над внесением изменений для предотвращения несанкционированных модификаций.

Примечание — Для осуществления руководства и применения административных и технических средств контроля необходимо принятие управленческих решений и наличие полномочий;

- обеспечивать документирование запросов на выполнение модификаций;
- обеспечивать анализ влияния предлагаемых модификаций и предусматривать утверждение либо отклонение модификации;
- обеспечивать подробное документирование модификации и выдачу полномочий на выполнение всех утвержденных модификаций;
- обеспечивать установление основных параметров конфигурации системы для стадий разработки СБЗС ПО и документирование результатов тестирования интеграции системы, которое подтверждает достижение целей стадии (см. 5.8);
- гарантировать объединение и встраивание всех подсистем ПО (включая переработку более ранних версий);
- предусматривать документирование перечисленной выше информации для обеспечения возможности последующего аудита: состояние конфигурации, текущее состояние системы, обоснование и утверждение всех модификаций, подробное описание всех модификаций;
- обеспечивать строгое документирование каждой версии СБЗС ПО, хранение всех версий ПО и всей относящейся к ним документации для обеспечения возможности сопровождения и выполнения модификаций на протяжении всего периода использования разработанного программного продукта.

5.5 Требования к жизненному циклу СБЗС ПО

5.5.1 Процесс разработки СБЗС ПО должен быть разделен на отдельные стадии, этапы и подпроцессы (см. рисунки 2—5).

5.5.2 При разработке СБЗС ПО жизненный цикл должен быть выбран и специфицирован при планировании безопасности СБЗС-систем в соответствии с требованиями ГОСТ Р 53195.2—2008 (раздел 6).

Примечание — Модель жизненного цикла E/E/PE СБЗС-систем, удовлетворяющая требованиям ГОСТ Р 53195.2—2008 (раздел 7), может быть переработана в соответствии с конкретными потребностями проекта или организации.

5.5.3 Процедуры оценки качества и безопасности СБЗС ПО должны быть интегрированы в процессы жизненного цикла E/E/PE СБЗС-систем.

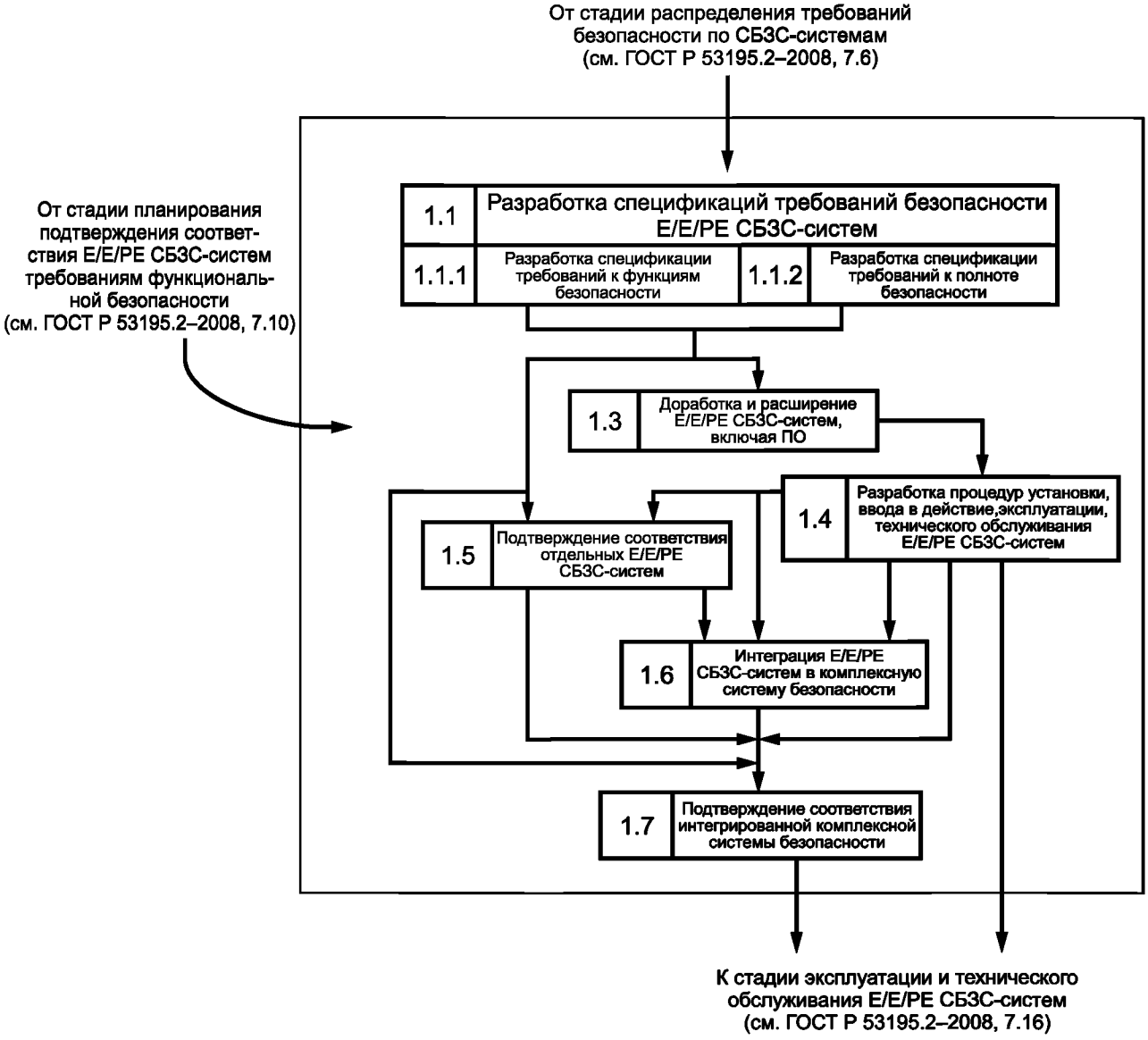


Рисунок 2 — Детализация стадии реализации жизненного цикла E/E/PE СБЗС-системы

5.5.4 Каждая стадия жизненного цикла СБЗС ПО должна быть разделена на элементарные процессы. Для каждой стадии должны быть определены область применения, входные данные и выходные данные.

Примечание — При совместном использовании компонентов E/E/PE СБЗС-систем, имеющих разные уровни полноты безопасности, следует учитывать требования 5.6.4.9.

5.5.5 Выходные данные стадии жизненного цикла СБЗС-системы, включая СБЗС ПО, должны соответствовать назначению системы и ПО.

Примечание — В случае относительно простых систем допускается объединение некоторых стадий жизненного цикла.

5.5.6 Если жизненный цикл СБЗС ПО удовлетворяет требованиям, приведенным на рисунке 4 и в таблице А.1 приложения А, допускается изменять глубину, число и размер этапов и процессов в зависимости от сложности проекта и требуемой полноты безопасности.

5.5.7 Для каждой стадии жизненного цикла следует использовать соответствующие методы/средства, приведенные в приложениях А и Б.

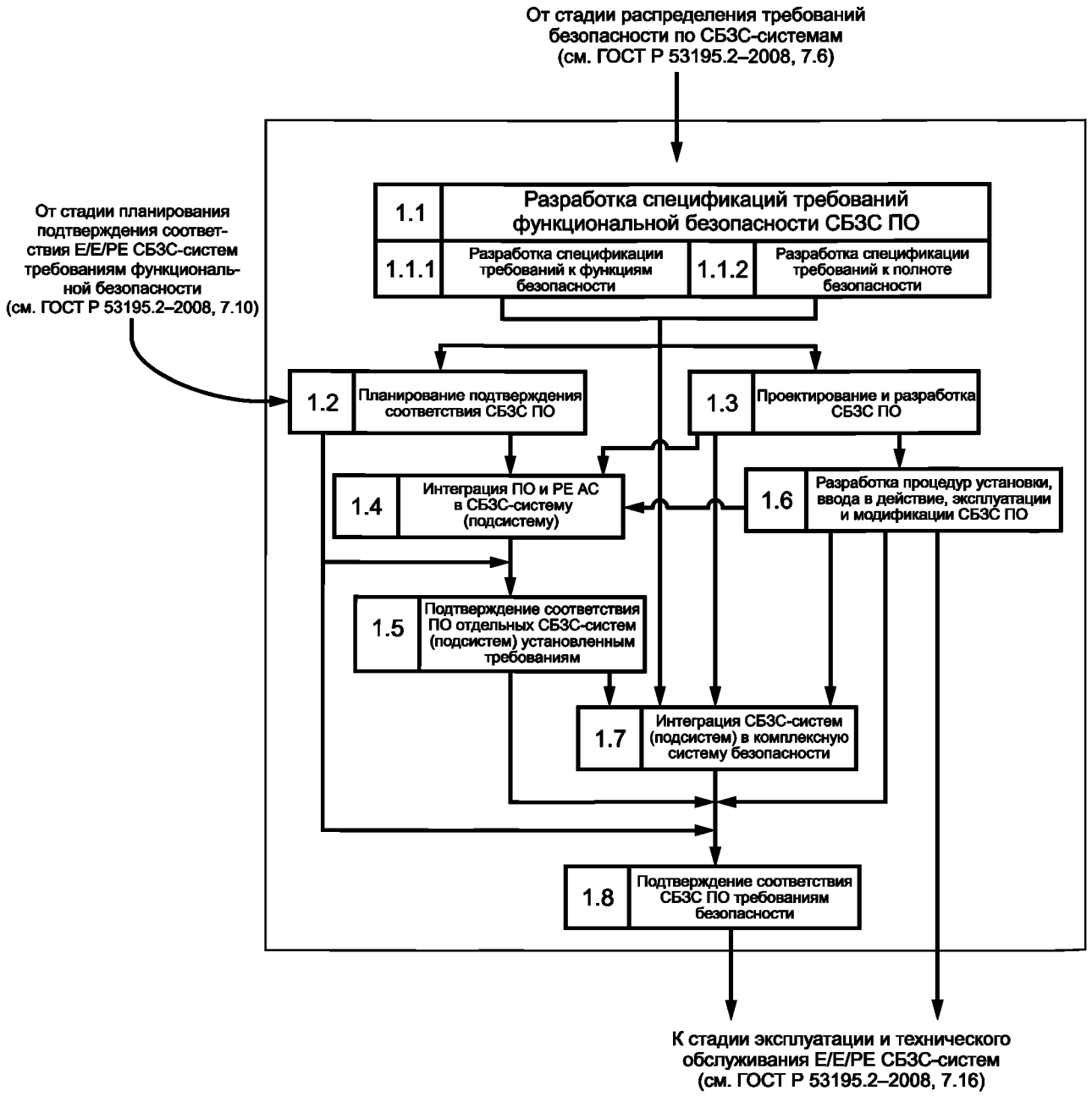


Рисунок 3 — Детализация стадии реализации жизненного цикла СБЗС ПО

Примечание — Применение методов/средств, выбранных из приложений А и Б, само по себе не гарантирует достижения необходимой полноты безопасности.

5.5.8 Результаты процессов жизненного цикла СБЗС ПО должны быть документированы (см. 5.2).

5.5.9 Если на какой-либо стадии жизненного цикла СБЗС ПО возникает необходимость внесения изменений, относящихся к более ранней стадии жизненного цикла, то следует повторно выполнить эту стадию и все последующие стадии.

