ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ УНИТАРНОЕ ПРЕДПРИЯТИЕ
«ВСЕРОССИЙСКИЙ НАУЧНО-ИССЛЕДОВАТЕЛЬСКИЙ ИНСТИТУТ МЕТРОЛОГИЧЕСКОЙ СЛУЖБЫ»
(ФГУП «ВНИИМС»)
ФЕДЕРАЛЬНОГО АГЕНСТВА ПО ТЕХНИЧЕСКОМУ РЕГУЛИРОВАНИЮ И МЕТРОЛОГИИ

УТВЕЖДАЮ
Директор
_____ А.Ю. Кузин
«07» ноября 2016 г.

РЕКОМЕНДАЦИЯ
ГОСУДАРСТВЕННАЯ СИСТЕМА ОБЕСПЕЧЕНИЯ ЕДИНСТВА ИЗМЕРЕНИЙ

ТИПОВАЯ МЕТОДИКА ИСПЫТАНИЙ И ПОДТВЕРЖДЕНИЯ СООТВЕТСТВИЯ (СЕРТИФИКАЦИИ) ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ СРЕДСТВ ИЗМЕРЕНИЙ

МИ 3566 - 2016

Москва
2016 г.

ПРЕДИСЛОВИЕ

РАЗРАБОТАНА Федеральным государственным унитарным предприятием «Всероссийский научно - исследовательский институт метрологической службы» (ФГУП «ВНИИМС»)

ИСПОЛНИТЕЛИ Ю.А. Кудеяров, д.ф.-м.н., профессор, (руководитель темы), А.Н. Паньков, к.т.н.

УТВЕРЖДЕНА ФГУП «ВНИИМС» «07» ноября 2016 г.

ЗАРЕГИСТРИРОВАНА ФГУП «ВНИИМС» «07» ноября 2016 г.

ВВЕДЕНА ВПЕРВЫЕ

СОДЕРЖАНИЕ

	Группа Т 80
Государственная система обеспечения единства измерений	РЕКОМЕНДАЦИЯ
Типовая методика испытаний и подтверждения соответствия (сертификации) программного обеспечения средств измерений	МИ 3566-2016

1. НАЗНАЧЕНИЕ И ОБЛАСТЬ ПРИМЕНЕНИЯ

Настоящая рекомендация разработана в целях реализации требований к программному обеспечению (далее – ПО) средств измерений (далее – СИ) и его алгоритмам в соответствии со статьей 9 Федерального закона Российской Федерации от 26 июня 2008 г. № 102-ФЗ «Об обеспечении единства измерений», с Федеральным законом Российской Федерации от 22 декабря 2002 г. № 184-ФЗ «О техническом регулировании», Порядком проведения испытаний стандартных образцов или средств измерений в целях утверждения типа и Порядком выдачи свидетельств об утверждении типа стандартных образцов или типа средств измерений, установления и изменения срока действия указанных свидетельств и интервала между поверками средств измерений, утвержденных приказом Минпромторга России от 30 ноября 2009 г. № 1081, национальным стандартом ГОСТ Р 8.654, и другими нормативными документами, указанными в разделе 2 настоящей рекомендации.

Настоящая рекомендация устанавливает методы испытаний ПО СИ и его алгоритмов и распространяются на:

ПО СИ, в том числе измерительных и информационно-измерительных систем, и его алгоритмы;

ПО автоматизированных систем, функционирующих с использованием СИ или компонентов измерительных систем, и его алгоритмы;

ПО контроллеров, вычислительных блоков, не входящих в состав измерительных систем, а также технических систем и устройств с измерительными функциями, осуществляющих обработку и представление измерительной информации, и его алгоритмы.

Рекомендации также могут быть использованы при испытаниях ПО СИ, не входящих в сферу государственного регулирования в области обеспечения единства измерений.

В настоящей методике применены следующие сокращения:

ЕСПД – Единая система программной документации;

Методика – Методика испытаний;

МХ – Метрологические характеристики;

ПО – Программное обеспечение;

СИ – Средство измерения;

ФЗ – Федеральный закон.

2. НОРМАТИВНЫЕ ССЫЛКИ

В настоящей рекомендации использованы нормативные ссылки на следующие стандарты:

ГОСТ Р 8.839 – 2013 Государственная система обеспечения единства измерений. Общие требования к измерительным приборам с программным управлением

ГОСТ Р 8.654 – 2015 ГСИ. Требования к программному обеспечению средств измерений. Основные положения

ГОСТ Р 8.596 – 2002 ГСИ. Метрологическое обеспечение измерительных систем. Основные положения

Р 50.2.077-2014 ГСИ. Испытания средств измерений с целью утверждения типа.Проверка защиты программного обесепечения.

ГОСТ ИСО/МЭК 17025 – 2011 Общие требования к компетентности испытательных и калибровочных лабораторий

ГОСТ Р ИСО 5725-1 – 2002 Точность (правильность и прецизионность) методов и результатов измерений. Часть I. Основные положения и определения

Примечание - При пользовании настоящей рекомендацией 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 ежегодно издаваемому информационному указателю «Национальные стандарты», который опубликован по состоянию на 1 января текущего года, и по соответствующим ежемесячно издаваемым информационным указателям, опубликованным в текущем году.

3. ТЕРМИНЫ И ОПРЕДЕЛЕНИЯ

В настоящей рекомендации применяются термины и определения в соответствии с ГОСТ Р 8.654-2015, кроме того, используются следующие термины:

3.1 **Алгоритмы программного обеспечения:** Последовательности арифметических и логических операций, производимых над измерительной информацией (с учетом априорной информации) с целью определения результатов измерений, а также для реализации хранения, защиты и передачи измерительной информации.

Алгоритмы могут быть заданы различными способами, в том числе, представлены в явном виде (конечной последовательностью арифметических и логических операций), или в виде рекуррентной процедуры.

3.2 **Закрытая сеть:** Сеть из фиксированного числа участников с известными адресами, функциями и пунктами их местонахождения.

[OIML D 31 Edition 2008 E, пункт 3.1.6]

3.3. **Интегрированная память:** Запоминающее устройство, являющееся частью средства измерений.

[WELMEC 7.2. Issue 5, раздел 2 «Terminology»]

3.4 **Опорное программное обеспечение:** Программное обеспечение, используемое для сравнения с испытываемым программным обеспечением и отвечающее повышенным требованиям к его вычислительным и функциональным характеристикам, подтвержденным (в ряде случаев независимыми методами) при его неоднократном тестировании и применении.

3.5 **Открытая сеть:** Сеть с произвольным числом участников (устройств с произвольными функциями). Число, идентификация и локализация участников могут изменяться и быть неизвестными другим участникам.

[OIML D 31 Edition 2008 E, пункт 3.1.35]

3.6 **Тестирование программного обеспечения и алгоритмов:** Серия технических операций (функциональных проверок) для подтверждения соответствия испытуемого ПО и его алгоритмов требованиям нормативных документов.

[OIML D 11 Edition:2004, п. 3.20]

Примечание – Тестирование программного обеспечения является, в частности, частью процедуры испытаний СИ в целях утверждения типа, когда проверяются уровень защиты ПО, его идентификационные признаки, а также при оценке влияния ПО на метрологические характеристики СИ.

4. ОБЩИЕ ПОЛОЖЕНИЯ

4.1 Рекомендация применяется при всех видах подтверждения соответствия ПО СИ, в том числе при испытаниях СИ в целях утверждения типа и при сертификации ПО.

4.2 Под испытаниями ПО СИ и его алгоритмов понимаются работы по определению их характеристик и свойств, в том числе, уровня защиты, идентификационных данных (признаков), степени влияния на метрологические характеристики СИ с целью подтверждения их соответствия требованиям ГОСТ Р 8.654-2015 и/или других нормативных документов.

4.3. При испытаниях ПО СИ и его алгоритмов должна быть обеспечена конфиденциальность предоставляемой заявителем испытаний информации.

