

6. Крыловецкий А. А., Протасов С. И. Алгоритмы анализа изображений в системах стереозрения реального времени // Вестник Воронежского государственного университета. Сер. Системный анализ и информационные технологии. 2010. № 2. С. 9–18.

7. Грузман И. С., Киричук В. С., Косых В. П. [и др.]. Цифровая обработка изображений в информационных системах. Новосибирск: Изд-во НГТУ, 2002. 352 с.

8. Фосайст Д., Понс Ж. Компьютерное зрение. Современный подход; пер. с англ. М.: Издат. дом Вильямс, 2004. 928 с. ISBN 5-845-0542-7.

9. Fursov V., Goshin Ye. Conformed identification of the fundamental matrix in the problem of a scene reconstruction, using stereo images // Image Mining. Theory and Applications Conf. Proceedings of IMTA-4 2013. 2013. P. 29–37.

10. Чумаченко А. В. Разработка и исследование алгоритмов установления пиксельных соответствий на стереопарах для решения задачи восстановления рельефа: дис. ... канд. техн. наук. Таганрог, 2014. 134 с.

**ПИВОВАРОВ Владимир Петрович**, кандидат технических наук, заведующий кафедрой «Электрооборудование и автоматика».

AuthorID (РИНЦ): 408317

Адрес для переписки: vpivovarov@mail.ru

**ЗУБАРЬ Алексей Владимирович**, кандидат технических наук, старший преподаватель кафедры «Электрооборудование и автоматика».

Адрес для переписки: alexey\_zubar@mail.ru

#### Для цитирования

Пивоваров В. П., Зубарь А. В. Дополнительные ограничения при поиске изображений на стереопарах вдоль эпиполярных линий // Омский научный вестник. 2018. № 2 (158). С. 124–128. DOI: 10.25206/1813-8225-2018-158-124-128.

Статья поступила в редакцию 16.02.2018 г.

© В. П. Пивоваров, А. В. Зубарь

УДК 004.02

DOI: 10.25206/1813-8225-2018-158-128-133

**Д. А. ПАНКОВ**  
**Л. А. ДЕНИСОВА**

Омский государственный  
технический университет,  
г. Омск

## КОНТРОЛЬ И ДИАГНОСТИКА НЕИСПРАВНОСТЕЙ ПРОГРАММНО- АППАРАТНОГО КОМПЛЕКСА

В работе предлагается комплекс показателей контроля основных неисправностей микроконтроллерной системы. Рассмотрены вопросы тестовой имитации неисправностей для автоматизации тестирования устройств на этапе разработки. Предложен подход к выявлению неисправностей программно-аппаратного комплекса с использованием программно-алгоритмических модулей контроля и диагностики. Экспериментально подтверждена эффективность данного подхода для контроля и диагностики сбоев и отказов в аппаратных компонентах при имитации неисправностей.

**Ключевые слова:** контроль и диагностика, имитация неисправностей, автоматизация тестирования, программно-аппаратный комплекс, аппаратный отказ.

**Введение.** Проектирование программно-аппаратных комплексов, состоящих из технически сложных аппаратных устройств, а также реализующих набор программ, выполняемых на одном или нескольких микроконтроллерных устройствах, является сложной задачей. Проблема контроля и диагностики неисправностей имеет актуальность для таких микроконтроллерных систем, как космические системы управления, информационно-вычислительные кластеры, поскольку для этих систем существуют жесткие требования к такому показателю надежности, как время наработки на отказ. Решение проблемы контроля и диагностики программно-аппаратных средств осложняется тем, что системы подвержены разнообразным внешним воздействиям, полный контроль которых физически нереализуем [1]. Отказ микросхем или воздействие излучения на устройство может привести к сбою при выполнении программы или к отказу интер-

фейса обмена между устройством и внешней средой. Следует отметить, что отдельной проблемой является проблема отказоустойчивости программного обеспечения, которое, к сожалению, нельзя рассматривать независимо от аппаратных средств, на которых оно реализовано [2].