5.6 Требования к обеспечению безопасности ПО

5.6.1 Спецификация требований к безопасности СБЗС ПО (см. блок 1.1 на рисунке 3) разрабатывается в целях:

- установления требований к функциям безопасности ПО как требований к функциям безопасности и требований к полноте безопасности;

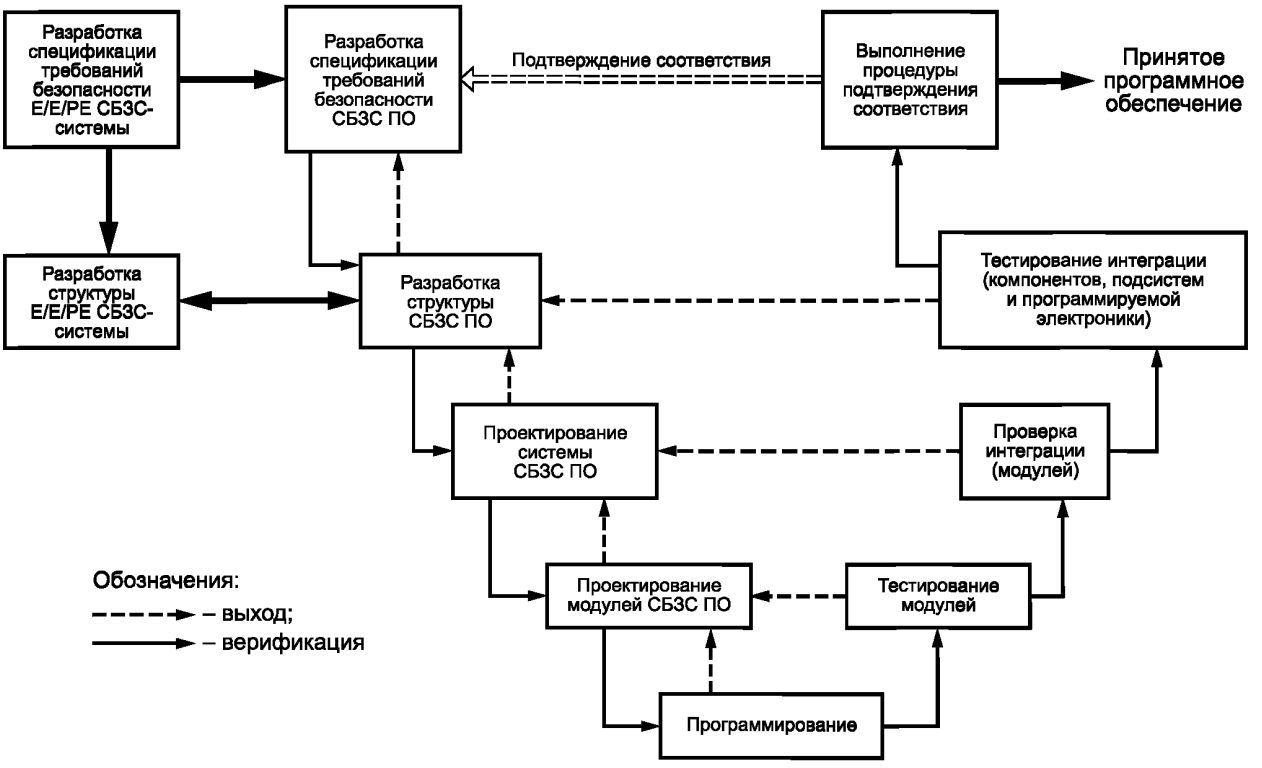
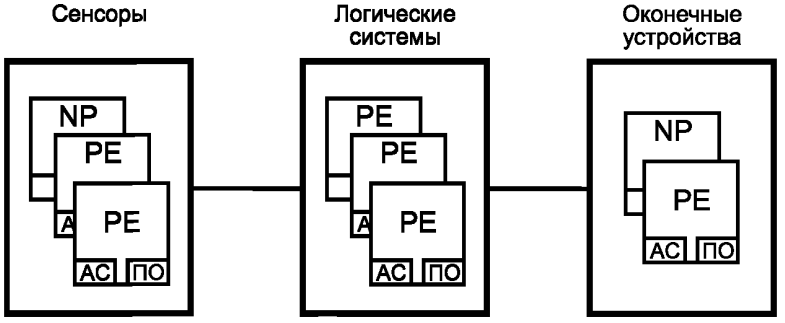


Рисунок 4 — V-модель проектирования и разработки СБЗС ПО



PE — программируемая электроника; NP — непрограммируемое устройство; АС — аппаратное средство; ПО — программное обеспечение

Структура программируемой электроники		
Структура аппаратных средств PE	Структура программного обеспечения PE (структура ПО, включая «зашитое» в ПЗУ ПО и прикладные программы)	
Базовые и прикладные АС, например: - встроенные устройства диагностического тестирования; - избыточные процессоры; - сдвоенные платы ввода/вывода	ПО, «зашитое» в ПЗУ, например: - коммуникационные драйверы; - программы обработки отказов; - исполнительные программы	Прикладные программы, например: - функции ввода/вывода; - производные функции (например, контроль сенсора, если он не обеспечен как сервис «зашитого» в ПЗУ ПО)

Рисунок 5 — Взаимосвязь между структурами АС и ПО программируемой электроники

- установления требований к функциям безопасности каждой СБЗС-системы, которые необходимы для реализации этих функций безопасности;

- установления требований к полноте безопасности каждой СБЗС-системы, необходимых для достижения уровня полноты безопасности, назначенного для каждой функции безопасности, реализуемой этой СБЗС-системой.

5.6.2 Требования

5.6.2.1 Выполнение требований достигается комбинацией базового встраиваемого ПО и прикладных программных модулей, разработанных специально для конкретного приложения. Точная граница между базовым и прикладным ПО зависит от выбранной структуры программной системы (см. рисунок 5).

5.6.2.2 Если требования к безопасности ПО были уже установлены в требованиях к безопасности СБЗС-систем (см. ГОСТ Р 53195.3), повторно устанавливать их не следует.

5.6.2.3 Спецификация требований к безопасности ПО должна быть разработана на основе требований к безопасности СБЗС-систем (см. ГОСТ Р 53195.3) и требований к планированию безопасности (см. 5.4.2 настоящего стандарта).

Примечание — Спецификация требований к безопасности СБЗС-систем должна быть доступна разработчикам ПО. Должно быть обеспечено взаимодействие между разработчиками АС и ПО СБЗС-систем.

5.6.2.4 Степень подробности спецификации требований к безопасности ПО должна быть достаточной для обеспечения стадии проектирования и реализации необходимой информации для достижения требуемой полноты безопасности и проведения оценки функциональной безопасности. Уровень детализации зависит от сложности проекта и определяется проектировщиком ПО.

5.6.2.5 Разработчиком ПО должен быть проведен анализ информации, содержащейся в 5.6.2, для гарантирования того, что требования определены адекватным образом. При этом должны быть учтены:

- функции безопасности;
- конфигурация или структура системы;
- требования к полноте безопасности АС (РЕ, датчиков и устройств привода);
- требования к полноте безопасности ПО;
- производительность и время отклика системы;
- интерфейсы оборудования и оператора.

5.6.2.6 Разработчиком ПО должны быть установлены процедуры для устранения разногласий при назначении уровня полноты безопасности ПО.

5.6.2.7 В степени, необходимой для конкретного уровня полноты безопасности, требования к безопасности ПО должны быть выражены и структурированы так, чтобы они:

- были ясными, точными, недвусмысленными, пригодными для верификации, тестирования, поддержки и выполнения, а также соразмерными с уровнем полноты безопасности;

- были пригодными для определения их источника в спецификации требований к безопасности СБЗС-систем;

- не содержали информации и описаний, которые являются двусмысленными и/или могут быть не поняты другими пользователями документа на различных стадиях жизненного цикла СБЗС-систем.

5.6.2.8 В требованиях к безопасности ПО должны быть подробно описаны все соответствующие режимы работы УО, если они не были уже адекватно определены в требованиях к безопасности СБЗС-систем.

5.6.2.9 В спецификации требований к безопасности ПО должны быть установлены и документированы все относящиеся к безопасности и иные необходимые ограничения, связанные с взаимодействием между АС и ПО.

5.6.2.10 В спецификации требований к безопасности ПО в степени, требуемой в описании проекта структуры АС СБЗС-системы, должны быть учтены:

- самоконтроль программного обеспечения;
- мониторинг РЕ, аппаратуры, датчиков и устройств привода;
- периодическое тестирование функций безопасности во время выполнения программы;
- разрешение тестирования функций безопасности во время работы УО.

5.6.2.11 Если СБЗС-система должна выполнять функции, не относящиеся к безопасности, эти функции должны быть четко указаны в спецификации требований к безопасности ПО.

5.6.2.12 Спецификация требований к безопасности ПО должна содержать необходимые характеристики безопасности продукции, а не проекта. С учетом 5.6.2.2—5.6.2.11 в зависимости от конкретных обстоятельств должны быть установлены следующие положения:

а) требования к функциям безопасности ПО:

- функции, обеспечивающие достижение и поддержание безопасного состояния УО;
- функции, связанные с обнаружением, оповещением и обработкой ошибок АС РЕ;
- функции, связанные с обнаружением, оповещением и обработкой ошибок датчиков и устройств привода;
- функции, связанные с обнаружением, оповещением и обработкой ошибок в самом ПО (самоконтроль ПО);
- функции, связанные с периодическим тестированием функций в режиме реального времени;
- функции, связанные с периодическим тестированием функций в автономном режиме;
- функции, обеспечивающие безопасную модификацию СБЗС-систем;
- интерфейсы функций, не связанных с безопасностью;
- емкость системы и диапазон временных характеристик;
- интерфейсы между ПО и РЕ СБЗС-системами.

Примечание — Интерфейсы должны содержать средства автономного программирования и программирования с внешнего устройства (онлайн);

б) требования к полноте безопасности ПО:

- уровни полноты безопасности для каждой функции, приведенной в перечислении а).

5.6.3 Планирование подтверждения соответствия (см. блок 1.2 на рисунке 3)

5.6.3.1 В ходе планирования подтверждения соответствия должны быть установлены процедурные и технические шаги, используемые для демонстрации того, что СБЗС ПО удовлетворяет требованиям безопасности (см. 5.6.2).

Примечания

1 В зависимости от состава и сложности полной СБЗС-системы возможны варианты осуществления подтверждения соответствия ПО: подтверждение соответствия ПО отдельных РЕ СБЗС-систем (подсистем) до объединения их в комплексную систему безопасности; оценка соответствия ПО отдельных РЕ СБЗС-систем (подсистем) в составе комплексной системы безопасности; оценка соответствия ПО комплексной системы безопасности.

2 Выбор варианта осуществления подтверждения соответствия ПО РЕ СБЗС-систем (подсистем) или их комбинации определяется разработчиком ПО.

5.6.3.2 План подтверждения соответствия безопасности ПО должен содержать следующую информацию:

а) описание этапов подтверждения соответствия;

б) перечень лиц, осуществляющих подтверждение соответствия;

в) идентификацию соответствующих режимов работы УО, включая:

- подготовку к использованию, а также установку и настройку;
- работу в режиме запуска и обучения, в автоматическом, ручном, полуавтоматическом и стационарном режимах;
- переустановку, выключение, сопровождение;
- предполагаемые ненормальные режимы;

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

д) техническую стратегию, применяемую для подтверждения соответствия (например, имитация/моделирование, вероятностное тестирование и т. п.) (см. таблицу А.7 приложения А);

е) методы/средства и процедуры в соответствии с перечислением д), которые должны быть использованы для подтверждения каждой функции требованиям к функциям безопасности ПО и требованиям к полноте безопасности ПО (см. 5.6.4);

ж) конкретные ссылки на требования к безопасности ПО (см. 5.6.4);

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

к) критерии соответствия/несоответствия при процедуре подтверждения соответствия (см. 5.6.3.4);

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

5.6.3.3 В рамках процедуры подтверждения соответствия СБЗС ПО, если этого требует уровень полноты безопасности (см. ГОСТ Р 53195.2—2008, 7.15 и ГОСТ Р 53195.3—2015, 5.16.2), область применения и содержание планирования подтверждения соответствия безопасности ПО должны быть изучены экспертом или третьей стороной, представляющей эксперта. Эта процедура должна также включать в себя заявление о присутствии эксперта при испытаниях.

5.6.3.4 Критерии прохождения/непрохождения при завершении подтверждения соответствия ПО должны включать в себя:

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

5.6.4 Проектирование и разработка ПО (см. блок 1.3 на рисунке 3)

5.6.4.1 На стадии проектирования и разработки СБЗС ПО должны быть:

- создана структура ПО, удовлетворяющая требованиям к безопасности ПО, относящаяся к необходимому уровню полноты безопасности;
- проведен анализ и оценка требований, предъявляемых к ПО, обусловленных АС СБЗС-системы, с учетом степени взаимодействия между АС и ПО СБЗС-системы для обеспечения безопасной работы УО;
- проведен выбор инструментальных средств, включая языки программирования и компиляторы, соответствующие заданному уровню полноты безопасности на протяжении всего жизненного цикла СБЗС-системы и способствующие выполнению процессов верификации, оценке, подтверждению соответствия и модификации ПО;
- спроектировано и реализовано ПО, удовлетворяющее установленным требованиям к безопасности ПО в соответствии с необходимым уровнем полноты безопасности, пригодное для анализа, верификации и способное к безопасной модификации;
- проведена проверка выполнения требований к безопасности ПО в отношении необходимых функций безопасности и полноты безопасности ПО.