4.4. При испытаниях ПО СИ и его алгоритмов должны использоваться методы определения и оценки их характеристик, основанные на международных [1, 2] и отечественных правилах и рекомендациях, которые позволяют с достаточной степенью достоверности подтвердить их соответствие требованиям нормативной документации, указанной в разделе 2 настоящей рекомендации, и определить действительные значения этих характеристик.

4.5. Характеристики ПО СИ и его алгоритмов можно разбить на две группы.

К первой группе относят характеристики, которые в соответствии с приказом Минпромторга России от 30 ноября 2009 г. № 1081 должны быть внесены в описание типа СИ, а именно:

идентификационные данные (признаки);

уровень защиты от непреднамеренных и преднамеренных изменений;

степень влияния на метрологические характеристики (МХ) СИ.

Ко второй группе относятся характеристики, которые не вносятся в описание типа СИ, но без оценки и проверки которых невозможно в полной мере установить действительные значения характеристик ПО в целом, в том числе, значения характеристик, относящихся к первой группе. К таким характеристикам относятся:

степень соответствия ПО сопровождающей документации;

разделение на метрологически значимую и незначимую части;

наличие или отсутствие защищенных интерфейсов;

другие характеристики, согласованные между заявителем испытаний и организацией, проводящей испытания.

5. МЕТОДИКА ИСПЫТАНИЙ И ИХ ОСНОВНЫЕ ЭТАПЫ

5.1. Для проведения испытаний ПО СИ и его алгоритмов, на основе методов, изложенных в настоящей рекомендации, разрабатывается методика испытаний, содержащая детальное описание всех действий, выполняемых в процессе испытаний. В методику рекомендуется включать следующие основные этапы испытаний:

определение перечня исследуемых характеристик и параметров, исходных данных и критериев, которым должны удовлетворять результаты, полученные испытуемым ПО и его алгоритмами;

проведение испытаний в соответствии с методикой испытаний и получение результатов анализа документации и тестирования (функциональных проверок) испытываемого ПО;

обработка результатов испытаний и их оформление в виде протокола.

5.2. Методика испытаний разрабатывается для каждого отдельного ПО СИ с учетом его назначения и функциональных особенностей.

5.3. В методике испытаний:

приводится перечень алгоритмов, характеристик, свойств и параметров ПО, необходимых исходных данных и опорных ПО, а также критерии, позволяющие производить оценку характеристик испытываемого ПО и его алгоритмов;

определяются и описываются методы испытаний, которые должны обеспечить проверку всех основных функций испытываемого ПО, а также его соответствие требованиям к ПО СИ и к его алгоритмам;

поэтапно описывается последовательность действий при проведении испытаний ПО и его алгоритмов.

5.4. По результатам испытаний и проверки идентификационных данных (признаков), степени влияния ПО на МХ СИ и уровня защиты ПО СИ составляется протокол испытаний, подписанный непосредственными исполнителями испытаний и утвержденный руководителем организации, проводящей испытания ПО.

5.5. Результаты испытаний ПО признаются положительными, если при анализе документации и проведении тестирования (функциональных проверок), предусмотренных методикой испытаний, подтверждается соответствие испытываемого ПО требованиям ГОСТ Р 8.654-2015 и/или другой нормативной документации, приведенной в разделе 2 настоящей Рекомендации.

6. МЕТОДЫ ИСПЫТАНИЙ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ СРЕДСТВ ИЗМЕРЕНИЙ И ЕГО АЛГОРИТМОВ

6.1 Проверка документации

6.1.1. Представление всей необходимой документации на испытания в соответствии с требованиями ГОСТ Р 8.654-2015, а также других документов, указанных в разделе 2 настоящей Рекомендации, является необходимым условием их проведения.

6.1.2. В соответствии с требованиями указанных нормативных документов проверяется наличие, достаточность и правильность представленной документации.

6.1.3. В отдельных случаях при проведении испытаний ПО СИ и его алгоритмов документацию рекомендуется дополнять текстами программ или их фрагментами при этом может быть заключен договор о соблюдении конфиденциальности.

6.1.4. Перечень документов, представляемых для испытаний, объем и методы их проверки определяется на этапе разработки методики испытаний и согласуется заявителем испытаний с организацией, проводящей испытания.

6.1.5. В документации на ПО СИ следует предоставить информацию, которая должна содержать сведения из приведенного ниже перечня в той части, которая применима к данному СИ:

обозначение ПО, включающее в себя его наименование, обозначение его версии или версий его модулей;

описание назначения ПО, его структуры и выполняемых функций (структура ПО может быть представлена в виде одного или нескольких взаимосвязанных модулей, реализующих функции ПО, с учетом его разделения, при этом описание структуры ПО может быть осуществлено в графическом виде с пояснениями и/или в текстовой форме);

описание методов и способов идентификации ПО, а также его метрологически значимых частей, функций и параметров, т.е. проверяется наличие информации о методе (алгоритме) идентификации ПО, способах идентификации ПО в соответствии с принятым методом, о системе кодификации номера версии;

описание реализованных в ПО расчетных алгоритмов, а также их блок-схемы, т.е. проверяется описание логических схем алгоритмов, функций, реализуемых алгоритмами ПО, а также всех величин, рассчитываемых с их помощью, с их математическим представлением в виде формул; проверяются также данные о степени округления при расчетах (точность алгоритмов);

описание интерфейсов пользователя, всех меню и диалогов;

описание интерфейсов связи ПО для передачи, обработки и хранения данных в том числе посредством открытых или закрытых сетей связи, т.е. проверяется наличие информации о методе связи СИ и ПО, о данных, получаемых от и передаваемых в СИ программным обеспечением,

наличие описания всех аппаратных и программных компонент СИ, а также описания исполняемых файлов (название, размер и т.п.));

описание реализованных методов защиты ПО и данных, т.е. проверяется описание реализованных методов (авторизация пользователя, журнал событий, кодирование данных и т.д.), защиты ПО и данных от случайных (непреднамеренных) и преднамеренных изменений и искажений, а также наличие в документации описания методов фиксации сообщений об ошибках;

описание способов хранения измеренных данных на встроенном, удаленном или съемном носителе;

описание требуемых системных и аппаратных средств, если эта информация не приведена в руководстве пользователя.

6.1.6. При испытаниях с целью утверждения типа средств измерений, включающих в себя или сопровождаемых ПО, в соответствии с Приложением Б Рекомендаций Р 50.2.077-2014 заказчиком испытаний представляется также «Декларация полноты документации, уровня защиты и отсутствия недокументированных возможностей программного обеспечения средств измерений».

6.1.7. Указанная в 6.1.5 информация может быть представлена в виде программных документов (например, описания программы, пояснительной записки, описания применения, руководства системного программиста, руководства оператора и т. д.) или иной программной документации, имеющейся у заявителя, при этом при ее составлении можно руководствоваться рекомендациями единой системы программной документации (ЕСПД) и/или другой аналогичной документации.

6.1.8. Результаты проверки, в том числе выявленные несоответствия, полученные при анализе документации ПО, заносятся в протоколы испытаний.

6.2 Проверка разделения программного обеспечения

6.2.1. Разделение ПО СИ проводят в целях выделения в составе ПО СИ метрологически значимой части, т.е. той его части, которая подлежит испытаниям.

6.2.2. К метрологически значимой части ПО СИ относятся программы, программные модули и алгоритмы, выполняющие функции обработки измерительной информации, по его идентификации и защите, а также части ПО, отнесение которых к метрологически значимым согласовано между участниками испытаний.

6.2.3. После испытаний ПО метрологически значимая часть ПО СИ не должна изменяться. Любая модификация метрологически значимой части ПО СИ приводит к изменению его идентификационных данных (признаков) и к необходимости проведения повторных испытаний, в частности, испытаний с целью утверждения типа СИ или внесению изменений в описание типа СИ в соответствии с [3].

6.2.4. Метрологически не значимая часть ПО СИ испытаниям не подлежит. Ее модификация может быть выполнена без уведомления организаций, проводящих испытания, если изменение этой части не проводит к изменению идентификационных данных (признаков) метрологически значимой части ПО СИ.