В процессе испытаний изделия на надежность не всегда удается корректно установить все неисправности в программной и аппаратных составляющих, поэтому при эксплуатации комплекса устройств возможны экономические потери и невыполнение установленного в техническом задании показателя времени безотказной работы. Применительно к вычислительным комплексам можно говорить о различных типах входящих в них устройств, таких как устройство управления (УУ), рабочая станция оператора (РС) и приемно-передающее устройство (ППУ). В качестве примера на рис. 1 показана схема возможной организации программ-

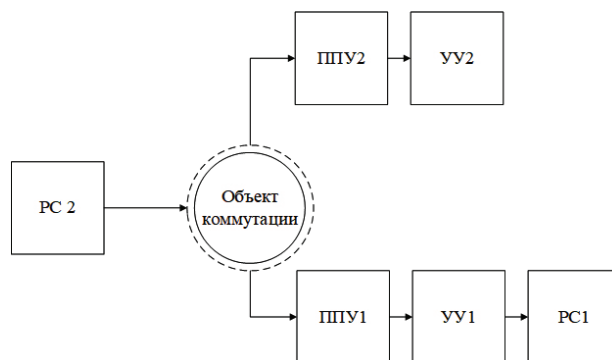


Рис. 1. Схема организации взаимодействия устройств в общей сети

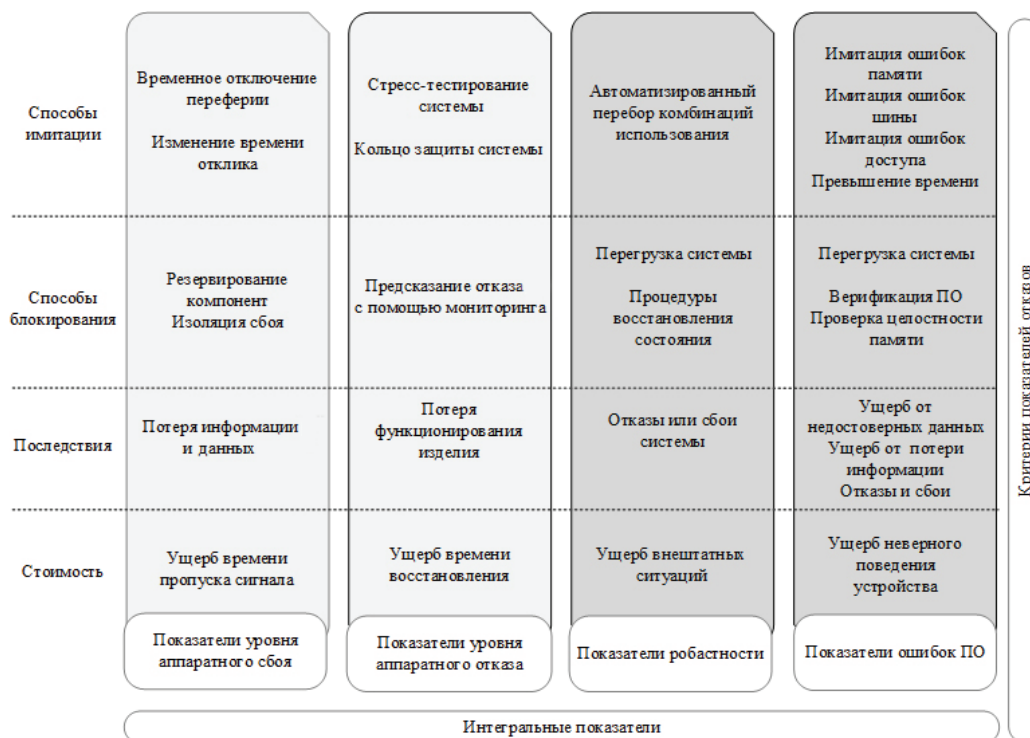


Рис. 2. Комплекс показателей оценки отказов и сбоев

но-аппаратного комплекса на базе такого набора средств. Как можно видеть, объект коммутации (в качестве которого рассматривается динамический узел радиосвязи) соединен в общую сеть с приемно-передающими устройствами, с устройствами управления, и доступен для связи с рабочей станцией оператора, удаленного от ППУ и УУ.