Примечание — Методы/средства, рекомендуемые для проектирования и разработки ПО, приведены в таблице А.1 приложения А.

5.6.4.2 В зависимости от характера процесса разработки ПО ответственность за выполнение требований 5.6.4 может быть возложена на поставщика ПО, на пользователя или на обе стороны. Распределение ответственности должно быть определено во время планирования безопасности (см. 5.6.3).

5.6.4.3 В соответствии с требуемым уровнем полноты безопасности выбранный метод проектирования должен обладать характеристиками, которые облегчают:

- абстракцию, разделение на модули и другие характеристики, контролирующие уровень сложности;
- выражение:
  - выполняемых функций;
  - обмена данными между компонентами;
  - информации, относящейся к последовательности и времени выполнения;
  - ограничений на время выполнения;
  - параллельного выполнения;
  - структур данных и их свойств;
  - проектных предположений и их зависимостей;
- понимание документации разработчиками и другими лицами, которые должны иметь дело с проектом;
- верификацию и оценку соответствия.

5.6.4.4 На стадии проектирования должны быть предусмотрены тестируемость и способность к безопасной модификации ПО для облегчения реализации этих возможностей в окончательной версии СБЗС-системы.

5.6.4.5 Выбранный метод проектирования должен обладать характеристиками, которые облегчают модификацию программного обеспечения. К числу таких характеристик относятся модульность, скрытие информации и инкапсуляция.

5.6.4.6 Представление проекта должно основываться на однозначно определенной или ограниченной до однозначно определенных свойств нотации.

5.6.4.7 В проекте должна быть минимизирована, насколько это возможно, та часть ПО, которая относится к безопасности.

5.6.4.8 Если ПО должно реализовать функции как относящиеся, так и не относящиеся к безопасности, оно в целом должно рассматриваться как относящееся к безопасности, если только в проекте не продемонстрирована достаточная независимость между этими функциями. Обоснование независимости должно быть документировано.

5.6.4.9 Если программное обеспечение должно реализовать функции безопасности, имеющие различный уровень полноты безопасности, то следует считать, что все ПО имеет наивысший уровень из этих уровней безопасности, если только в проекте не будет продемонстрирована достаточная независимость функций, имеющих различный уровень полноты безопасности. Обоснование независимости должно быть документировано.

5.6.4.10 Уровень полноты безопасности ПО должен быть не ниже, чем уровень полноты функции безопасности, к которой оно относится.

Примечание — Уровень полноты безопасности компонента ПО может быть ниже, чем уровень полноты функции безопасности, к которой он относится, если этот компонент используется в сочетании с другими компонентами АС, такими, что уровень полноты безопасности сочетания компонентов не меньше уровня полноты безопасности функции безопасности.

5.6.4.11 В состав проекта по мере возможности должны быть включены функции, выполняющие проверки и все диагностические тесты для обеспечения выполнения требований к полноте безопасности СБЗС-системы (как установлено в ГОСТ 53195.3).

5.6.4.12 В состав проекта по мере возможности должны быть включены средства самоконтроля потоков управления и потоков данных, адекватные требуемому уровню полноты безопасности. При обнаружении ошибки должны быть выполнены соответствующие действия (см. таблицы А.2 и А.4 приложения А).

5.6.4.13 Если стандартное или ранее разработанное программное обеспечение должно использоваться как часть проекта (см. таблицы А.2 и А.4 приложения А), оно должно быть четко идентифицировано. Способность программного обеспечения удовлетворять требованиям, изложенным в спецификации требований к безопасности СБЗС ПО (см. 5.6), должна быть обоснована. Обоснование этой способности должно быть подкреплено данными по удовлетворительной работе ПО в схожем приложении или быть предметом тех же самых процедур верификации и подтверждения соответствия, которые подразумеваются для любого вновь разрабатываемого ПО. При этом должны быть оценены ограничения, связанные с условиями, в которых работало ПО (например, зависимость от операционной системы и компилятора).

Примечание — Такое обоснование может быть разработано при планировании безопасности (см. 5.4).

5.6.4.14 Подраздел 5.6.4 должен по мере возможности применяться к данным, включая любой язык генерации данных.

5.6.5 Требования к структуре ПО

5.6.5.1 Структура ПО должна определять основные компоненты и подсистемы ПО, их взаимосвязь, способ реализации необходимых характеристик, в том числе полноты безопасности.

Примечания

1 Примерами компонентов ПО являются операционные системы, базы данных, коммуникационные подсистемы, прикладные программы, инструментальные средства программирования и диагностики и т. п.

2 С точки зрения безопасности стадия разработки структуры программного обеспечения соответствует периоду разработки базовой стратегии безопасности для ПО.

5.6.5.2 В зависимости от характера разработки ПО ответственность за выполнение требований 5.6.5 может быть возложена только на поставщика, только на разработчика или на обе стороны. Распределение ответственности определяется разработчиком и должно быть документировано во время планирования безопасности (см. 5.4).

Примечания

1 Для пользовательского прикладного программирования в языках с ограниченной изменчивостью, в частности в языках, используемых в ПЛК, структура определяется поставщиком как стандартная характеристика ПЛК. Однако в рамках настоящего стандарта к поставщику может быть предъявлено требование гарантировать пользователю соответствие поставляемого продукта требованиям 5.6.4. В этом случае пользователь приспосабливает ПЛК, используя стандартные возможности программирования, например многозвенные логические схемы. При

этом требования 5.6.5—5.6.11 сохраняют свою силу. Требование определения и документирования структуры может рассматриваться как информация, которую пользователь может использовать при выборе ПЛК (или эквивалентного ему устройства), для приложения.

2 В другом крайнем случае в некоторых встроенных приложениях, использующих язык с полной изменчивостью, например в устройствах биометрической идентификации системы контроля и управления доступом, управляемых микропроцессором, структура должна создаваться поставщиком специально для приложения (или класса приложений). Пользователь обычно не имеет инструментария для программирования. В этих условиях ответственность за обеспечение соответствия по 5.4 ложится на поставщика.

3 Имеются системы, попадающие в промежуток между типами, упомянутыми в примечаниях 1 и 2, в таких случаях ответственность за соответствие разделяется между пользователем и поставщиком.

5.6.5.3 Разрабатываемый проект структуры ПО должен быть создан поставщиком и/или разработчиком ПО. Описание структуры должно быть по дробным. Описание должно:

- содержать выбор и обоснование интегрированного набора методов/средств, которые будут необходимы в течение жизненного цикла модулей безопасности для удовлетворения требований к безопасности ПО на заданном уровне полноты безопасности (см. 5.5). Эти методы/средства включают в свой состав стратегию проектирования ПО для обеспечения устойчивости к отказам (совместимую с аппаратными средствами) и для избегания отказов, в том числе (при необходимости) могут включать в свой состав избыточность и разнообразие;
- основываться на разделении на компоненты/подсистемы, для каждой из которых должна быть предоставлена следующая информация:
  - являются ли они вновь разработанными, уже существующими или находящимися в частной собственности;
  - проводилась ли верификация, и если проводилась, то при каких условиях;
  - связан ли каждый из этих компонентов/подсистем с безопасностью;
  - каков уровень полноты безопасности для компонента/подсистемы;
- определять все взаимодействия между ПО и АС СБЗС-системы, а также оценивать и детализировать их значения;
- использовать для представления структуры нотацию, однозначно определенную или ограниченную до подмножества однозначно определенных характеристик;
- содержать набор проектных характеристик, которые должны использоваться для поддержания полноты безопасности всех данных. В число таких данных допускается включать входные и выходные данные процесса, коммуникационные данные, данные интерфейса оператора, данные сопровождения и данные, хранящиеся во внутренних базах данных;
- определять тесты интеграции структуры ПО для обеспечения выполнения требований, изложенных в спецификации требований к безопасности ПО, на заданном уровне полноты безопасности (см. 5.5).

Примечание — Методы/средства, рекомендуемые для применения при проектировании структуры ПО, приведены в таблице А.2 приложения А и таблице Б.7 приложения Б.

5.6.5.4 Любые изменения, которые может потребоваться внести в специфицированные требования к СБЗС-системе после использования мероприятий 5.6.5, должны быть согласованы с разработчиком СБЗС-систем и документированы.

Примечание — Итерационное взаимодействие между структурой АС и ПО является неизбежным (см. рисунок 4). Взаимодействие разработчика ПО с разработчиком АС при рассмотрении спецификации тестирования интеграции РЕ и ПО является обязательным условием процесса разработки СБЗС ПО (см. 5.6.11).

5.6.5.5 Структура ПО комплексных СБЗС-систем должна иметь многоуровневую (двух- или трехуровневую) распределенную архитектуру.

5.6.5.6 ПО автоматизированного рабочего места (далее — АРМ) не должно выполнять функции хранения базы данных системы и непосредственного взаимодействия с УО оборудования СБЗС-подсистем.

5.6.5.7 В структуре ПО комплексных СБЗС-систем должны быть предусмотрены отдельные программные модули, реализующие непосредственное взаимодействие с УО оборудования СБЗС-подсистем.

5.6.5.8 Протоколы взаимодействия программными модулями и остальной частью СБЗС ПО должны быть, по возможности, открытыми, документированными и должны поставляться вместе с ПО для

обеспечения возможности дальнейшего наращивания перечня УО в ходе модификации без изменения общей структуры СБЗС ПО.

5.6.5.9 В структуре СБЗС ПО должны быть предусмотрены возможность переключения на избыточный (резервный) компонент СБЗС-системы (сервер, технологический компьютер, линию связи, источник электропитания и т. п.) в случае отказа основного компонента, а также возможность автоматического переключения на прежний компонент при восстановлении его работоспособности.

5.6.5.10 В структуре ПО комплексной СБЗС-системы должна быть предусмотрена возможность поддержки работы в многосерверной среде, в том числе с поддержкой «горячего» резервирования серверов и синхронизации информационной базы данных между серверами, а также автоматической синхронизации времени во всех компонентах.

5.6.5.11 В структуре ПО комплексной СБЗС-системы должна быть предусмотрена возможность автоматической посылки управляющего воздействия и выполнения управляющих воздействий при возникновении любого события в любой подсистеме СБЗС-системы.

5.6.5.12 В структуре СБЗС ПО должна быть предусмотрена возможность поддержания удаленного контроля и тестирования АС и ПО СБЗС-систем и их составляющих.

5.6.6 Требования к инструментальным средствам поддержки и языкам программирования

5.6.6.1 Выбор инструментов для разработки ПО определяется разработчиком в зависимости от характера процессов разработки ПО и его структуры (см. 5.6.5).

5.6.6.2 В зависимости от характера ПО ответственность за выполнение требований 5.6.6 может быть возложена только на поставщика, только на пользователя (разработчика СБЗС ПО) или на обе стороны. Разделение ответственности должно быть документировано во время планирования безопасности (см. 5.6.3).

Примечания

1 Если пользовательское прикладное ПО выполняется на языке с ограниченной варьируемостью при низких уровнях полноты безопасности, то круг необходимых инструментов и языков программирования может быть ограничен стандартными языками ПЛК, редакторами и загрузчиками. В этом случае ответственность за выполнение требований 5.6.6 возлагается, главным образом, на поставщика.

2 При более высоких уровнях полноты безопасности могут потребоваться ограниченные подмножества языка ПЛК, а также средства верификации и отладки, такие как анализаторы кода и имитаторы. В этих условиях ответственность возлагается на поставщика и на пользователя.

3 Инструментарий для встраиваемых приложений, использующий языки с полной варьируемостью, должен быть более разнообразным даже в случае низких уровней полноты безопасности. Ответственность за выполнение требований 5.6.6 возлагается, главным образом, на разработчиков ПО. Ответственность несет также поставщик ПЛК, который может использовать языки с полной варьируемостью в языке с низким уровнем варьируемости для обеспечения пользовательского прикладного программирования.

5.6.6.3 Набор интегрированных инструментальных средств должен выбираться в соответствии с требуемым уровнем полноты безопасности и должен включать в свой состав языки программирования, компиляторы, средства управления конфигурацией и, при необходимости, автоматизированные средства тестирования. При выборе инструментальных средств (за исключением тех средств, которые использовались при первоначальной разработке системы) должна быть учтена их способность выполнять необходимые задачи на протяжении всего жизненного цикла СБЗС-систем.