6.2.5. Если разделение ПО СИ не проведено, то все ПО рассматривается как метрологически значимое.

6.2.6. Разделение ПО на метрологически значимые и не значимые части может быть проведено как на «низком», так и на «высоком» уровнях.

«Низкий» уровень разделения выполняется независимо от операционной системы внутри кода ПО (на уровне языка программирования). Такой уровень разделения ПО может быть реализован как в СИ со встроенным ПО, так и в СИ с автономным ПО, т.е. в СИ на основе персонального компьютера.

«Высокий» уровень разделения означает, что оно реализуется в виде независимых объектов операционной системы (например, части ПО содержатся в отдельных файлах операционной системы). «Высокий» уровень разделения возможен только в СИ с автономным ПО, т.е в СИ на основе универсального компьютера.

6.2.7. На основе анализа документации и проведения тестирования (функциональных проверок) определяется правильность разделения ПО СИ или устанавливается отсутствие разделения. При этом проверяется, что к метрологически значимой части ПО относятся:

программы, программные модули и алгоритмы, принимающие участие в обработке (расчетах) результатов измерений или влияющие на них;

программы, программные модули и алгоритмы, осуществляющие передачу, идентификацию и обновление (загрузку) ПО, защиту ПО и данных;

параметры ПО СИ, участвующие в вычислениях и влияющие на результат измерений;

компоненты защищенного интерфейса для обмена данными между метрологически значимыми и незначимыми частями ПО СИ.

6.2.8. В случаях, когда проводятся испытания сложных измерительных систем или систем, используемых при коммерческих расчетах, или когда к этим системам предъявляются исключительные требования по безопасности и надежности их функционирования, правильность разделения ПО СИ дополнительно проверяется при помощи анализа его исходного кода.

6.2.9. Сведения о разделении ПО или об его отсутствии заносятся в протокол испытаний.

6.3 Проверка идентификационных данных (признаков) и методов идентификации программного обеспечения

6.3.1. Проверку идентификационных данных (признаков) ПО и методов идентификации проводят при испытаниях ПО, а также при поверке (калибровке) автоматизированных СИ. Согласно рекомендациям Р 50.2.077-2014 идентификация ПО СИ, осуществляемая при поверке (калибровке) СИ, представляет собой проверку соответствия ПО СИ тому, которое было зафиксировано (документировано) в описании типа СИ, с последующим обеспечением защиты ПО от несанкционированного доступа во избежание искажений результатов измерений.

6.3.2. Проверку идентификационных данных (признаков) ПО и методов идентификации проводят при испытаниях ПО СИ на основе анализа документации и проведения тестирования (функциональных проверок). При этом для представителей контролирующих органов и организаций, проводящих испытания, в ряде случаев рекомендуется обеспечить доступ к исполняемому коду ПО СИ.

6.3.3. Доступ к исполняемому коду может быть организован с помощью стандартных интерфейсов связи (RS 232, USB и т.п.) или с помощью иных интерфейсов связи, описанных в документации, в комплекте с необходимым набором аппаратно-программных средств.

6.3.4. На основе анализа документации определяют, какими из следующих способов осуществляется идентификация ПО СИ:

с помощью интерфейса пользователя (например, по команде пользователя на дисплее СИ);

в процессе штатного функционирования ПО (например, на дисплее СИ через определенные интервалы времени);

с помощью интерфейса связи (например, на экране персонального компьютера, подключенного к СИ).

6.3.5. При тестировании (функциональной проверке) способов идентификации ПО СИ убеждаются в том, что они соответствуют тем способам идентификации, которые описаны в документации.

6.3.6. В случае, если идентификация может быть осуществлена несколькими способами, проверяется независимость идентификационных данных (признаков) от способа идентификации.

6.3.7. К идентификационным данным (признакам) относятся следующие данные (их содержание и вид записи может зависеть от типа СИ):

идентификационное наименование программного обеспечения;

номер версии (идентификационный номер) программного обеспечения;

цифровой идентификатор программного обеспечения (контрольные суммы исполняемого кода метрологически значимых частей ПО, рассчитанные по алгоритмам CRC32, md5, SHA1 и т. п. или по специально разработанным алгоритмам с указанием способа их вычисления).

В особых случаях к идентификационным данным (признакам) ПО можно отнести также наименования ПО, наименование разработчика, серийный номер СИ, номер свидетельства или сертификата соответствия и т. д., если эти данные непосредственно связаны с ПО.

Идентификационные данные (признаки) должны иметь структуру, однозначно связанную с метрологически значимой частью программного обеспечения.

Допускается представление номера версии ПО в форме записи «номер версии не ниже ...» или замены некоторых элементов в его обозначении, отвечающих за метрологически незначимую часть, специальными символами (например, «X» или «-»). Например, версия программного обеспечения «Система фотограмметрическая однокамерная СФО» v.3.5.7.2 может быть представлена в виде v.3.5.x.x, при этом только часть версии ПО «v.3.5» отвечает за метро-

логически значимую часть, а запись «версия ПО не ниже v.3.5.7.2» может означать, что допускаются версии v.3.5.7.2, v.3.5.7.3 , v.3.5.8.1 и т. д.

6.3.8. На основе анализа документации и проведения тестирования (функциональных проверок) определяют реализованные в ПО СИ методы идентификации ПО. Идентификация ПО СИ может быть реализована следующими методами:

с помощью ПО СИ или аппаратно-программных средств, разработанных организацией – производителем СИ (ПО СИ);

с использованием специальных утвержденных аппаратно-программных средств и/или с помощью утвержденного ПО.

6.3.9. Проверяются наличие и достаточность идентификационных данных (признаков) ПО СИ для его однозначной идентификации.

6.3.10. Проверяется, что расчет контрольной суммы производится для метрологически значимой части ПО СИ. При этом реализованный в ПО СИ алгоритм расчета контрольной суммы также относится к метрологически значимой части ПО СИ.

6.3.11. В случае, когда идентификация ПО СИ осуществляется с использованием специальных утвержденных аппаратно-программных средств и/или утвержденного ПО, проверку контрольной суммы метрологически значимой части ПО СИ осуществляет организация, проводящая испытания.

6.3.12. Организация – разработчик ПО СИ вправе использовать для идентификации ПО большее количество идентификационных данных (признаков), чем это указано в п. 6.3.7 настоящей Рекомендации. В этом случае проверяется, что структура идентификационных данных (признаков) ПО позволяет однозначно выделить идентификационные данные (признаки), относящиеся к метрологически значимой части ПО.

6.3.13. В случаях, когда проводятся испытания сложных измерительных систем или систем, используемых при коммерческих расчетах, или, когда к этим системам предъявляются исключительные требования по безопасности и

надежности их функционирования, дополнительно проводят проверку методов и способов идентификации ПО СИ при помощи анализа его исходного кода.

6.3.14. Сведения об идентификационных данных (признаках) ПО СИ и методах его идентификации вносят в протокол испытаний.

6.4 Проверка структуры программного обеспечения

6.4.1. Под проверкой структуры ПО понимают:

проверку отсутствия недопустимого влияния на метрологически значимую часть ПО и данные, осуществляемого через интерфейс пользователя;

проверку отсутствия недопустимого влияния на метрологически значимую часть ПО и данные, осуществляемого через интерфейсы связи;

проверку правильности взаимодействия между метрологически значимой и незначимой частями ПО.

6.4.2. Проверка отсутствия недопустимого влияния на метрологически значимую часть ПО и данные, осуществляемого через интерфейс пользователя.

6.4.2.1. Анализом документации на ПО убеждаются в том, что она включает в себя:

полный перечень всех команд интерфейса пользователя вместе с изложением способа их организации (меню, пункты меню и т.д.);

описание их назначения и воздействия на функции СИ и/или данные.