**1. Комплекс показателей оценки отказов программно-аппаратных средств.** Для эффективной диагностики отказов и сбоев программно-аппаратной системы требуется выделить основные типы причин, вызывающих появление отказа или сбоя [3, 4] и разработать комплекс показателей оценки отказов и сбоев. Определим основные типы неисправностей:

— аппаратный сбой — временный выход из строя компонента микроконтроллера с последующим возвращением к нормальному режиму работы;

— аппаратный отказ — выход из строя компонента микроконтроллера и невозможность его дальнейшей работы без замены компонентов или резервирования;

— отсутствие робастности — характеризует не устойчивость системы к входным данным, не определенным техническим заданием (под робастностью понимается свойство устойчивости системы при непредвиденных входных данных) [3];

— ошибки программного обеспечения (ПО) — непреднамеренное отклонение программы от заданного исполнения вследствие внутренних и внешних факторов (ошибки компиляторов, программистов, влияние аппаратных отказов и сбоев) [5].

Для принятия решения о реакции на системный отказ или сбой разработан комплекс показателей, представленный на рис. 2. Он включает четыре уровня:

— стоимостный уровень — нулевой уровень, задача которого состоит в диагностике ущерба, который способен нанести данный вид отказа или сбоя для контролируемой подсистемы;

— уровень оценивания последствий — первый уровень, задача которого состоит в выявлении основных параметров проявления неисправности, по которым можно распознать сигнатуру (вид) данного типа отказа или сбоя;

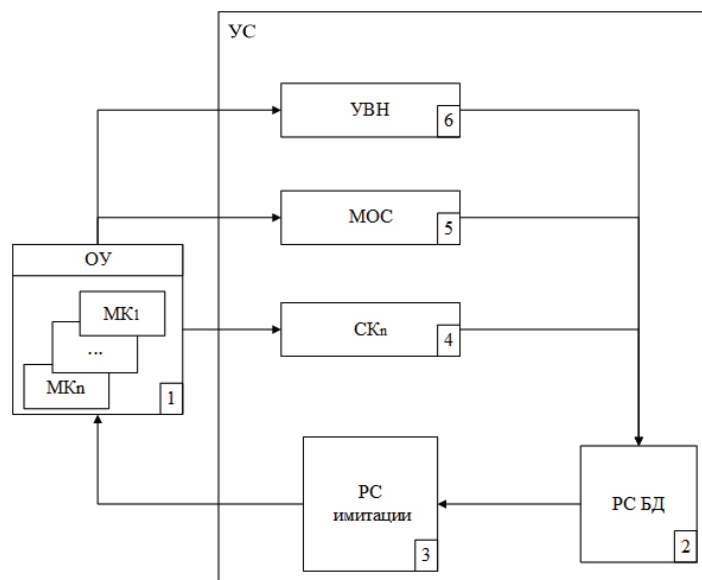


Рис. 3. Тестовая среда для имитации и контроля неисправностей

— уровень оценивания способа блокирования — *второй* уровень, задача которого состоит в описании мер по предотвращению или минимизации ущерба от данного вида отказа или сбоя;

— уровень определения способов имитации — *третий* уровень, задача которого состоит в описании подходов к организации тестовой инфраструктуры для проведения испытаний на надежность с моделированием данного вида отказа или сбоя.

Разработанный комплекс показателей и принятая классификация типов неисправностей позволяют определить меры, необходимые для устранения каждого вида отказа [6], поскольку часть сбоев не вносит в систему критических изменений. В связи с этим реакция на отказ или сбой может состоять в прогнозировании, предотвращении, смягчении последствий, сообщении пользователю или игнорировании отказа/сбоя [7].