5.6.6.4 В степени, необходимой для требуемого уровня полноты безопасности, выбранные языки программирования должны:

- иметь транслятор/компилятор, сертифицированный на соответствие национальному или международному стандарту либо оцененный на пригодность его для применения;
- быть полностью и однозначно определенными либо ограниченными до подмножества однозначно определяемых элементов;
- соответствовать характеристикам приложения;
- обладать свойствами, облегчающими обнаружение ошибок программирования;
- поддерживать характеристики, соответствующие методу проектирования.

Примечание — Инструментальные средства поддержки, а также языки программирования, рекомендованные к применению, приведены в таблице А.3 приложения А и таблице Б.3 приложения Б.

5.6.6.5 Если требования 5.6.6.4 не могут быть выполнены, то при описании проекта структуры ПО (см. 5.6.5) следует документировать обоснование использования альтернативного языка программирования. В обосновании должны быть подробно рассмотрены пригодность языка программирования, а также применяемые дополнительные меры, обусловленные известными недостатками языка.

5.6.6.6 Стандарты программирования (составления программ), используемые при разработке всего СБЗС ПО, должны быть рассмотрены экспертом на предмет определения их пригодности.

5.6.6.7 Стандарты программирования должны определять правильные методы программирования, запрещать использование небезопасных возможностей языка программирования (например, неопределенных особенностей языка, неструктурированных конструкций и т. п.) и определять процедуры для документирования исходного текста.

Документация, относящаяся к исходному тексту, должна содержать, по меньшей мере, следующую информацию:

- наименование (имя) юридического лица (например, компании, авторов и т. п.);
- описание;
- входные и выходные данные;
- историю управления конфигурацией ПО.

5.6.7 Требования к детальному проектированию и разработке

5.6.7.1 Под детальным проектированием здесь понимается разделение основных компонентов структуры на систему программных модулей, проектирование отдельных программных модулей и их программирование. В небольших приложениях проектирование программных систем и структуры могут быть объединены.

5.6.7.2 Допускается изменение характера детального проектирования и разработки в зависимости от характера процессов разработки программ и структуры ПО (см. 5.6.2).

Примечания

1 При осуществлении прикладного программирования пользователем, применяющим языки с ограниченной варьируемостью, например языки многозвенных логических схем и языки функциональных блоков, детальное проектирование может рассматриваться скорее как конфигурирование, чем как программирование.

2 В ходе детального проектирования целесообразно применять приемы «установившейся практики», например:

- структурирование программного обеспечения, включая организацию модульной структуры, которая выделяет (настолько, насколько это возможно) блоки, связанные с безопасностью;
- использование проверок на попадание в интервал допустимых значений и других возможностей защиты от ошибок при вводе исходных данных;
- использование ранее верифицированных программных модулей;
- применение конструкций, которые облегчают выполнение будущих модификаций ПО.

3 Методы/средства, рекомендуемые при детальном проектировании ПО, приведены в таблице А.4 приложения А и таблицах Б.1, Б.7 и Б.9 приложения Б.

5.6.7.3 В зависимости от характера ПО ответственность за выполнение требований 5.6.7 может быть возложена только на поставщика, только на пользователя или на обе стороны. Разделение ответственности должно быть документировано во время планирования безопасности (см. 5.5.2).

5.6.7.4 Для начала детального проектирования должна быть использована следующая информация:

- спецификация требований к безопасности ПО (см. 5.6.2);
- описание проекта структуры (см. 5.6.5.3);
- план подтверждения соответствия ПО требованиям безопасности (см. 5.7.3).

5.6.7.5 Разработка ПО должна осуществляться таким образом, чтобы обеспечивались модульность, тестируемость и способность к безопасной модификации ПО.

5.6.7.6 Дальнейшее уточнение проекта для каждого главного компонента/подсистемы в описании проекта структуры ПО (см. 5.6.5.3) должно быть основано на разделении ПО на программные модули (т. е. на спецификации проекта программной системы). Должны быть разработаны каждый программный модуль и тесты, которые необходимо использовать для проверки этих модулей.

Примечание — Для стандартных или ранее разработанных компонентов программных модулей не требуются проект или спецификации тестирования, если может быть показано, что они удовлетворяют требованиям 5.6.4.11.

5.6.7.7 Должны быть установлены соответствующие тесты интеграции программных систем для удостоверения того, что программные системы удовлетворяют требованиям к безопасности ПО для установленного уровня полноты безопасности (см. 5.4).

5.6.8 Требования к реализации исходных текстов программ

5.6.8.1 Исходные тексты программ должны:

- быть читаемыми, понятными и пригодными к тестированию;
- удовлетворять установленным требованиям к программным модулям (см. 5.6.7);

- удовлетворять установленным требованиям к стандартам программирования;
- удовлетворять всем требованиям, установленным при планировании безопасности (см. 5.4.2).

5.6.8.2 Каждый модуль должен быть просмотрен.

Примечания

1 Просмотр модуля относится к процессам верификации.

2 Методы/средства, рекомендуемые при подготовке текстов программ, приведены в таблице А.4 приложения А и в таблицах Б.1, Б.7 и Б.9 приложения Б.

5.6.9 Требования к тестированию программных модулей

5.6.9.1 Тестирование программных модулей следует проводить в сочетании с просмотром исходных текстов для удостоверения в том, что программный модуль корректно выполняет все требования, содержащиеся в спецификации тестирования и в спецификации модуля, что эквивалентно верификации модуля.

5.6.9.2 Каждый программный модуль должен быть протестирован в соответствии со спецификацией, разработанной при детальном проектировании ПО (см. 5.6.7).

5.6.9.3 Тестирование должно продемонстрировать, что каждый программный модуль выполняет функции, для которых он предназначен, и не выполняет не предусмотренные для него функции.

Примечания

1 Тестирование программных модулей не означает тестирование всех комбинаций входных данных и всех комбинаций выходных данных. Достаточным может быть тестирование всех классов эквивалентности или структурное тестирование. Анализ граничных значений, анализ управляющей логики или анализ скрытых путей выполнения программы может уменьшить количество проверок до приемлемого уровня. Программы, пригодные для анализа, могут позволить достичь более быстрого выполнения требований.

2 Если при разработке используются формальные методы, формальные доказательства или операторы проверки условий, область применения подобных проверок может быть уменьшена.

3 Допускается использовать также статистические данные.

4 Методы/средства, рекомендуемые для тестирования программных модулей, приведены в таблице А.5 приложения А и в таблицах Б.2, Б.3 и Б.6 приложения Б.

5.6.9.4 Результаты тестирования программных модулей следует документировать.

5.6.9.5 Должны быть установлены процедуры для коррекции при непрохождении теста.

5.6.10 Требования к тестированию интеграции ПО

5.6.10.1 Тесты интеграции ПО должны быть разработаны параллельно с разработкой ПО на стадии проектирования и разработки СБЗС ПО.

5.6.10.2 Спецификации тестов интеграции ПО должны устанавливать:

- разделение ПО на контролируемые интегрируемые подмножества;
- контрольные примеры и контрольные данные;
- типы проверок, которые должны быть выполнены;
- условия тестирования, используемые инструменты, конфигурацию и программы;
- критерии, на основании которых выносится решение о прохождении теста;
- процедуры, которые необходимо выполнить, если тестирование дало отрицательный результат.

5.6.10.3 ПО должно быть протестировано в соответствии с заранее определенными тестами интеграции программ. Тестирование должно продемонстрировать, что все программные модули и программные компоненты/подсистемы корректно взаимодействуют для выполнения функций, для которых они предназначены, и не выполняют непредусмотренных функций.

Примечания

1 Тестирование программных модулей не означает тестирование всех комбинаций входных данных и всех комбинаций выходных данных. Достаточным может быть тестирование всех классов эквивалентности или структурное тестирование. Анализ граничных значений, анализ управляющей логики или анализ скрытых путей выполнения программы может уменьшить количество проверок до приемлемого уровня. Программы, пригодные для анализа, могут позволить достичь более быстрого выполнения требований.

2 Если при разработке используются формальные методы, формальные доказательства или операторы проверки условий, область применения подобных проверок может быть уменьшена.

3 Допускается использовать также статистические данные.

4 Методы/средства, рекомендуемые для тестирования при интеграции, приведены в таблице А.5 приложения А и таблицах Б.2, Б.3 и Б.6 приложения Б.

5.6.10.4 Результаты тестирования интеграции ПО должны быть документированы. В документации должны быть сформулированы результаты тестирования и должно быть указано, были ли выпол-

нены цели и критерии проверки. При неудачных результатах тестирования должны быть описаны причины этого.

5.6.10.5 При интеграции ПО все модификации или изменения должны быть объектом анализа влияния, который позволяет определить, какие программные модули затрагиваются изменениями, и установить необходимость повторной верификации и проектирования.

5.6.11 Интеграция программируемой электроники (АС и ПО) (см. блок 1.4 на рисунке 3)

5.6.11.1 При интеграции программируемой электроники осуществляются интеграция ПО в предусмотренные АС, объединение АС и ПО в РЕ СБЗС-систему (подсистему), а также проверка для удостоверения их совместимости и того, что выполняются требования достижения необходимого уровня полноты безопасности.

5.6.11.2 Тесты интеграции должны быть определены на стадии проектирования и разработки в целях проверки совместимости ПО и АС в связанной с безопасностью РЕ СБЗС-системе (подсистеме).

5.6.11.3 Спецификация тестирования интеграции программируемой электроники должна устанавливать:

- разделение ПО на контролируемые интегрируемые подмножества;
- контрольные примеры и контрольные данные;
- типы проверок, которые должны быть выполнены;
- условия тестирования, используемые инструменты, конфигурацию и программы;
- критерии, на основании которых выносится решение о прохождении теста;
- процедуры, которые необходимо выполнить, если тестирование дало отрицательный результат.

5.6.11.4 Специфицированные тесты интеграции программируемой электроники (АС и ПО) должны быть различными для операций, которые выполняются разработчиком на его оборудовании, и операций, требующих доступа к оборудованию пользователя.

5.6.11.5 Тесты интеграции программируемой электроники (АС и ПО) должны быть различными для следующих процессов:

а) загрузки ПО в целевое программируемое электронное оборудование;

б) интеграции Е/Е/РЕ-устройств, т. е. добавления интерфейсов, таких как датчики и устройства привода;

в) полной интеграции УО и Е/Е/РЕ СБЗС-системы (подсистемы).

Примечание — Перечисления б) и в) охватываются ГОСТ Р 53195.1 — ГОСТ Р 53195.3.

5.6.11.6 ПО должно быть интегрировано с АС программируемой электроники в соответствии со специфицированными тестами интеграции для РЕ (АС и ПО).

5.6.11.7 При тестировании интеграции СБЗС программируемой электроники (АС и ПО) все модификации или изменения должны быть объектом анализа влияния, в результате которого необходимо определить, какие программные модули затрагиваются изменениями, и установить необходимость повторной верификации.

Примечание — Рекомендуемые методы/средства, применяемые при осуществлении интеграции, приведены в таблице А.6 приложения А и таблицах Б.3, Б.6 приложения Б.

5.6.11.8 Тестовые примеры и результаты их выполнения должны быть документированы для последующего анализа.

5.6.11.9 Результаты тестирования интеграции программируемой электроники (АС и ПО) должны быть документированы. В документации должны быть сформулированы результаты тестирования, а также указано, были ли выполнены цели и критерии проверки. Если тестирование окончилось неудачно, должны быть описаны причины этого.

5.6.12 Интеграция СБЗС-систем (подсистем) в комплексную систему безопасности (см. блок 1.7 на рисунке 3)

5.6.12.1 При интеграции отдельных Е/Е/РЕ СБЗС-систем (подсистем) в комплексную систему безопасности осуществляют интеграцию ПО в предусмотренные АС, объединение СБЗС-систем (подсистем) в комплексную систему безопасности, тестирование интеграции для удостоверения в их совместимости в том, что для каждой из СБЗС-систем (подсистем) выполняются необходимые требования к уровню полноты безопасности, а также выполняются требования к функциям безопасности комплексной системы безопасности.