6.4.2.2. Проведением тестирования (функциональных проверок) всех команд интерфейса пользователя ПО СИ убеждаются в их соответствии описанным в документации. Проверяется однозначное назначение каждой команды для инициирования функции или изменения данных в соответствии с представленной документацией.

6.4.2.3. С помощью тестирования (функциональных проверок) убеждаются, что команды и данные, введенные через интерфейс пользователя

ПО СИ, не оказывают влияние на достоверность результатов измерений. При этом проверяют:

возможность обнаружения программным обеспечением СИ неправильно введенных через интерфейс пользователя данных (например, данных, превышающих установленные ограничения) и выдачу соответствующего предупреждения;

невозможность изменения значений параметров ПО СИ, участвующих в вычислениях и влияющих на результат измерений, с помощью команд и данных, вводимых через интерфейс пользователя во время проведения измерений;

невозможность искажения значений измеренных данных, хранящихся в памяти СИ, с помощью команд и данных, вводимых через интерфейс пользователя.

6.4.2.4. С учетом способа организации интерфейса пользователя проверяют, что команды или их комбинации, не описанные в документации, не оказывают влияния на функции метрологически значимой части ПО СИ и данные.

6.4.2.5. С помощью визуального осмотра и анализа элементов, находящихся внутри корпуса СИ, убеждаются в отсутствии устройств, не описанных в документации на СИ, способных быть частью интерфейса пользователя и оказывать влияние на функции метрологически значимой части ПО СИ, данные или команды интерфейса пользователя (переключатели, свободные контакты на печатной плате и т.д.).

6.4.2.6. В случаях, когда проводятся испытания сложных измерительных систем или систем, используемых при коммерческих расчетах, или, когда к этим системам предъявляются исключительные требования по безопасности и надежности их функционирования, проверку отсутствия недопустимого влияния на метрологически значимую часть ПО и данные, осуществляемого

через интерфейс пользователя ПО СИ, дополнительно проводят при помощи анализа его исходного кода.

6.4.3. Проверка отсутствия недопустимого влияния на метрологически значимую часть ПО и данные, осуществляемого через интерфейсы связи.

6.4.3.1. Анализом документации на ПО убеждаются в том, что она включает в себя:

полный перечень всех интерфейсов связи, используемых ПО СИ (RS-232, USB и т.п.);

полный перечень команд каждого интерфейса связи вместе с изложением способа их организации;

описание их назначения и воздействия на функции СИ и/или данные.

6.4.3.2. Проведением тестирования (функциональных проверок) команд интерфейсов связи, используемых ПО СИ, убеждаются в их соответствии описанным в документации. Проверяется однозначное назначение каждой команды для инициирования функции или изменения данных в соответствии с представленной документацией.

6.4.3.3. С помощью тестирования (функциональных проверок) убеждаются, что команды и данные, переданные через интерфейсы связи, не оказывают влияние на достоверность результатов измерений СИ. При этом проверяют:

возможность обнаружения программным обеспечением СИ неправильно переданных через интерфейсы связи данных (например, данных, превышающих установленные ограничения);

невозможность изменения значений параметров ПО СИ, участвующих в вычислениях и влияющих на результат измерений, с помощью команд и данных, переданных через интерфейсы связи во время проведения измерений;

невозможность искажения значений измеренных данных, хранящихся в памяти СИ, с помощью команд и данных, переданных через интерфейсы связи.

6.4.3.4. Проверяют, что недокументированные как команды сигналы или коды, переданные через интерфейсы связи, не оказывают влияние на функции метрологически значимой части ПО СИ и данные.

6.4.3.5. Проверяют, что команды, передаваемые (получаемые) через интерфейсы связи метрологически незначимой частью ПО СИ, не искажают команды и данные, передаваемые (получаемые) через интерфейсы связи метрологически значимой частью ПО СИ.

6.4.3.6. В случае, когда в ПО СИ используется часть интерфейсов связи СИ (например, в случае СИ с автономным ПО, т.е. на основе универсального компьютера), проверяют, что сигналы или коды, переданные через неиспользуемые интерфейсы связи, не оказывают влияние на функции метрологически значимой части ПО СИ и данные.

6.4.3.7. Проверяют, что ПО, использующее интерфейс связи СИ для передачи (получения) команд и данных метрологически значимой части ПО СИ (например, ПО, разработанное организацией-разработчиком (производителем) СИ и используемое для обновления ПО), прошло подтверждение соответствия в установленном порядке.

6.4.3.8. С помощью визуального осмотра и анализа элементов, находящихся внутри корпуса СИ, убеждаются в отсутствии устройств, не описанных в документации на СИ, способных быть частью интерфейсов связи и оказывать влияние на функции метрологически значимой части ПО СИ, данные или команды интерфейсов связи.

6.4.3.9. В случаях, когда проводятся испытания сложных измерительных систем или систем, используемых при коммерческих расчетах, или, когда к этим системам предъявляются исключительные требования по безопасности и надежности их функционирования, проверку отсутствия недопустимого

влияния на метрологически значимую часть ПО и данные, осуществляемого через интерфейсы связи, дополнительно проводят при помощи анализа его исходного кода.

6.4.4. Проверка правильности взаимодействия между метрологически значимой и незначимой частями ПО.

6.4.4.1. Тестированием (функциональными проверками) убеждаются, что обмен данными между метрологически значимой и незначимой частями ПО СИ проходит через защищенный интерфейс. Проверяется однозначное назначение каждого набора команд, переданного через защищенный интерфейс, для инициирования функции или изменения данных в метрологически значимой части ПО СИ в соответствии с представленной документацией.

6.4.4.2. Проверяют, что все взаимодействия между метрологически значимой и незначимой частями ПО СИ и прохождение данных не оказывают искажающее воздействие на метрологически значимую часть ПО и данные.

6.4.4.3. Убеждаются, что взаимодействия между метрологически значимой и незначимой частями ПО СИ, не описанные в документации, не оказывают влияния на метрологически значимую часть ПО СИ и данные.

6.4.4.4. В случаях, когда проводятся испытания сложных измерительных систем или систем, используемых при коммерческих расчетах, или, когда к этим системам предъявляются исключительные требования по безопасности и надежности их функционирования, проверку правильности взаимодействия между метрологически значимой и незначимой частями ПО, дополнительно проводят при помощи анализа его исходного кода.

6.4.5. Сведения об отсутствии недопустимого влияния на метрологически значимую часть ПО и данные, осуществляемого через интерфейс пользователя, об отсутствии недопустимого влияния на метрологически значимую часть ПО и данные, осуществляемого через интерфейсы связи, о правильности взаимодей-

ствия между метрологически значимой и незначимой частями ПО вносят в протокол испытаний.

6.5 Оценка влияния программного обеспечения и его алгоритмов на метрологические характеристики средств измерений

6.5.1. Оценка влияния ПО и его алгоритмов на МХ СИ определяется методикой испытаний и может включать в себя:

анализ ПО и его алгоритмов (например, адекватность измерительной задаче, их сложность и возможность использования при разработке опорного ПО и т.д.);

определение критерия оценки влияния ПО на метрологические характеристики СИ (например, значение вклада ПО в суммарную погрешность (неопределенность) СИ, значение относительного отличия тестовых результатов вычислений от опорных и т.п.);

выбор (или разработка) опорного ПО;

выбор (определение) исходных данных и/или их получение методом генерации или какими-либо другими методами;

получение результатов обработки исходных данных в тестируемом ПО (получение тестовых результатов);

получение опорных результатов;

получение оценки влияния ПО на метрологические характеристики СИ посредством обработки результатов тестирования (сравнения тестовых результатов с опорными);

дополнительные исследования свойств, параметров и характеристик используемых алгоритмов (область устойчивости, время, затрачиваемое на обработку результатов измерений и т.п.).