Для смягчения последствий предлагаются способы блокирования для выбранного типа неисправностей. Для предотвращения части отказов и сбоев используется имитация, которая требует наличия специальной инфраструктуры в зависимости от типа имитируемого отказа или сбоя. При разработке комплексной системы виды реакций выбираются исходя из требований технического задания.

Для реализации процедур контроля и диагностики неисправностей программно-аппаратного комплекса, основанных на использовании разработанного комплекса показателей отказов и сбоев, создана тестовая среда, позволяющая осуществлять имитацию неисправностей. Тестовая среда состоит из двух подсистем — объекта управления (ОУ) и управляющей системы (УС). На рис. 3 приведена схема взаимодействия УС и ОУ.

ОУ представляет собой совокупность (1) тестируемых микроконтроллеров (МК) с общим входом и общим выходом (количество МК равно  $n$ ). На микроконтроллерах выполняется тестовая программа, которая обрабатывает входную последовательность данных и выполняет ряд действий, требуемый для обнаружения неисправностей в рамках теста. На выход с помощью блока (4) контроллера связи (СК $n$ ) передаются данные от микроконтроллерной системы. Данные поступают на блок (2) рабочей станции баз данных (РС БД), представля-

ющий собой устройство с развернутым сетевым сервисом баз данных. Это устройство передает информацию с входа микроконтроллера для хранения во встроенную память.

Блок (3) рабочей станции для имитации неисправностей (РС имитации) представляет собой устройство с загруженной программой, выполняющей классификацию признаков неисправностей, выбирающей данные от блока (2) и сохраняющей на этом блоке информацию о поданной на вход модуля последовательности с меткой времени. Модуль (5) обратной связи (МОС) регулярно представляет набор данных о работе системы на РС БД. Устройство (6) внешнего управления (УВН) представляет собой специализированный интерфейс съема информации с интерфейсов микроконтроллера, не обслуживаемых контроллером связи.

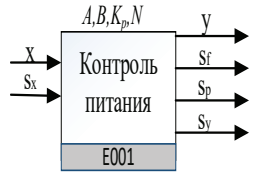
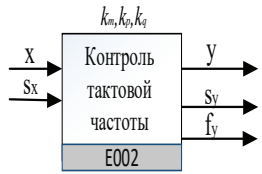
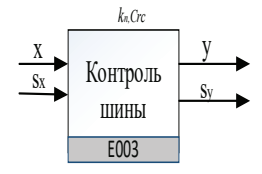
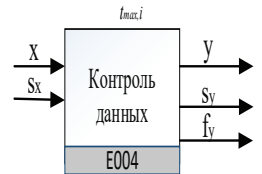
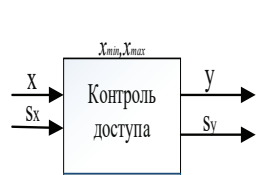
Тестовая среда обеспечивает обнаружение ошибок ПО или других видов ошибок, которые детектируются совместно с отказами и сбоями в ПО. Следует отметить, что для точного определения типа обнаруженной ошибки необходимо обеспечить явное определение большинства аппаратных отказов и сбоев, из-за которых сложно определить, к какому классу относится выявленный отказ или сбой.

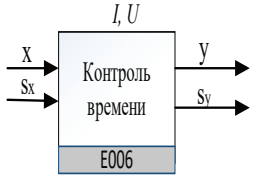
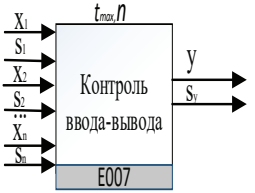
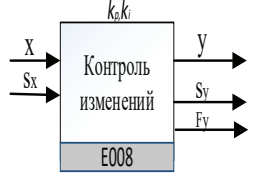
## 2. Выявление аппаратных отказов и сбоев.