Примечание — В зависимости от состава, сложности СБЗС-систем и сложности проекта проектировщиком ПО могут быть выбраны различные варианты интеграции СБЗС-систем (подсистем) в комплексную систему безопасности и тестирования интеграции:

- автономная интеграция ПО и АС каждой из СБЗС-систем (подсистем), тестирование интеграции до объединения СБЗС-систем (подсистем) в комплексную систему безопасности и последующие интеграция и тестирование интеграции при объединении СБЗС-систем (подсистем) в комплексную систему безопасности;
- интеграция ПО и АС СБЗС-систем (подсистем) в составе комплексной системы безопасности после объединения АС СБЗС-систем (подсистем) в комплексную систему безопасности;
- комбинация перечисленных выше двух вариантов.

5.6.12.2 Варианты интеграции и тесты интеграции комплексной системы безопасности должны быть определены на стадии проектирования и разработки СБЗС ПО.

5.6.12.3 Спецификация тестирования интеграции СБЗС-систем (подситем) в комплексную систему безопасности должна устанавливать:

- разделение ПО на контролируемые интегрируемые подмножества;
- контрольные примеры и контрольные данные;
- типы проверок, которые должны быть выполнены;
- условия тестирования, используемые инструменты, конфигурацию и программы;
- критерии, на основании которых выносится решение о прохождении теста;
- процедуры, которые необходимо выполнить, если тестирование дало отрицательный результат.

5.6.12.4 Тесты интеграции СБЗС-систем (подсистем) должны быть различными для операций, которые выполняются разработчиком на его оборудовании, и операций, требующих доступа к оборудованию пользователя.

5.6.12.5 Тесты интеграции СБЗС-систем (подсистем) должны быть различными для следующих процессов:

а) загрузки ПО в предназначенное программируемое электронное оборудование;

б) интеграции ПО и АС и расширения СБЗС-систем (подсистем) путем добавления новых средств контроля и управления;

в) интеграции УО и E/E/PE СБЗС-систем (подсистем);

г) интеграции СБЗС-систем (подсистем) в комплексную систему безопасности.

Примечание — Перечисления б) и г) охватываются ГОСТ Р 53195.1—ГОСТ Р 53195.3.

5.6.12.6 ПО должно быть интегрировано с программируемой электроникой АС СБЗС-систем (подсистем) в соответствии с определенными на стадии проектирования вариантами интеграции и тестами интеграции СБЗС-систем (подсистем).

5.6.12.7 При тестировании интеграции СБЗС-систем (подсистем) все модификации или изменения должны быть объектом анализа влияния, в результате которого следует определить, какие составляющие ПО затрагиваются изменениями, и установить необходимость их повторной верификации.

5.6.12.8 Тестовые примеры и результаты их выполнения должны быть документированы для последующего анализа.

5.6.12.9 Результаты тестирования интеграции СБЗС-систем (подсистем) должны быть документированы. В документации должны быть сформулированы результаты тестирования, а также указано, были ли выполнены цели и критерии проверки. Если тестирование окончилось неудачно, должны быть описаны причины этого.

5.7 Ввод в действие, эксплуатация и модификация ПО

5.7.1 Проектировщиком ПО должны быть разработаны процедуры установки, ввода в действие, эксплуатации и модификации СБЗС ПО (см. блок 1.6 на рисунке 3), обеспечивающие сохранение установленной проектом функциональной безопасности СБЗС-систем при работе и модификациях ПО в период эксплуатации.

5.7.2 СБЗС-система может быть введена в эксплуатацию только после подтверждения соответствия АС и ПО требованиям безопасности.

5.7.3 Подтверждение соответствия ПО требованиям безопасности (см. блоки 1.5 и 1.8 на рисунке 3)

5.7.3.1 Если соответствие требованиям к безопасности ПО уже было установлено для СБЗС-системы (ГОСТ Р 53195.2—2008, 7.8 и 7.15), проведение повторного подтверждения соответствия не требуется.

5.7.3.2 Операции подтверждения соответствия должны выполняться в соответствии со спецификациями, разработанными при планировании подтверждения соответствия безопасности ПО (см. 5.6.3).

Примечание — Методы/средства, рекомендуемые для применения при оценке и подтверждении соответствия, приведены в таблице А.7 приложения А и в таблицах Б.3, Б.5 приложения Б.

5.7.3.3 Результаты подтверждения соответствия безопасности программного обеспечения должны быть документированы.

5.7.3.4 При проведении подтверждения соответствия безопасности ПО для каждой функции безопасности должны быть документированы следующие данные:

- хронологический перечень операций подтверждения соответствия;
- используемая версия плана подтверждения соответствия безопасности ПО (см. 5.6.3);
- подтверждаемые функции безопасности (с использованием тестирования или анализа) со ссылками на план подтверждения соответствия безопасности ПО (см. 5.6.3);
- использованные инструменты и оборудование, а также данные калибровки;
- критерии соответствия/несоответствия при процедуре подтверждения соответствия;
- результаты операций подтверждения соответствия;
- расхождения между ожидаемыми и фактическими результатами.

5.7.3.5 При наличии расхождений между ожидаемыми и фактическими результатами проводят анализ и принимают решение о продолжении проверки или о подготовке запроса на внесение изменений и возврате к более ранней стадии жизненного цикла разработки. Это решение должно быть документировано как часть результатов подтверждения соответствия безопасности ПО.

5.7.3.6 Подтверждение соответствия СБЗС ПО должно удовлетворять следующим требованиям:

- основным методом подтверждения соответствия для ПО должно быть тестирование; анимацию и моделирование допускается использовать как дополнительные методы;
- прогон программного обеспечения следует выполнять путем имитации:
  - входных сигналов в нормальном режиме работы;
  - предполагаемых случаев (моделей событий);
  - нежелательных условий, требующих вмешательства системы;
- поставщиком и/или разработчиком должны быть предоставлены документированные результаты подтверждения соответствия безопасности ПО и вся имеющая отношение к этой операции документация в распоряжение разработчика системы для предоставления ему возможности выполнить требования ГОСТ Р 53195.1 — ГОСТ Р 53195.3.

5.7.3.7 К качеству программных инструментов предъявляют следующие требования:

- все инструменты, используемые при подтверждении соответствия, должны быть квалифицированы в соответствии со спецификациями, разработанными на основе национального стандарта (если таковой имеется) или международного стандарта (если таковой имеется) либо в соответствии с общепринятыми процедурами;
- оборудование, используемое при подтверждении соответствия, должно быть определенным образом квалифицировано, должна быть продемонстрирована его пригодность для подтверждения соответствия всего используемого инструментария АС и ПО.

Примечание — В настоящем стандарте квалифицирование представляет собой операцию, которая демонстрирует выполнение конкретной спецификации в отличие от базовых процедур проверки соответствия, которые могут использоваться по отношению к любой спецификации.

5.7.3.8 К результатам подтверждения соответствия ПО предъявляются следующие требования:

- проверки должны показать, что все требования, предъявляемые к безопасности ПО, выполняются правильно и что ПО не выполняет непредусмотренных функций;
- тестовые примеры и результаты тестирования должны быть документированы для последующего анализа и независимой экспертизы в соответствии с требованиями уровня полноты безопасности (см. ГОСТ Р 53195.2—2008, 7.6.10);
- документированные результаты подтверждения соответствия безопасности ПО должны содержать утверждение о том, что программа прошла подтверждение соответствия, либо описание причин, по которым она не прошла его.

5.7.4 Модификация ПО

5.7.4.1 Внесение корректировок, улучшений или изменений в ПО, подтвержденное на соответствие требованиям безопасности, не должно приводить к снижению уровня полноты безопасности ПО.

5.7.4.2 Перед выполнением какой-либо модификации ПО должны быть подготовлены процедуры модификации (см. ГОСТ Р 53195.2—2008, 7.17).

Примечания

1 Требования 5.7.4 относятся, в первую очередь, к изменениям, выполняемым на этапе работы программного обеспечения. Они могут также применяться во время интеграции программируемой электроники, а также во время общей установки и ввода в эксплуатацию (см. ГОСТ Р 53195.2—2008, 7.17).

2 Пример модели процедуры модификации приведен в ГОСТ Р 53195.2—2008, рисунок 5.

3 Методы/средства, рекомендуемые к применению в случае модификации ПО, приведены в таблице А.8 приложения А.

5.7.4.3 Процесс модификации может быть начат только после появления запроса на санкционированную модификацию ПО в рамках процедур, определенных на этапе планирования безопасности (см. ГОСТ Р 53195.2—2008, 6.2, 7.17.2, 7.17.3), в котором приведена подробная информация:

- об опасностях, к которым могут привести изменения;
- о предлагаемых изменениях;
- о причинах изменений.

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

- тем, что функциональная безопасность оказалась ниже той, которая была установлена в спецификациях;
- систематическими отказами;
- появлением нового или изменением действующего законодательства, относящегося к безопасности;
- модификацией УО или способа его использования;
- модификацией общих требований к безопасности;
- результатом анализа характеристик работы и обслуживания СБЗС-систем, который показывает, что эти характеристики имеют значения ниже запланированных;
- результатом текущего аудита функциональной безопасности.

5.7.4.4 Должен быть проведен анализ влияния предлагаемых модификаций ПО на функциональную безопасность СБЗС-систем с целью определения:

- необходим ли анализ риска;
- какие из стадий жизненного цикла ПО и программных модулей следует повторить.

5.7.4.5 Результаты анализа влияния, полученные в 5.7.4.4, должны быть документированы.

5.7.4.6 Все модификации, оказывающие влияние на функциональную безопасность СБЗС-систем, должны приводить к возврату на соответствующую стадию жизненного цикла ПО и программных модулей. Все последующие стадии должны выполняться в соответствии с процедурами, определенными для отдельных стадий согласно требованиям настоящего стандарта. При планировании безопасности (см. ГОСТ Р 53195.2—2008, 6.2) должны быть подробно описаны все последующие процессы.

Примечание — Может потребоваться выполнение полного анализа опасностей и рисков, в результате которого может появиться необходимость в установлении иных уровней полноты безопасности, отличных от ранее определенных уровней для СБЗС-систем и внешних средств уменьшения риска.

5.7.4.7 Планирование безопасности для модификации СБЗС ПО должно включать в себя следующую информацию:

- идентификацию персонала и определение требований к его квалификации;
- подробную спецификацию модификации;
- план верификации;
- область применения операций повторного подтверждения соответствия и тестирования модификации в степени, необходимой для конкретного уровня полноты безопасности.

5.7.4.8 Модификация должна быть выполнена в соответствии с разработанным планом проведения модификации.

5.7.4.9 Все модификации должны быть подробно документированы, включая:

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

5.7.4.10 Информация в хронологическом порядке о деталях всех выполненных модификаций должна быть документирована. Документация должна включать в себя данные и результаты повторной верификации и повторного подтверждения соответствия.

Примечание — Требования 5.7.4 относятся, в первую очередь, к изменениям, выполняемым на этапе работы ПО. Они могут также применяться во время интеграции программируемой электроники, а также во время общей установки и ввода в эксплуатацию (ГОСТ Р 53195.2—2008, 7.17).

5.7.4.11 Оценка необходимых модификаций или корректировок должна зависеть от результатов анализа влияния модификаций и уровня полноты безопасности программного обеспечения.

5.8 Верификация ПО

5.8.1 Верификация в рамках настоящего стандарта представляет собой процедуру, выполняемую путем анализа и/или тестирования для каждой стадии жизненного цикла ПО в целях установления, удовлетворяют ли выходные данные для используемых входных данных во всех отношениях набору целей и требований для соответствующей стадии жизненного цикла ПО.

5.8.2 Требования к верификации

5.8.2.1 Верификация ПО для каждой стадии жизненного цикла программных модулей должна планироваться одновременно с планированием их разработки; вся информация, относящаяся к верификации, должна документироваться.

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

- оценка требований полноты безопасности;
- выбор и документирование процедур, 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 сделаны.

5.8.2.3 Верификация программного обеспечения должна быть выполнена в соответствии с планом.

Примечания

1 Выбор методов и средств, предназначенных для верификации, а также степень независимости процессов верификации устанавливаются проектировщиком ПО с учетом факторов, к которым относятся:

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

2 Методы/средства, рекомендуемые для применения при верификации, приведены в таблице А.9 приложения А и таблице Б.8 приложения Б.

5.8.2.4 Должны быть документированы свидетельства того, что верифицируемая стадия завершена удовлетворительно во всех отношениях.

5.8.2.5 Документация, составляемая после каждой верификации, должна содержать:

- перечень пунктов, подлежащих верификации;
- идентификацию информации, по отношению к которой выполняется верификация;
- перечень несоответствий.