6.5.2. Основными методами, применяемыми при оценке влияния ПО на МХ СИ, являются:

испытания с применением опорного ПО;

в отсутствие опорного ПО – испытания с использованием моделей исходных данных [4,7], либо с применением метода генерации «эталонных» данных [5, 6];

при наличии нескольких ПО сопоставимого уровня вычислительных возможностей и в отсутствие опорного ПО – сличения, подобные сличению эталонных СИ;

испытания с применением метода перекрестной проверки (кросс-валидации);

испытания на основе анализа исходного кода ПО, а также комбинации указанных методов.

Метод оценки влияния ПО на МХ СИ выбирают с учетом наличия или возможности разработки того или иного вида опорного ПО, а также возможности применения указанных методов в каждом конкретном случае.

6.5.3. Испытания с применением опорного («эталонного») ПО.

6.5.3.1. Данный метод испытаний применяется при наличии опорного («эталонного») ПО, с помощью которого могут быть идентично воспроизведены функции тестируемого ПО.

6.5.3.2. В качестве опорного («эталонного») ПО может быть применено:

ПО СИ, прошедшее испытания (утвержденное ПО), функциональное назначение которого аналогично тестируемому ПО;

специально разработанное ПО с функциями, идентичными тестируемому;

ПО для решения задач технических вычислений (например, электронные таблицы, ПО для математических и статистических вычислений и т.д.).

6.5.3.3. К разработке опорного ПО прибегают в тех случаях, когда испытываемое ПО является не очень сложным, а его алгоритмы достаточно просты. Это означает, что затраты на разработку опорного ПО должны быть сопоставимыми со стоимостью работ по испытаниям ПО. Данный метод

позволяет максимально учитывать особенности тестируемого ПО, а также МХ соответствующего СИ, и может быть рекомендован как основной метод при испытаниях встроенного ПО.

6.5.3.4. Разрабатываемое опорное ПО может содержать только метрологически значимые функции и параметры. В некоторых случаях могут не учитываться особенности графического интерфейса пользователя, а также функции, не участвующие в обработке результатов измерений (например, функции отображения, хранения данных и т.д.).

6.5.4. Испытания с использованием моделей исходных данных.

6.5.4.1. Метод испытаний с использованием моделей исходных данных рекомендуется методикой МИ 2174 [4] для тестирования алгоритмов обработки результатов измерений. Метод позволяет оценивать возможности тестируемых алгоритмов сравнением результатов обработки ими моделей исходных данных с самими моделями, параметры которых заданы (известны).

6.5.4.2. Метод моделей исходных данных является разновидностью метода генерации «эталонных» данных, когда эти данные не генерируются специально разработанной программой, а программно задаются на входе испытываемого ПО. Модели исходных данных выбираются таким образом, чтобы они максимально соответствовали частной измерительной задаче, решаемой тестируемыми алгоритмами.

6.5.4.3. В модели исходных данных могут быть включены:

данные, указанные в разделе 4 методики МИ 2174;

данные, полностью перекрывающие диапазон возможных значений;

данные, близкие к наибольшим и наименьшим значениям, а также ряд промежуточных значений;

особые значения входных переменных - точки резкого возрастания или разрыва производных, нулевые, единичные и предельно малые численные значения переменных и т.п.

6.5.4.4. Если значения некоторой переменной зависят от значения другой переменной, то испытания проводят при особых сочетаниях этих переменных, таких, как равенство обеих переменных, малое и предельно большое их различие, нулевые и единичные значения и т.п.

6.5.5. Генерация опорных наборов данных.

6.5.5.1. Метод генерации опорных наборов данных, как и метод моделей исходных данных, применяется как альтернатива использованию опорного ПО в случае его отсутствия или невозможности использования при оценке отдельных функций, реализуемых испытываемым ПО. Необходимым условием применения метода генерации опорных данных является наличие априорной информации о модельном решении соответствующей измерительной задачи. С этим модельным решением проводится сравнение тестовых результатов.

6.5.5.2. Опорные данные получают путем генерации таких данных с помощью специально разработанной программы – генератора опорных данных, который представляет собой алгоритм, предназначенный для моделирования опорных данных на основе выбранных (заданных) исходных данных.

Генератор опорных данных реализуют на одном из языков программирования или при помощи стандартного математического или статистического программного пакета [5,6].

6.5.5.3. Исходные данные для тестирования, в том числе и для генерации опорных данных, формируются с учетом свойств программно реализованных алгоритмов.

6.5.6. Сличение ПО.

6.5.6.1. При наличии нескольких программ сопоставимого уровня вычислительных возможностей и в отсутствие опорного ПО рекомендуется проводить сличение таких программ, когда на их входы подаются согласованные одинаковые наборы опорных данных и производится сравнение

соответствующих результатов испытаний. При этом результаты сличения признаются удовлетворительными, если различия в результатах испытаний не выходят за пределы согласованного допуска.

Примечание - Примером программ, указанных в п. 6.5.6.1, являются программы расчета параметров расходомеров на основе стандартных сужающих устройств [8, 9]. Это сложные программы, основанные в ряде случаев на громоздких формулах и математических соотношениях и использующие эмпирические данные о свойствах проходящих через расходомеры сред, которые в разных программах выбираются с разной точностью, либо вычисляются с помощью различных интерполяционных процедур и т.п. В этих условиях выбрать среди этих программ или разработать опорную программу не представляется возможным. В виду сложности таких программ не удается также применить методы моделей исходных данных или генерации опорных данных.

6.5.7. Метод перекрестной проверки (кросс-валидации)

Метод кросс-валидации в совокупности с количественным критерием оценки качества ПО п. 6.5.9.3. применяется для задач оценки качества подбора значений модельно-зависимых параметров при применении моделей интерполяции.

Метод основан на проведении оценки для части данных, выбранных из основного набора, по остальным данным с последующим вычислением ошибки оценки. После оценок по всем наборам или выборкам оценивается среднее значение полученных оценок. По нему сравниваются различные методы или выбираются наилучшие параметры модели. Процедура перекрестной проверки (кросс-валидации) сводится к следующему:

1. Исходная выборка разбивается N различными способами (Рисунок 1) на две непересекающиеся подвыборки $X^L = X_n^m \cup X_n^k$, где:

X_n^m - обучающая подвыборка длины m,

X_n^k - контрольная подвыборка длины $k = L- m$, $n=1,\ldots, n$ - номер разбиения.

2. Для каждого разбиения n строится алгоритм

$$a_n = f(c_i, X_n^m),$$
$$b_n = f(c_j, X_n^k),$$

где c_i и c_j наборы параметров модели,

3. Вычисляется значение функционала качества $Q_n = Q(a_n, b_n)$.

В качестве количественного критерия оценки качества используется относительное расхождение между параметрами модели, описывающими на n-ом разбиении обучающую и контрольную подвыборки.

	Блок 1	Блок 2	…	…	Блок N-1	Блок N	
	Обучающая подвыборка					Контрольная подвыборка	Точность
Шаг 1	Блок 1	Блок 2	…	…	Блок N-1	Блок N	Q_1
Шаг 2	Блок 1	Блок 2	…	…	Блок N-1	Блок N	Q_2
	.	.	.	.	.	.	
	.	.	.	.	.	.	
Шаг K-1	Блок 1	Блок 2	…	…	Блок N-1	Блок N	Q_{N-1}
Шаг K	Блок 1	Блок 2	…	…	Блок N-1	Блок N	Q_N
Итоговая оценка точности $\frac{1}{N} \cdot \sum_{i=1}^{N} Q_N$							

6.5.8. Тестирование алгоритмов на основе анализа исходного кода ПО.

6.5.8.1. При тестировании алгоритмов на основе анализа исходного кода ПО проверяется:

соответствие структуры алгоритмов представленной документации;

правильность записи алгоритмов на выбранном языке программирования;

адекватность выбранных алгоритмов измерительной задаче (в частности, выявление неустойчивых алгоритмов).

6.5.8.2. При проверке соответствия структуры алгоритмов представленной документации, по тексту программы могут быть составлены блок-схемы реализуемых алгоритмов, которые сравниваются с алгоритмами, изложенными в документации. В случае нахождения различий в структуре алгоритмов проводится дополнительный анализ элементов блок-схем, в которых обнаружены различия.