Для выявления аппаратных отказов требуется использование модулей контроля, определяющих технические неисправности компонент комплекса. Одним из эффективных подходов является сравнение сигналов на входах и выходах и параметрическая настройка модуля. При отклонении работы модуля от режима нормального функционирования фиксируется неисправность в *микроконтроллерной системе*, после чего информация передается рабочей станции. Детектирование неисправностей программными средствами с применением разработанных модулей представляется целесообразным для основных подсистем микроконтроллера: системы управления внутренней памятью, шин данных, ввода-вывода, аппаратного таймера.

Процедура обнаружения неисправностей на базе разработанных модулей выполняется одновременно с работой тестовых программ, изменяющих значения параметров и имитирующих отказы и сбои, которые контролируют модули [8, 9].

Модули контроля и диагностики

Наименование модуля Изображение модуля	Входы и выходы модуля Параметры настройки	Алгоритм работы
<p>1. Модуль контроля питания</p> 	<p>Входы: x — величина напряжения питания;  <math>S_x</math> — признак недостоверности входного сигнала.                  Выходы: y — функция контроля питания;  <math>S_f</math> — признак отклонения среднего значения питания от граничных значений;  <math>S_p</math> — признак отклонения величины напряжения от граничных значений;  <math>S_y</math> — признак достоверности функции контроля.                  Параметры настройки:  <math>k_p</math> — коэффициент передачи;  <math>k_n</math> — размер буфера для замеров напряжения питания;                  A — нижняя граница напряжения питания;                  B — верхняя граница напряжения питания.</p>	$y = \sum_{i=1}^n k_p x_{t+i} \cdot k_n^{-1}$ $S_f = \begin{cases} 0 & \text{при } y \leq (B - A) \\ 1 & \text{при невыполнении} \end{cases}$ $S_p = \begin{cases} 0 & \text{при } A \leq x_t \leq B \\ 1 & \text{при невыполнении} \end{cases}$ $S_y = \begin{cases} 0 & \text{при } s_f = 0, s_p = 0, s_x = 0 \\ 1 & \text{при невыполнении} \end{cases}$
<p>2. Модуль контроля тактовой частоты шин</p> 	<p>Входы: x — значение тактовой частоты на входе микроконтроллера;  <math>S_x</math> — признак достоверности входной частоты.                  Выходы: y — реальное значение тактовой частоты;  <math>S_y</math> — признак достоверности выходной частоты;  <math>f_y</math> — функция сравнения основного и дублирующего выхода.                  Параметры настройки:  <math>k_m</math> — тактовая частота шины 1  <math>k_p</math> — тактовая частота шины 2  <math>k_q</math> — тактовая частота шины 3</p>	$y = \begin{cases} k_m k_p k_q^{-1} x & \text{при } S_x = 0 \\ 1 & \text{при невыполнении} \end{cases}$ $S_y = \begin{cases} 0 & \text{при } S_x = 0, f_y = 0 \\ 1 & \text{при невыполнении} \end{cases}$ $f_y = \begin{cases} 0 & \text{при } k_m k_p k_q^{-1} x y^{-1} < 0,1 \\ 1 & \text{при невыполнении} \end{cases}$
<p>3. Модуль контроля шины данных</p> 	<p>Входы: x — значение сигнала на шине данных;  <math>S_x</math> — признак достоверности установленной частоты.                  Выходы: y — полученные данные за <math>k_n</math> тактов;  <math>S_y</math> — признак недостоверности данных.                  Параметры настройки:  <math>k_n</math> — заданное число тактов шины;                  Crc — аппаратная функция вычисления контрольной суммы.</p>	$y = Crc(x(0; k_n - 4))$ $S_y = \begin{cases} 0 & \text{при } y = x(k_n - 4; k_n) \text{ и } S_x = 0 \\ 1 & \text{при невыполнении} \end{cases}$
<p>4. Модуль контроля корректности данных</p> 	<p>Входы: x — значение сигнала на входе контроля;  <math>S_x</math> — признак потери или повреждения данных.                  Выходы: y — функция сравнения обработанных данных с входным эталоном x;  <math>S_y</math> — признак отказа контроллера во время операции приема данных.  <math>f_y</math> — функция сравнения входного и выходного сигнала.                  Параметры настройки:  <math>t_{max}</math> — максимальное время реакции контроллера на заданной частоте;                  i — число проверок модуля.</p>	$y_t = \begin{cases} x_t & \text{при } S_x = 0 \\ y_{t-t_{max}} & \text{при невыполнении} \end{cases}$ $f_y = \begin{cases} 0 & \text{при } \sum_i y_i^{-1} x_i > \frac{1}{2} \\ 1 & \text{при невыполнении} \end{cases}$ $S_y = \begin{cases} 0 & \text{при } S_x = 0 \text{ и } f_y = 0 \\ 1 & \text{при невыполнении} \end{cases}$
<p>5. Модуль контроля нарушения прав доступа к памяти</p> 	<p>Входы: x — адрес обращения к области памяти;  <math>S_x</math> — признак привилегированного доступа для x.                  Выходы: y — функция проверки корректной обработки доступа;  <math>S_y</math> — признак отклонения среднего значения от граничных значений.                  Параметры настройки:  <math>x_{min}</math> — младший доступный адрес памяти;  <math>x_{max}</math> — старший доступный адрес памяти.</p>	$y = \begin{cases} 0 & \text{при } x_{min} \leq x \leq x_{max} \\ 1 & \text{при невыполнении} \end{cases}$ $S_y = \begin{cases} 0 & \text{при } S_x = 0 \text{ и } y = 0 \\ 1 & \text{при невыполнении} \end{cases}$