Примечание — Примерами несоответствий служат программные модули, структуры данных и алгоритмы, которые плохо адаптированы к задаче.

5.8.2.6 Вся существенная информация, относящаяся к стадии *N* жизненного цикла СБЗС ПО, необходимая для правильного выполнения следующей стадии *N* + 1, должна быть доступна разработчику ПО и верифицирована. К выходной информации стадии *N* относятся:

- информация об адекватности спецификации описания проекта либо исходного текста программ, разработанных в ходе стадии N в части:
  - функциональности;
  - полноты безопасности, характеристик и других требований планирования безопасности;
  - требований понятности для коллектива разработчиков;
  - безопасной модификации, допускающей дальнейшее совершенствование ПО;
- информация об адекватности планирования подтверждения соответствия и тестов, определенных для стадии *N*, определению и описанию проекта стадии *N*;

- результаты несоответствия:
  - между тестами, определенными для стадии N, и тестами, определенными для предыдущей стадии $N - 1$;
  - между выходными данными стадии N.

5.8.2.7 С учетом 5.8.2 должны быть выполнены следующие операции верификации:

- верификация требований к безопасности программного обеспечения;
- верификация структуры программного обеспечения;
- верификация проекта системы программного обеспечения;
- верификация проектов программных модулей;
- верификация исходных текстов программ;
- верификация данных;
- тестирование программных модулей;
- тестирование интеграции программного обеспечения;
- тестирование интеграции программируемой электроники;
- тестирование требований к безопасности программного обеспечения (подтверждение соответствия программного обеспечения).

5.8.2.8 Верификация требований к безопасности программного обеспечения

После определения требования к безопасности ПО и перед началом следующей стадии проектирования и разработки ПО верификация должна обеспечивать проверку:

- соответствуют ли требования к безопасности ПО требованиям к безопасности СБЗС-систем (см. ГОСТ Р 53195.3) в отношении функциональности, безопасности, полноты безопасности, характеристик и других требований к планированию безопасности;
- соответствует ли планирование подтверждения соответствия программ для обеспечения безопасности требованиям к безопасности ПО;
- наличия несоответствия:
  - между специфицированными требованиями к безопасности ПО и специфицированными требованиями к безопасности СБЗС-систем (см. ГОСТ Р 53195.3);
  - между специфицированными требованиями к безопасности ПО и планированием подтверждения соответствия безопасности программного обеспечения.

5.8.2.9 Верификация структуры ПО

После проектирования структуры ПО верификация должна обеспечить проверку:

- удовлетворяет ли описание проекта структуры ПО специфицированным требованиям к безопасности ПО;
- адекватны ли специфицированные тесты интеграции структуры ПО описанию проекта структуры ПО;
- адекватны ли атрибуты каждого основного компонента/подсистемы в отношении:
  - реализуемости требуемых характеристик безопасности;
  - возможности проверки при последующей верификации;
  - пониманию структуры ПО персоналом, выполняющим разработку и верификацию;
  - безопасной модификации, позволяющей выполнять дальнейшее совершенствование программы;
- наличия несовместимости:
  - между описанием проекта структуры ПО и специфицированными требованиями к безопасности ПО;
  - между описанием проекта структуры ПО и специфицированными тестами интеграции структуры ПО;
  - между специфицированными тестами интеграции структуры ПО и планированием подтверждения соответствия безопасности ПО;

5.8.2.10 Верификация проекта системы ПО

После завершения спецификации системы ПО верификация должна проверить:

- удовлетворяет ли специфицированный проект системы ПО проекту структуры ПО;
- удовлетворяют ли специфицированные тесты интеграции системы ПО проекту системы ПО;
- адекватны ли атрибуты каждого основного компонента проекта системы ПО в отношении:
  - реализуемости требуемых характеристик безопасности;
  - возможности проверки при последующей верификации;
  - понимания персоналом, выполняющим разработку и верификацию;

- безопасной модификации, позволяющей выполнять дальнейшее совершенствование программы;
- наличия несоответствий:
  - между специфицированным проектом системы ПО и описанием проекта структуры ПО;
  - между описанием проекта системы ПО и специфицированными тестами интеграции системы ПО;
  - между тестами интеграции системы ПО и специфицированными тестами интеграции структуры.

5.8.2.11 Верификация проекта модулей ПО

После завершения разработки и спецификации каждого программного модуля верификация должна обеспечить проверку:

- удовлетворяет ли специфицированный проект программного модуля проекту системы ПО;
- адекватны ли специфицированные проверки каждого программного модуля проекту программного модуля;
- адекватны ли атрибуты каждого программного модуля в отношении:
  - реализуемости требуемых характеристик безопасности;
  - возможности проверки при последующей верификации;
  - понимания персоналом, выполняющим разработку и верификацию;
  - безопасной модификации, позволяющей выполнять дальнейшее совершенствование программы;
- наличия несоответствий:
  - между специфицированным проектом программного модуля и специфицированным проектом системы ПО;
  - между специфицированным проектом каждого программного модуля и специфицированными тестами программных модулей;
  - между специфицированными тестами программных модулей и специфицированными тестами интеграции системы ПО.

5.8.2.12 Верификация исходного текста

Для гарантирования соответствия исходного текста программы специфицированным проектам программных модулей, необходимым стандартам кодирования и требованиям планирования безопасности исходный текст должен быть верифицирован с использованием статических методов.

Примечание — На ранних стадиях жизненного цикла ПО верификация является статической (например, изучение, просмотр, формальная проверка и т.п.). Верификация исходного текста включает в себя такие методы, как просмотр и прогон ПО. Сочетание положительных результатов верификации исходных текстов и тестирования программных модулей гарантирует, что каждый программный модуль будет соответствовать своей спецификации.

5.8.2.13 Верификация данных

При верификации данных должны быть предусмотрены следующие проверки:

- структуры данных, специфицированные во время проектирования, должны быть проверены в части:
  - полноты;
  - согласованности;
  - защиты от изменения или повреждения;
  - соответствия функциональным требованиям системы, управляемой данными;
- прикладные данные должны быть проверены в части:
  - соответствия структурам данных;
  - полноты;
  - совместимости с базовым ПО (например, в отношении последовательности исполнения, совместимости на этапе исполнения и др.);
  - правильности значений данных.

Все параметры, которые могут быть изменены, должны быть проверены на защиту:

- от неверных и неопределенных начальных значений;
- от ошибочных, несовместимых или необоснованных значений;
- от несанкционированных изменений;
- от повреждения данных.

Все промышленные интерфейсы и соответствующее ПО (т. е. датчиков и устройств привода, а также автономных интерфейсов) должны быть проверены в части:

- обнаружения ошибок;
- защиты от повреждения;
- подтверждения данных.

5.8.2.14 Верификация данных в процессе эксплуатации СБЗС ПО должна осуществляться автоматически при включении оборудования, периодически, а также по запросу оператора.

6 Оценка функциональной безопасности

6.1 Цели и требования к оценке соответствия СБЗС ПО устанавливаются по ГОСТ Р 53195.2—2008 (раздел 8).

6.2 Минимальный уровень независимости лиц или организаций, выполняющих оценку функциональной безопасности, должен устанавливаться по ГОСТ 53195.2—2008 (8.14, таблицы 3 и 4).

6.3 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 дает полной гарантии достижения необходимой полноты безопасности. Лицами или организациями, выполняющими оценку соответствия, должно быть также рассмотрено и учтено следующее:

- совместимость и взаимное дополнение выбранных методов, языков и инструментов для всего цикла разработки;
- полностью ли понимают разработчики методы, языки и инструменты, которые они используют;
- насколько хорошо адаптированы методы, языки и инструменты к конкретным проблемам, с которыми приходится сталкиваться при разработке.

Приложение А
(справочное)

Руководство по выбору методов и средств

Некоторые из подразделов настоящего стандарта связаны с таблицами, приведенными в приложениях А и Б. Например, подраздел 5.6 связан с таблицей А.1. Более подробные таблицы, которые раскрывают содержание некоторых элементов таблиц приложения А, содержатся в приложении Б. Например, таблица Б.2 раскрывает содержание динамического анализа и тестирования из таблицы А.5.

Для каждого метода/средства, упомянутого в приложениях А и Б, даны рекомендации по уровню полноты безопасности, изменяющемуся от SIL1 до SIL4. Эти рекомендации обозначаются следующим образом:

КР (HR) — метод/средство, крайне рекомендуемые для данного уровня полноты безопасности. Если указанные метод/средство не используют, то на этапе планирования должно быть дано подробное обоснование отказа от их применения, согласованное с экспертом;

Р (R) — метод/средство, рекомендуемые для данного уровня полноты безопасности. Степень обязательности их применения ниже, чем в случае рекомендации КР (HR).

-- — метод/средство, не имеющие рекомендаций по применению или неприменению;

НР (NR) — метод/средство, не рекомендуемые к применению для данного уровня полноты безопасности. Если эти метод/средство применяют, то на стадии планирования должно быть приведено подробное обоснование его применения, которое необходимо согласовать с экспертом.

Методы/средства следует выбирать в соответствии с уровнем полноты безопасности. Альтернативные или эквивалентные методы/средства обозначены буквой, следующей за номером. Следует выполнять только один из альтернативных или эквивалентных методов/средств.

Ранжирование методов/средств связано с концепцией эффективности, используемой в ГОСТ Р 53195.3. При прочих равных условиях методы, имеющие ранг КР (HR), будут более эффективны в предотвращении внесения систематических ошибок при разработке ПО либо при разработке структуры программ, будут более эффективны при выявлении ошибок, оставшихся не обнаруженными на этапе выполнения программ, по сравнению с методами, имеющими ранг Р (R).

При большом числе факторов, влияющих на полноту безопасности ПО, невозможно установить алгоритм, определяющий такую комбинацию методов и средств, которая была бы корректной для любого заданного приложения.

При планировании безопасности для конкретного приложения должна быть установлена соответствующая комбинация подлежащих использованию методов/средств, если примечания к таблице не налагают иных требований.

Таблица А.1 — Рекомендации по применению методов/средств проектирования и разработки ПО (см. 5.5, 5.6.4)

Методы/средства проектирования и разработки ПО	Ссылка на структурную единицу стандарта	Ранг применимости методов/средств для			
		SIL1	SIL2	SIL3	SIL4
1 Компьютерные средства разработки спецификаций	В.2.4, ГОСТ Р 53195.5—2010	Р (R)	Р (R)	КР (HR)	КР (HR)
2а Полуформальные методы	Таблица Б.7, настоящий стандарт	Р (R)	Р (R)	КР (HR)	КР (HR)
2б Формальные методы, использующие, например, CCS, CSP, HOL, OBJ, LOTOS, временную логику, VDM и Z	В.2.4, ГОСТ Р 53195.5—2010	—	Р (R)	Р (R)	КР (HR)

Примечания

1 Спецификация требований к безопасности программного обеспечения всегда будет требовать описания задачи на естественном языке и использования необходимой системы математических обозначений, отражающих содержание приложения.

2 Таблица отражает дополнительные требования для ясного и точного определения требований к безопасности программного обеспечения.