6.5.8.3. Проверяется правильность записи алгоритмов на выбранном языке программирования. При этом устанавливается соответствие кода правилам программирования, наличие неопределенных переменных и операторов, правильность организации циклов и т.д.

6.5.8.4. Соответствие выбранных алгоритмов измерительной задаче может быть осуществлено путем математического анализа программно реализованных алгоритмов. При этом могут исследоваться логические и точностные характеристики реализованных алгоритмов, в частности, анализироваться пригодность и оптимальность примененных численных методов решения измерительной задачи.

6.5.9. Представление результатов оценки влияния ПО и его алгоритмов на МХ СИ.

6.5.9.1. На основе используемых методов оценки влияния ПО на МХ СИ, перечисленных в предыдущих пунктах настоящей рекомендации, рассчитывают характеристики вычислительной точности алгоритмов, осуществляющих расчеты при обработке измерительной информации, например, его исполнительную характеристику (performance measure) [5] или относительное отличие результатов вычислений от опорных [6].

Могут быть оценены и другие характеристики алгоритмов такие, как их сложность, скорость исполнения, адекватность измерительной задаче, выбор численной схемы расчета, коэффициент обусловленности (устойчивости), область устойчивости и т.п.

6.5.9.2. Исполнительная характеристика алгоритма.

Исполнительная характеристика алгоритма вычисляется по формуле [5]

$$P(\vec{x}) = \lg(1 + \frac{1}{k(\vec{x})\eta} \cdot \frac{\|\Delta\vec{y}\|}{\|\vec{y}^{(эт)}\|}), \qquad (1)$$

где $k(x)$ - коэффициент обусловленности (устойчивости) (для устойчивых алгоритмов $k(x) \approx 1$), η - машинная относительная предельная точность вычислений ($\eta \approx 10^{-16}$), $\|\Delta\vec{y}\|$ - норма (длина) вектора отличия тестовых результатов от опорных, $\|\vec{y}^{(эт)}\|$ - норма опорных результатов.

Например, если в процессе вычислений получено m тестовых результатов $y_1^{(тест)}, y_2^{(тест)}, ..., y_m^{(тест)}$ и опорных $y_1^{(эт)}, y_2^{(эт)}, ..., y_m^{(эт)}$, то норма $\|\Delta\vec{y}\|$ вычисляется по формуле

$$\|\Delta\vec{y}\| = \sqrt{(y_1^{(тест)} - y_1^{(эт)})^2 + (y_2^{(тест)} - y_2^{(эт)})^2 + ... + (y_m^{(тест)} - y_m^{(эт)})^2},$$

норма опорных результатов – по формуле

$$\|\vec{y}^{(эт)}\| = \sqrt{y_1^{(эт)2} + y_2^{(эт)2} + ... + y_m^{(эт)2}}.$$

Исполнительная характеристика показывает число потерянных цифр точности в тестируемом ПО по сравнению с опорным [5].

6.5.9.3. Исполнительная характеристика, определенная формулой (1), зависит, в частности, от величины

$$\delta = \frac{\|\Delta\vec{y}\|}{\|\vec{y}^{(эт)}\|}, \qquad (2)$$

которая характеризует относительное отличие результатов вычислений от опорных. Эта величина может рассматриваться как одна из количественных характеристик алгоритмов. Иногда ее удобно выражать в процентах.

Для единичного результата вычислений ($m=1$) формула (2) упрощается и принимает вид

$$\delta = \frac{\left| y^{(тест)} - y^{(эт)} \right|}{\left| y^{(эт)} \right|} \cdot 100\%. \qquad (3)$$

Примечание - Исполнительная характеристика (1) может применяться также для нахождения числа потерянных цифр точности в результатах испытаний по сравнению с любыми другими результатами, используемыми для сравнения с ними (модельными, сгенерированными и т.п.). Это примечание относится также и к величине, определяемой формулами (2) и (3).

6.5.9.4. Критерии, которым должны удовлетворять определенные и оцененные характеристики алгоритмов ПО, а также допускаемые значения характеристик могут быть установлены на основе требований к точности решения измерительной задачи (если они имеются), точности выполняемых расчетов (степени округления) и т.п. Критерии и допуски на значения характеристик фиксируются в методике испытаний и согласовываются с ее заказчиком.

6.5.9.5. Все определенные и оцененные характеристики и свойства алгоритмов вносят в протокол испытаний.

6.5.9.6. Перечень характеристик испытываемого ПО может корректироваться соглашением между организацией, проводящей испытания, и заказчиком сертификации.

6.6. Проверка защиты программного обеспечения и определение ее уровня

6.6.1. Проверку защиты ПО СИ и его алгоритмов проводят с целью установления наличия средств защиты метрологически значимой части ПО и измеренных данных и определения уровня защиты ПО от непреднамеренных и преднамеренных изменений. Под проверкой защиты программного обеспечения понимается:

проверка защиты метрологически значимой части ПО и измеренных данных от случайных или непреднамеренных изменений;

проверка защиты метрологически значимой части ПО и измеренных данных от преднамеренных изменений.

6.6.2. Проверка защиты метрологически значимой части ПО и измеренных данных от случайных или непреднамеренных изменений.

6.6.2.1. Возможными причинами случайных или непреднамеренных изменений метрологически значимой части ПО и измеренных данных могут быть:

непредсказуемые физические воздействия;

эффекты, обусловленные действиями пользователя.

6.6.2.2. На основе анализа документации определяется наличие (отсутствие) средств защиты метрологически значимой части ПО и измеренных данных от изменения или удаления в случае возникновения непредсказуемых физических воздействий (например, наличие энергонезависимой памяти для хранения измеренных данных).

6.6.2.3. С помощью функциональных проверок, имитирующих непредсказуемые физические воздействия, убеждаются в действии средств защиты метрологически значимой части ПО и измеренных данных от изменения или удаления в случае возникновения непредсказуемых физических воздействий.

6.6.2.4. На основе анализа документации и проведения функциональных проверок убеждаются, что интерфейс пользователя ПО содержит в себе средства предупреждения пользователя, если его действия могут повлечь изменение или удаление метрологически значимой части ПО и/или измеренных данных.

6.6.2.5. На основе анализа документации и проведения функциональных проверок, имитирующих различного рода ошибки или иные изменения случайного или непреднамеренного характера, проверяется их обнаружение и фиксация в журнале(ах) событий.

6.6.3. Проверка защиты метрологически значимой части ПО и измеренных данных от преднамеренных изменений.

6.6.3.1. Метрологически значимая часть ПО в необходимых случаях должна содержать специальные средства защиты, исключающие возможность несанкционированной модификации, загрузки (в том числе загрузки фальсифицированного ПО и данных), считывания из памяти СИ, удаления или иных преднамеренных изменений метрологически значимой части ПО и измеренных данных. К специальным средствам защиты метрологически значимой части ПО и измеренных данных от преднамеренных изменений могут быть, в частности, отнесены:

средства проверки целостности ПО (например, несанкционированная модификация метрологически значимой части ПО может быть проверена расчетом контрольной суммы для метрологически значимой части ПО и сравнением ее с действительным значением);

средства обнаружения и фиксации событий;

средства управления доступом;

иные средства защиты.

6.6.3.2. На основе анализа документации и проведения тестирования (функциональных проверок) убеждаются, что действие средства проверки

целостности ПО распространяется на метрологически значимую часть ПО и данные. Для этой цели вносят изменения в метрологически значимую часть ПО и измеренные данные и проверяют реакцию средства проверки целостности ПО на внесенные изменения.

6.6.3.3. На основе анализа документации проверяется соответствие алгоритма проверки целостности ПО достаточному уровню защиты метрологически значимой части ПО и измеренных данных от преднамеренных изменений.

6.6.3.4. Если для проверки целостности ПО применяют расчет контрольной суммы, то на основе документации проверяется, что алгоритм, используемый для расчета контрольной суммы, и количество символов контрольной суммы соответствуют достаточному уровню защиты метрологически значимого ПО и измеренных данных от преднамеренных изменений.