Таблица А.2 — Рекомендации по применимости методов/средств для проектирования и разработки ПО: проектирование структуры программ (см. 5.6.5)

Методы/средства проектирования структуры программ	Ссылка на структурную единицу стандарта	Ранг применимости методов/средств для			
		SIL1	SIL2	SIL3	SIL4
1 Обнаружение и диагностика сбоев и ошибок	В.3.1, ГОСТ Р 53195.5—2010	—	Р (R)	КР (HR)	КР (HR)
2 Обнаружение и исправление ошибок	В.3.2, ГОСТ Р 53195.5—2010	Р (R)	Р (R)	Р (R)	КР (HR)
3а Программирование с проверкой ошибок	В.3.3, ГОСТ Р 53195.5—2010	Р (R)	Р (R)	Р (R)	КР (HR)
3б Методы «подушки безопасности»	В.3.4, ГОСТ Р 53195.5—2010	—	Р (R)	Р (R)	Р (R)
3в Многовариантное программирование	В.3.5, ГОСТ Р 53195.5—2010	Р (R)	Р (R)	Р (R)	КР (HR)
3г Блоки восстановления	В.3.6, ГОСТ Р 53195.5—2010	Р (R)	Р (R)	Р (R)	Р (R)
3д Восстановление предыдущего состояния	В.3.7, ГОСТ Р 53195.5—2010	Р (R)	Р (R)	Р (R)	Р (R)
3е Прямое восстановление	В.3.8, ГОСТ Р 53195.5—2010	Р (R)	Р (R)	Р (R)	Р (R)
3ж Повторный запуск механизмов восстановления после ошибок	В.3.9, ГОСТ Р 53195.5—2010	Р (R)	Р (R)	Р (R)	КР (HR)
3и Сохранение достигнутых состояний	В.3.10, ГОСТ Р 53195.5—2010	—	Р (R)	Р (R)	КР (HR)
4 Постепенное отключение функций	В.3.11, ГОСТ Р 53195.5—2010	Р (R)	Р (R)	КР (HR)	КР (HR)
5 Исправление ошибок методами искусственного интеллекта	В.3.12, ГОСТ Р 53195.5—2010	—	НР (NR)	НР (NR)	НР (NR)
6 Динамическая реконфигурация	В.3.13, ГОСТ Р 53195.5—2010	—	НР (NR)	НР (NR)	НР (NR)
7а Структурные методы, включая, например, JSD, MASCOT, SADT и Yourdon	В.2.1, ГОСТ Р 53195.5—2010	КР (HR)	КР (HR)	КР (HR)	КР (HR)
7б Полуформальные методы	Таблица Б.7, настоящий стандарт	Р (R)	Р (R)	КР (HR)	КР (HR)
8 Компьютерные средства разработки спецификаций	Б.2.4, ГОСТ Р 53195.5—2010	Р (R)	Р (R)	КР (HR)	КР (HR)

Примечание — Приведенные в данной таблице средства, касающиеся устойчивости к ошибкам (контроль ошибок), должны рассматриваться совместно с требованиями, описанными в ГОСТ Р 53195.3 к структуре и контролю ошибок для АС программируемых электронных устройств.

Таблица А.3 — Рекомендации по применимости методов/средств для проектирования и разработки ПО: инструментальные средства поддержки и языки программирования (см. 5.6.6)

Методы/средства для проектирования инструментальных средств поддержки и языки программирования	Ссылка на структурную единицу ГОСТ Р 53195.5—2010	Ранг применимости методов/средств для			
		SIL1	SIL2	SIL3	SIL4
1 Выбор соответствующего языка программирования	В.4.7	КР (HR)	КР (HR)	КР (HR)	КР (HR)
2 Строго типизированные языки программирования	В.4.1	КР (HR)	КР (HR)	КР (HR)	КР (HR)
3 Подмножество языка	В.4.2	—	—	КР (HR)	КР (HR)
4а Сертифицированные средства	В.4.3	Р (R)	КР (HR)	КР (HR)	КР (HR)
4б Инструментальные средства, заслуживающие доверия на основании опыта использования	В.4.4	КР (HR)	КР (HR)	КР (HR)	КР (HR)
4в Сравнение исходных программ и исполнимых кодов	В.4.5	КР (HR)	КР (HR)	КР (HR)	КР (HR)
5а Сертифицированный компилятор	В.4.3	Р (R)	КР (HR)	КР (HR)	КР (HR)
5б Трансляторы, заслуживающие доверия на основании опыта использования	В.4.4	КР (HR)	КР (HR)	КР (HR)	КР (HR)
6 Библиотека проверенных/верифицированных модулей и компонентов	В.4.6	Р (R)	КР (HR)	КР (HR)	КР (HR)

Таблица А.4 — Рекомендации по применимости методов/средств для проектирования и разработки ПО: детальное проектирование (см. 5.6.4, 5.6.5 и 5.6.7)

Методы/средства детального проектирования	Ссылка на структурную единицу стандарта	Ранг применимости методов/средств для			
		SIL1	SIL2	SIL3	SIL4
1а Структурные методы, включая, например, JSD, MASCOT, SADT и Yourdon	В.2.1, ГОСТ Р 53195.5—2010	КР (HR)	КР (HR)	КР (HR)	КР (HR)
1б Полуформальные методы	Таблица Б.7, настоящий стандарт	Р (R)	КР (HR)	КР (HR)	КР (HR)
1в Формальные методы, включая, например, CCS, CSP, HOL, LOTOS, OBJ, временную логику, VDM и Z	В.2.4, ГОСТ Р 53195.5—2010	—	Р (R)	Р (R)	КР (HR)
2 Средства автоматизированного проектирования	Б.3.5—2010, ГОСТ Р 53195.5—2010	Р (R)	Р (R)	КР (HR)	КР (HR)
3 Программирование с защитой	В.2.5—2010, ГОСТ Р 53195.5—2010	—	Р (R)	КР (HR)	КР (HR)
4 Модульный подход	Таблица Б.9, настоящий стандарт	КР (HR)	КР (HR)	КР (HR)	КР (HR)
5 Стандарты для проектирования и кодирования	Таблица Б.1, настоящий стандарт	Р (R)	КР (HR)	КР (HR)	КР (HR)
6 Структурное программирование	В.2.7, ГОСТ Р 53195.5—2010	КР (HR)	КР (HR)	КР (HR)	КР (HR)
7 Использование проверенных/верифицированных программных модулей и компонентов (по возможности)	В.2.10, В.4.6, ГОСТ Р 53195.5—2010	Р (R)	КР (HR)	КР (HR)	КР (HR)

Таблица А.5 — Рекомендации по применимости методов/средств для проектирования и разработки ПО: тестирование программных модулей и интеграция (см. 5.6.9 и 5.6.10)

Методы/средства тестирования программных модулей и интеграции	Ссылка на структурную единицу стандарта	Ранг применимости методов/средств для			
		SIL1	SIL2	SIL3	SIL4
1 Вероятностное тестирование	В.5.1, ГОСТ Р 53195.5—2010	—	Р (R)	Р (R)	КР (HR)
2 Динамический анализ	Б.6.4, ГОСТ Р 53195.5—2010; таблица Б.2, настоящий стандарт	Р (R)	КР (HR)	КР (HR)	КР (HR)
3 Регистрация и анализ данных	В.5.2, ГОСТ Р 53195.5—2010	КР (HR)	КР (HR)	КР (HR)	КР (HR)
4 Функциональное тестирование и тестирование методом «черного ящика»	Б.5.1, Б.5.2, ГОСТ Р 53195.5—2010; таблица Б.3, настоящий стандарт	КР (HR)	КР (HR)	КР (HR)	КР (HR)
5 Моделирование реализации	В.5.20, ГОСТ Р 53195.5—2010; таблица Б.6, настоящий стандарт	Р (R)	Р (R)	КР (HR)	КР (HR)
6 Тестирование интерфейса	В.5.3, ГОСТ Р 53195.5—2010	Р (R)	Р (R)	КР (HR)	КР (HR)
Примечание — Тестирование программных модулей и интеграции относится к процессам верификации (см. таблицу А.9 настоящего приложения).					

Таблица А.6 — Рекомендации по применимости методов/средств для интеграции программируемой электроники (ПО и АС) (см. 5.6.11)

Методы/средства для интеграции РЕ (ПО и АС)	Ссылка на структурную единицу стандарта	Ранг применимости методов/средств для			
		SIL1	SIL2	SIL3	SIL4
1 Функциональное тестирование и тестирование методом «черного ящика»	Б.5.1, Б.5.2, ГОСТ Р 53195.5—2010; таблица Б.3, настоящий стандарт	КР (HR)	КР (HR)	КР (HR)	КР (HR)
2 Моделирование выполнения	В.5.20, ГОСТ Р 53195.5—2010; таблица Б.6, настоящий стандарт	Р (R)	КР (HR)	КР (HR)	КР (HR)
Примечание — Тестирование программных модулей и интеграции относится к процессам верификации (см. таблицу А.9 настоящего приложения).					

Таблица А.7 — Рекомендации по применимости методов/средств подтверждения соответствия ПО (см. 5.6.3, 5.7.3)

Методы/средства для подтверждения соответствия ПО	Ссылка на структурную единицу стандарта	Ранг применимости методов/средств для			
		SIL1	SIL2	SIL3	SIL4
1 Вероятностное тестирование	В.5.1, ГОСТ Р 53195.5—2010	—	Р (R)	Р (R)	КР (HR)
2 Имитация/моделирование	Таблица Б.5, настоящий стандарт	Р (R)	Р (R)	КР (HR)	КР (HR)
3 Функциональное тестирование и тестирование методом «черного ящика»	Б.5.1, Б.5.2, ГОСТ Р 53195.5—2010; таблица Б.3, настоящий стандарт	КР (HR)	КР (HR)	КР (HR)	КР (HR)

Таблица А.8 — Рекомендации по применимости методов/средств для модификации ПО (см. 5.7.4)

Методы/средства для модификации ПО	Ссылка на структурную единицу стандарта	Ранг применимости методов/средств для			
		SIL1	SIL2	SIL3	SIL4
1 Анализ влияния	В.5.23, ГОСТ Р 53195.5—2010	КР (HR)	КР (HR)	КР (HR)	КР (HR)
2 Имитация/моделирование	В.5.23, ГОСТ Р 53195.5—2010	Р (R)	КР (HR)	КР (HR)	КР (HR)
3 Повторная верификация программных модулей, на которые оказывают влияние изменения в других модулях	В.5.23, ГОСТ Р 53195.5—2010	КР (HR)	КР (HR)	КР (HR)	КР (HR)
4 Повторная верификация системы в целом	В.5.23, ГОСТ Р 53195.5—2010	—	Р (R)	КР (HR)	КР (HR)
5 Управление конфигурацией ПО	В.5.24, ГОСТ Р 53195.5—2010	КР (HR)	КР (HR)	КР (HR)	КР (HR)
6 Регистрация и анализ данных	В.5.2, ГОСТ Р 53195.5—2010	КР (HR)	КР (HR)	КР (HR)	КР (HR)

Таблица А.9 — Рекомендации по применимости методов/средств верификации ПО (см. 5.8)

Методы/средства для верификации ПО	Ссылка на структурные единицы стандартов	Ранг применимости методов/средств для			
		SIL1	SIL2	SIL3	SIL4
1 Формальное доказательство	В.5.13, ГОСТ Р 53195.5—2010	—	Р (R)	Р (R)	КР (HR)
2 Вероятностное тестирование	В.5.1, ГОСТ Р 53195.5—2010	—	Р (R)	Р (R)	КР (HR)
3 Статический анализ	Б.6.3, ГОСТ Р 53195.5—2010; таблица Б.8, настоящий стандарт	Р (R)	КР (HR)	КР (HR)	КР (HR)
4 Динамический анализ	Б.6.5, ГОСТ Р 53195.5—2010; таблица Б.2, настоящий стандарт	Р (R)	КР (HR)	КР (HR)	КР (HR)
5 Метрики сложности программного обеспечения	В.5.14, ГОСТ Р 53195.5—2010	Р (R)	Р (R)	Р (R)	Р (R)
6 Тестирование и интеграция программных модулей	Таблица А.5 настоящего приложения				
7 Проверка интеграции программируемых электронных устройств	Таблица А.6 настоящего приложения				
8 Тестирование программной системы (подтверждение соответствия)	Таблица А.7 настоящего приложения				

Примечания

1 В настоящей таблице все процессы, связанные с верификацией, объединены для удобства изложения материала. При ее применении дополнительные требования к элементам верификации, связанным с динамическим тестированием в таблицах А.5 и А.6, которые относятся к процессам верификации, не предъявляются. Не требуется также проведение верификационного тестирования в дополнение к подтверждению соответствия ПО (см. таблицу А.7 настоящего приложения).

2 Верификация предусматривается в ГОСТ Р 53195.1—ГОСТ Р 53195.3. Следовательно, первая верификация СБЗС-системы проводится на более ранних стадиях жизненного цикла системы.

3 На ранних стадиях жизненного цикла СБЗС ПО верификация является статической. Она может включать в себя, например, изучение, просмотр, формальную проверку. Проведение динамического тестирования возможно только после завершения разработки программы. Верификация программ динамическими средствами включает в себя функциональное тестирование, тестирование методом «белого ящика», статистическое тестирование. Для удостоверения в том, что каждый программный модуль удовлетворяет соответствующей спецификации, требуется объединение информации обоих типов.

Таблица А.10 — Рекомендации по применимости методов/средств для оценки функциональной безопасности ПО (см. раздел 6)

Методы/средства для оценки функциональной безопасности ПО	Ссылка на структурную единицу стандарта	Ранг применимости методов/средств для			
		SIL1	SIL2	SIL3	SIL4
1 Таблица контрольных проверок	Б.2.5, ГОСТ Р 53195.5—2010	Р (R)	Р (R)	Р (R)	Р (R)
2 Таблицы решений и таблицы истинности	В.6.1, ГОСТ Р 53195.5—2010	Р (R)	Р (R)	Р (R)	Р (R)
3 Метрики сложности программного обеспечения	В.5.14, ГОСТ Р 53195.5—2010	Р (R)	Р (R)	Р (R)	Р (R)
4 Анализ отказов	Таблица Б.4, настоящий стандарт	Р (R)	Р (R)	КР (HR)	КР (HR)
5 Анализ отказов по общей причине	В.6.3, ГОСТ Р 53195.5—2010	—	Р (R)	КР (HR)	КР (HR)
6 Структурные схемы надежности	В.6.5, ГОСТ Р 53195.5—2010	Р (R)	Р (R)	Р (R)	Р (R)

Приложение Б
(справочное)

Подробные таблицы

Таблица Б.1 — Рекомендации по применению стандартов для проектирования и кодирования (упомянутых в таблице А.4 приложения А)

Методы/средства для оценки функциональной безопасности ПО	Ссылка на структурную единицу ГОСТ Р 53195.5—2010	Ранг применимости методов/средств для			
		SIL1	SIL2	SIL3	SIL4
1 Использование стандартов кодирования	B.2.6.2	КР (HR)	КР (HR)	КР (HR)	КР (HR)
2 Отказ от использования динамических объектов	B.2.6.3	Р (R)	КР (HR)	КР (HR)	КР (HR)
3а Отказ от использования динамических переменных	B.2.6.3	—	Р (R)	КР (HR)	КР (HR)
3б Проверка создания динамических переменных при выполнении программы	B.2.6.4	—	Р (R)	КР (HR)	КР (HR)
4 Ограниченное использование прерываний	B.2.6.5	Р (R)	Р (R)	КР (HR)	КР (HR)
5 Ограниченное использование указателей	B.2.6.6	—	Р (R)	КР (HR)	КР (HR)
6 Ограниченное использование рекурсий	B.2.6.7	—	Р (R)	КР (HR)	КР (HR)
7 Неиспользование безусловных переходов в программах, написанных на языках высокого уровня	B.2.6.2	Р (R)	КР (HR)	КР (HR)	КР (HR)
Примечание — Применение методов 2 и 3а не требуется, если использован компилятор, который гарантирует выделение достаточного объема памяти для всех динамических переменных и объектов до начала выполнения программы либо вставляет проверки корректного выделения памяти в процессе выполнения.					

Таблица Б.2 — Рекомендации по применению методов/средств динамического анализа и тестирования (упомянутых в таблицах А.5 и А.9 приложения А)

Методы/средства динамического анализа и тестирования	Ссылка на структурную единицу ГОСТ Р 53195.5—2010	Ранг применимости методов/средств для			
		SIL1	SIL2	SIL3	SIL4
1 Выполнение контрольного примера, начиная с анализа граничных значений	B.5.4	Р (R)	КР (HR)	КР (HR)	КР (HR)
2 Выполнение контрольного примера, начиная с предполагаемой ошибки	B.5.5	Р (R)	Р (R)	Р (R)	Р (R)
3 Выполнение контрольного примера, начиная с введения ошибки	B.5.6	—	Р (R)	КР (HR)	КР (HR)
4 Моделирование реализации	B.5.20	Р (R)	Р (R)	Р (R)	КР (HR)
5 Разделение входных данных на классы эквивалентности	B.5.7	Р (R)	Р (R)	Р (R)	КР (HR)
6 Структурное тестирование	B.5.8	Р (R)	Р (R)	КР (HR)	КР (HR)
Примечание — Анализ с использованием тестовых примеров проводят на уровне подсистем. Он основывается на спецификациях и/или спецификациях и текстах программ.					

Таблица Б.3 — Рекомендации по применению методов/средств функционального тестирования и тестирования методом «черного ящика» (упомянутых в таблицах А.5, А.6 и А.7 приложения А)

Методы/средства функционального тестирования и тестирования методом «черного ящика»	Ссылка на структурную единицу ГОСТ Р 53195.5—2010	Ранг применимости методов/средств для			
		SIL1	SIL2	SIL3	SIL4
1 Выполнение контрольного примера, начиная с причинно-следственных диаграмм	Б.6.5.2	—	—	Р (R)	Р (R)
2 Макетирование/анимация	В.5.17	—	—	Р (R)	Р (R)
3 Анализ граничных значений	В.5.4	Р (R)	КР (HR)	КР (HR)	КР (HR)
4 Разделение входных данных на классы эквивалентности	В.5.7	Р (R)	КР (HR)	КР (HR)	КР (HR)
5 Моделирование процесса	В.5.18	Р (R)	Р (R)	Р (R)	Р (R)
Примечания 1 Анализ с использованием контрольных примеров выполняется на уровне систем ПО и основывается только на спецификациях. 2 Необходимая полнота моделирования должна выбираться в зависимости от уровня полноты безопасности, сложности и условий применения.					

Таблица Б.4 — Рекомендации по применению методов/средств анализа отказов (упомянутых в таблице А.10 приложения А)

Методы/средства анализа отказов	Ссылка на структурную единицу ГОСТ Р 53195.5—2010	Ранг применимости методов/средств для			
		SIL1	SIL2	SIL3	SIL4
1а Диаграммы последовательностей событий	Б.6.5.2	Р (R)	Р (R)	Р (R)	Р (R)
1б Анализ методом дерева событий	Б.6.5.3	Р (R)	Р (R)	Р (R)	Р (R)
2 Анализ методом дерева отказов	Б.6.5.5	Р (R)	Р (R)	КР (HR)	КР (HR)
3 Анализ видов отказов и критичности компонентов	Б.6.5.4	Р (R)	Р (R)	КР (HR)	КР (HR)
4 Моделирование методом Монте-Карло	В.6.6	Р (R)	Р (R)	Р (R)	Р (R)
Примечание — Для отнесения ПО к соответствующему уровню полноты безопасности должен быть предварительно выполнен анализ рисков.					

Таблица Б.5 — Рекомендации по применению методов/средств моделирования (упомянутого в таблице А.7 приложения А)

Методы/средства моделирования	Ссылка на структурную единицу ГОСТ Р 53195.5—2010	Ранг применимости методов/средств для			
		SIL1	SIL2	SIL3	SIL4
1 Диаграммы потоков данных	В.2.2	Р (R)	Р (R)	Р (R)	Р (R)
2 Метод конечных автоматов	Б.2.3.2	—	Р (R)	КР (HR)	КР (HR)
3 Формальные методы	В.2.4	—	Р (R)	Р (R)	КР (HR)
4 Моделирование реализации	В.5.20	Р (R)	КР (HR)	КР (HR)	КР (HR)
5 Метод сетей Петри	Б.2.3.3	—	Р (R)	КР (HR)	КР (HR)
6 Макетирование/анимация	В.5.17	Р (R)	Р (R)	Р (R)	Р (R)
7 Структурные диаграммы	В.2.3	Р (R)	Р (R)	Р (R)	КР (HR)
Примечание — Должны быть приняты во внимание и другие методы/средства, не указанные в настоящей таблице.					

Таблица Б.6 — Рекомендации по применению методов/средств тестирования характеристик (упомянутых в таблицах А.5 и А.6 приложения А)

Методы/средства тестирования характеристик	Ссылка на структурную единицу ГОСТ Р 53195.5—2010	Ранг применимости методов/средств для			
		SIL1	SIL2	SIL3	SIL4
1 Тестирование в предельных режимах	В.5.21	Р (R)	Р (R)	КР (HR)	КР (HR)
2 Ограничение временных интервалов и объема памяти	В.5.22	КР (HR)	КР (HR)	КР (HR)	КР (HR)
3 Требования к реализации	В.5.19	КР (HR)	КР (HR)	КР (HR)	КР (HR)

Таблица Б.7 — Рекомендации по применению полуформальных методов/средств (упомянутых в таблицах А.1, А.2 и А.4 приложения А)

Полуформальные методы/средства	Ссылка на структурную единицу ГОСТ Р 53195.5—2010	Ранг применимости методов/средств для			
		SIL1	SIL2	SIL3	SIL4
1 Логические/функциональные блок-схемы	В.7	Р (R)	Р (R)	КР (HR)	КР (HR)
2 Диаграммы последовательности	В.7	Р (R)	Р (R)	КР (HR)	КР (HR)
3 Диаграммы потоков данных	В.2.2	Р (R)	Р (R)	Р (R)	Р (R)
4 Метод конечных автоматов/диаграммы переходов	Б.2.3.2	Р (R)	Р (R)	КР (HR)	КР (HR)
5 Метод сетей Петри	Б.2.3.3	Р (R)	Р (R)	КР (HR)	КР (HR)
6 Таблицы решений и таблицы истинности	В.6.1	Р (R)	Р (R)	КР (HR)	КР (HR)

Таблица Б.8 — Рекомендации по применению методов/средств статического анализа (упомянутого в таблице А.9 приложения А)

Методы/средства статического анализа	Ссылка на структурную единицу ГОСТ Р 53195.5—2010	Ранг применимости методов/средств для			
		SIL1	SIL2	SIL3	SIL4
1 Анализ граничных значений	В.5.4	Р (R)	Р (R)	КР (HR)	КР (HR)
2 Таблица контрольных проверок	Б.2.5	Р (R)	Р (R)	Р (R)	Р (R)
3 Анализ потоков управления	В.5.9	Р (R)	КР (HR)	КР (HR)	КР (HR)
4 Анализ потоков данных	В.5.10	Р (R)	КР (HR)	КР (HR)	КР (HR)
5 Предположение ошибок	В.5.5	Р (R)	Р (R)	Р (R)	Р (R)
6 Инспекция программ по Фейгану	В.5.15	—	Р (R)	Р (R)	КР (HR)
7 Анализ скрытых путей исполнения	В.5.11	—	—	Р (R)	Р (R)
8 Тестирование на символьном уровне	В.5.12	Р (R)	Р (R)	КР (HR)	КР (HR)
9 Сквозной прогон/просмотр проекта	В.5.16	КР (HR)	КР (HR)	КР (HR)	КР (HR)

Таблица Б.9 — Рекомендации по применению методов/средств модульного подхода (упомянутого в таблице А.4 приложения А)

Методы/средства статического анализа	Ссылка на структурную единицу ГОСТ Р 53195.5—2010	Ранг применимости методов/средств для			
		SIL1	SIL2	SIL3	SIL4
1 Ограничение размера программного модуля	В.2.9	КР (HR)	КР (HR)	КР (HR)	КР (HR)
2 Ограничение доступа/инкапсуляция информации	В.2.8	Р (R)	КР (HR)	КР (HR)	КР (HR)
3 Ограничение числа параметров	В.2.9	Р (R)	Р (R)	Р (R)	Р (R)
4 Одна точка входа и одна точка выхода в каждой подпрограмме и функции	В.2.9	КР (HR)	КР (HR)	КР (HR)	КР (HR)
5 Полностью определенный интерфейс	В.2.9	КР (HR)	КР (HR)	КР (HR)	КР (HR)

Библиография

[1] МЭК 61131-3:2003 Программируемые контроллеры. Часть 3. Языки программирования (IEC 61131-3:2003 Programmable controllers — Part 3: Programming languages)

УДК 621.5:814.8:006.354 ОКС 13.110, 13.220.01, 13.310, 13.320, 29.130.20, 35.240 ОКП 43 7000, 43 7100, 43 7200, 43 7280, 70 3000

Ключевые слова: безопасность функциональная, связанные с безопасностью зданий и сооружений системы, требования к программному обеспечению

Редактор *Л.В. Коретникова*
Технический редактор *В.Н. Прусакова*
Корректор *М.В. Бучная*
Компьютерная верстка *Л.А. Круговой*

Сдано в набор 25.10.2018. Подписано в печать 20.11.2018. Формат 60×84 1/8. Гарнитура Ариал.
Усл. печ. л. 4,65. Уч.-изд. л. 4,21.
Подготовлено на основе электронной версии, предоставленной разработчиком стандарта

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

gost.e-ecolog.ru gost.e-ecolog.ru

gost.e-ecolog.ru