6.6.3.5. На основе анализа документации определяется набор событий, который подлежит обнаружению и фиксации в соответствующем журнале событий. Убеждаются, что в набор событий, подлежащий обнаружению и фиксации, включены события, связанные с обновлением (загрузкой) метрологически значимой части ПО, изменением или удалением измеренных данных в памяти СИ, изменением параметров ПО, участвующих в вычислениях и влияющих на результат измерений.

6.6.3.6. Проведением тестирования (функциональных проверок), имитирующих наступление событий, подлежащих обнаружению и фиксации в журнале событий ПО, проверяют соответствующую реакцию средства обнаружения и фиксации событий.

6.6.3.7. Проверяют, что данные журнала событий невозможно исказить либо несанкционированно удалить без нарушения защиты иных средств защиты

метрологически значимой части ПО и измеренных данных от преднамеренных изменений.

6.6.3.8. На основе анализа документации определяются полномочия пользователей, имеющих различные права доступа к функциям метрологически значимой части ПО и измеренным данным.

6.6.3.9. С помощью функциональных проверок убеждаются в соответствии полномочий пользователей, имеющих различные права доступа к функциям метрологически значимой части ПО и измеренным данным, полномочиям, описанным в документации на ПО.

6.6.3.10. Проверяют корректность реализации управления доступом пользователя к функциям метрологически значимой части ПО и измеренным данным. Для этого проверяется реакция средства управления доступом на неоднократный ввод неправильных идентификационных данных пользователя. Формат идентификационных данных пользователя, используемых для доступа к функциям метрологически значимой части ПО и измеренным данным, должен соответствовать достаточному уровню защиты метрологически значимой части ПО и измеренных данных от преднамеренных изменений.

6.6.3.11. Для СИ, в которых отсутствуют интерфейсы связи, несанкционированная модификация, удаление или иные преднамеренные изменения метрологически значимой части ПО и измеренных данных возможны посредством замены запоминающего устройства(в) другим, содержащим фальсифицированную метрологически значимую часть ПО и измеренные данные. Проверяется, что конструкцией СИ, непосредственно запоминающим устройством(ми) или иным способом, обеспечивается защита запоминающего устройства(в) от несанкционированной замены.

6.6.4. В случаях, когда проводятся испытания сложных измерительных систем или систем, используемых при коммерческих расчетах, или, когда к этим системам предъявляются исключительные требования по безопасности и надежности их функционирования, защита метрологически значимой части ПО

и измеренных данных от преднамеренных, случайных или непреднамеренных изменений дополнительно проверяется при помощи анализа исходного кода ПО.

6.6.5. Определение уровня защиты ПО СИ от непреднамеренных и преднамеренных изменений.

6.6.5.1. Определение уровня защиты ПО от непреднамеренных и преднамеренных изменений проводят в случае проверки декларации, указанной в п. 6.1.6 настоящей методики, на основании результатов исследований ПО, выполненных в соответствии с разделами 6.2, 6.3, 6.4 и 6.6 настоящей методики.

6.6.5.2. При определении уровня защиты ПО от непреднамеренных и преднамеренных изменений учитывают необходимость применения и достаточность примененных специальных средств защиты метрологически значимой части ПО и измеренных данных от преднамеренных изменений (см. табл.1).

6.6.5.3. Уровню «низкий» защиты ПО соответствует такой уровень защиты метрологически значимой части ПО и измеренных данных, при котором не требуется специальных средств защиты, исключающих возможность несанкционированной модификации, обновления (загрузки), удаления и иных преднамеренных изменений метрологически значимой части ПО и измеренных данных.

6.6.5.4. Уровню «средний» защиты ПО соответствует такой уровень защиты метрологически значимой части ПО СИ и измеренных данных, при котором программное обеспечение защищено от преднамеренных изменений с помощью простых программных средств (например, текстовых редакторов). Примерами защиты могут служить: пароли, авторизация пользователя и т. п.;

6.6.5.5. Уровню «высокий» защиты ПО соответствует такой уровень защиты метрологически значимой части ПО и измеренных данных, при котором примененные специальные средства защиты в достаточной мере исключают

возможность несанкционированной модификации, обновления (загрузки), удаления и иных преднамеренных изменений метрологически значимой части ПО и измеренных данных.

Содержание п.п. 6.6.5.3 – 6.6.5.5 представлено в таблице 1.

Таблица 1

Уровень защиты ПО СИ от непреднамеренных и преднамеренных изменений	Описание
Низкий	Не используются никакие специальные средства защиты от преднамеренных изменений
Средний	Метрологически значимая часть ПО и измеренные данные защищены от преднамеренных изменений с помощью простых программных средств. Например, пароли, авторизация пользователя и т. п.
Высокий	Метрологически значимая часть ПО СИ и измеренные данные достаточно защищены с помощью специальных средств защиты от преднамеренных изменений. Например, криптографические методы защиты, электронное и механическое опечатывание и т. д.

6.6.5.6. Для каждого уровня защиты ПО от непреднамеренных и преднамеренных изменений должна быть в достаточной степени обеспечена защита метрологически значимой части ПО и измеренных данных от случайных или непреднамеренных изменений.

6.6.5.7. Сведения о защите метрологически значимой части ПО и измеренных данных от случайных или непреднамеренных изменений, о защите метрологически значимой части ПО и измеренных данных от преднамеренных изменений, об уровне защиты ПО от непреднамеренных и преднамеренных изменений вносят в протокол испытаний.

6.7. Проверка программного обеспечения при использовании в нем информационных технологий

6.7.1. Проверку ПО СИ и его алгоритмов при использовании в нем информационных технологий проводят с целью обеспечения функционирования этих технологий в соответствии с технической документацией, подлинности, целостности и необходимого уровня защиты ПО и данных от непреднамеренных и преднамеренных измерений.

Под проверкой ПО СИ при использовании в нем информационных технологий понимается:

проверка обновления (загрузки) ПО;

проверка долговременного хранения данных и их передачи через сети коммуникации;

проверка разделения ПО на метрологически значимую и незначимую части.

6.7.2. Проверка обновления (загрузки) ПО СИ.

6.7.2.1. Проверку обновления (загрузки) ПО, проводят при обнаружении и исправлении ошибок в ПО, при расширении/модификации его функций, при обновлении служебных программных модулей (драйверов, библиотек и т.п.).

6.7.2.2. Анализом документации и функциональными проверками убеждаются, что программное обеспечение было загружено с разрешения пользователя или владельца средства измерений. Если для распознавания пользователя или владельца СИ используется электронная подпись, она должна сохраняться в установленной части ПО СИ. Подлинность подписи, присоединенной к ПО, должна устанавливаться автоматически.

6.7.2.2. Анализом документации и функциональными проверками убеждаются, что программное обеспечение было загружено с разрешения пользователя или владельца средства измерений.

6.7.2.3. Тестированием (функциональными проверками) убеждаются, что загрузка и последующая инсталляция программного обеспечения осуществляется автоматически и что загрузка метрологически значимой части ПО осуществляется в защищенную область ПО, при этом автоматически осуществляется проверка целостности и подлинности ПО и уничтожаются излишние файлы.

6.7.2.3. Функциональными проверками убеждаются, что загрузка метрологически значимой части ПО СИ осуществляется в защищенную область ПО и при этом выполняется автоматическая проверка его целостности и подлинности.

6.7.2.4. Анализом документации и функциональными проверками убеждаются, что ПО СИ, предусматривающее возможность обновления, содержит средства проверки подлинности загружаемого ПО. ПО может быть загружено только после успешной проверки его подлинности. Подлинность загружаемого ПО проверяется перед началом загрузки. В случае если проверка подлинности показала отрицательный результат, ПО прерывает загрузку с записью данных о попытке загрузки.

6.7.2.5. Проверка подлинности ПО и его принадлежности к СИ, прошедшему испытания с целью утверждения типа, осуществляется автоматической идентификацией метрологически значимой части. При этом генерируется контрольная сумма (электронная подпись) для загружаемой части ПО. Подлинность гарантируется, если контрольная сумма, сохраняемая в установленной части ПО, совпадает с контрольной суммой, зафиксированной при подтверждении соответствия ПО. Установление этого соответствия должно производиться автоматически.

6.7.2.6. Анализом документации и функциональными проверками убеждаются, что ПО СИ, предусматривающее возможность обновления, содержит средства проверки целостности загружаемого ПО, т.е. проверки того, что оно не было изменено в процессе загрузки. Проверка целостности загружаемого ПО обеспечивается, например, добавлением хеш-суммы (контрольной суммы) за-

гружаемого обновления с проверкой значения контрольной суммы в начале и по окончании загрузки.

6.7.2.7. Целостность ПО проверяется вычислением контрольной суммы для метрологически значимой части ПО и сравнением ее с контрольной суммой, сохраненной в ПО после подтверждения его соответствия.

6.7.2.8. Если для облегчения загрузки было произведено снижение уровня защиты, то убеждаются, что после завершения загрузки уровень защиты автоматически восстанавливается до исходного уровня.

6.7.2.9. Тестированием (функциональными проверками) убеждаются, что процесс загрузки не влияет на функционирование метрологически значимых функций ПО СИ, при этом техническими средствами обеспечивается корректная работа ПО в процессе загрузки или приостановка его действия на период загрузки.

6.7.2.10. Вся информация об обновлении ПО фиксируется. Запись об обновлении создается при каждом его инициировании вне зависимости от его результата, а также вне зависимости от загружаемой части, т.е. вне зависимости от того, относится ли обновляемая часть к метрологически значимому ПО или не относится. К данным об обновлении, как минимум, относятся: дата загрузки, результат загрузки (успешно/ошибка, включая код ошибки), прежнее и новое значение идентификации, данные об источнике загрузки.

6.7.3. Проверка долговременного хранения данных и их передачи через сети коммуникации.

6.7.3.1. Проверку долговременного хранения данных проводят в случаях, когда ПО СИ использует данные, полученные вне места проведения измерений, и когда возможны передача и хранение таких данных в незащищенной среде.

6.7.3.2. Анализом документации убеждаются в достоверности сохраняемых данных, при этом проверяется, что сохраняемые данные содержат необхо-

димую информацию об измерениях, в процессе которых они были получены, т.е. проверяется, что данные об измерениях, как минимум, содержат:

измеренные значения, включая единицы измерения,

время измерения,

идентификационные данные (признаки) СИ или ПО, которое было использовано для получения этих данных.

6.7.3.4. Анализом документации и проведением функциональных проверок убеждаются, что сохраняемые данные защищены с помощью средств, обеспечивающих их подлинность и целостность, при этом использование данных, полученных вне места проведения измерений, возможно только после успешной проверки и подтверждения их подлинности и целостности.

6.7.3.5. Для проверки отсутствия изменения данных вследствие физических эффектов вычисляется контрольная сумма по всему вводимому массиву данных, которая сравнивается с контрольной суммой, сохраняемой в проверяемом массиве.

6.7.3.6. Тестированием (функциональными проверками) убеждаются, что сохраняемые данные измерений не могут быть удалены без предварительного разрешения, при этом должно выводиться диалоговое сообщение или «окно», задающее вопрос о подтверждении удаления.

6.7.3.7. Целостность данных проверяется повторным вычислением контрольной суммы по сохраняемому массиву данных и сравнением с сохраняемым номинальным значением перед повторным использованием данных. Если значения контрольных сумм совпадают, массив данных принимается и может быть использован, в противном случае он должен быть удален или помечен как неверный.

6.7.3.8. Для проверки корректности процедуры восстановления сохраняемых данных в случае, если возникают сомнения в их достоверности после восстановления, массив данных считывается из устройства хранения или передачи программой, прошедшей подтверждение соответствия. Затем снова вычисляется

контрольная сумма по всему массиву данных, которая сравнивается с сохраняемым номинальным значением. Если оба значения совпадают, данные признаются корректными, в противном случае данные не используются и удаляются или помечаются программой как неверные.

6.7.3.9. Тестированием (функциональными проверками) проверяют, что:

каждый набор передаваемых данных имеет единственный идентификационный номер, который может содержать информацию о времени, когда измерение было выполнено (отметку времени);

каждый набор передаваемых данных содержит информацию о происхождении данных измерения, т.е. регистрационный номер или идентификационные данные средства измерений, который произвел измерение;

в сети с неизвестными участниками набор данных имеет однозначную электронную подпись, при этом электронная подпись перекрывает все области набора данных (тем самым гарантируется их подлинность и целостность);

приёмник массива данных проверяет все данные на их подлинность и целостность.

6.7.3.9. С помощью технических средств имитируется ситуация, когда программа, принимающая данные, обнаруживает несоответствие между набором данных и номинальным значением контрольной суммы (электронной подписи). При этом убеждаются, что принимающая программа сначала пытается восстановить правильное значение контрольной суммы, если доступна избыточная (добавочная) информация. Если же восстановление невозможно, то должно генерироваться соответствующее предупреждение пользователю, измеренное значение не выводится и устанавливается значок в искаженной области массива данных со значением «не действительно» или искаженный набор данных удаляется.

6.7.3.9. Для проверки функционирования средств защиты данных от прерывания передачи имитируется повреждение сети передачи данных, при этом убеждаются, что переданные данные не теряются.

6.7.4. Проверка разделения ПО на метрологически значимую и незначимую части.

Проверка разделения ПО на метрологически значимую и незначимую части изложена в разделе 6.2 настоящей рекомендации.

Библиография

1. OIML D 31. Edition 2008 (E). General requirements for software controlled measuring instruments. (Общие требования к программно-контролируемым средствам измерений).
2. WELMEC 7.2. Issue 5. Software Guide (Measuring Instruments Directive 2004/22/EC) March 2012 (Русский перевод: ВЕЛМЕК 7.2. Руководство по программному обеспечению. Директива 2004/22/ЕС на средства измерений) – М.; АНО «РУС-Консалтинг», 2009).
3. Административный регламент по предоставлению Федеральному агентству по техническому регулированию и метрологии государственной услуги по утверждению типа стандартных образцов или средств измерений, утвержденный приказом Минпромторга от 25.06.2013 г. № 970.
4. МИ 2174-91. ГСИ. Аттестация алгоритмов и программ обработки данных при измерениях. Основные положения.
5. H.R. Cook, M.G. Cox, M.P. Dainton, P.M. Harris. Testing Spreadsheets and Other Packages Used in Mertrology. A Case Study. Report to National Measurements System Policy Unit. September 1999.
6. Ю.А. Кудеяров, А.А. Сатановский. Генерация эталонных данных методом нуль – пространства для тестирования электронных таблиц, прикладных

математических пакетов и алгоритмов. Законодательная и прикладная метрология, № 2, 2005, с.с. 39-46.

7. В.А. Слаев, А.Г. Чуновкина. Аттестация программного обеспечения, используемого в метрологии: Справочная книга / Под ред. В.А. Слаева. – СПб.: «Профессионал», 2009 – 320 с.
8. Р 50.2.077-2014 ГСИ. Испытания средств измерений в целях утверждения типа. Проверка защиты программного обеспечения.

ЛИСТ СОГЛАСОВАНИЯ

Типовая методика испытаний и подтверждения соответствия (сертификации) программного обеспечения средств измерений (МИ 3566-2016)

наименование документа

Первый заместитель директора
ФГУП «ВНИИМС» по науке — личная подпись — Ф.В. Булыгин (инициалы, фамилия)

Руководитель лаб. 009 ФГУП «ВНИИМС» — личная подпись — Е.В. Кулябина (инициалы, фамилия)

Руководитель ИЛ ПО ФГУП «ВНИИМС» — личная подпись — А.Н. Паньков (инициалы, фамилия)