Наименование модуля Изображение модуля	Входы и выходы модуля Параметры настройки	Алгоритм работы
6. Модуль контроля времени выполнения 	Входы: $x$ — входной сигнал данных по интерфейсу $I$ ; $S_x$ — признак отказа интерфейса передачи данных. Выходы: $y$ — время передачи байта данных; $S_y$ — признак отклонения времени от допустимых значений. Параметры настройки: $I$ — номер интерфейса передачи; $U$ — регулируемая скорость передачи.	$y_t = \begin{cases} t \text{ при } (x_t - x_{t-1}) U^{-1} \leq t_{\max} \\ y_{t-t_{\max}} \text{ при невыполнении} \end{cases}$ $S_y = \begin{cases} 0 \text{ при } y = 0 \text{ и } S_x = 0 \\ 1 \text{ при невыполнении} \end{cases}$
7. Модуль контроля ввода-вывода 	Входы: $x$ — входной сигнал; $S_1 - S_n$ — признак запрещения доступа к портам ввода-вывода. Выходы: $y$ — изменение выходного сигнала по входному сигналу; $S_y$ — признак отказа порта ввода-вывода. Параметры настройки: $t_{\max}$ — максимальное время реакции порта ввода-вывода. $n$ — число проверяемых портов ввода-вывода.	$y_{t+t_{\max}} = X_t \bmod n$ $S_y = \begin{cases} 0 \text{ при } y_{t+t_{\max}} = (y_t + 1) \text{ и } S_1, \dots, S_n = 0 \\ 1 \text{ при невыполнении} \end{cases}$ $S_y = \begin{cases} 0 \text{ при } y = 0 \text{ и } S_x = 0 \\ 1 \text{ при невыполнении} \end{cases}$
8. Модуль контроля вносимых изменений 	Входы: $x$ — сигнал вносимого изменения; $S_x$ — признак отказа доступа по адресу. Выходы: $y$ — функция детектирования изменений; $S_y$ — признак корректности реакции; $f_y$ — признак отказа интерфейса. Параметры настройки: $k_p$ — признак внесения изменения в память процессора; $k_i$ — признак внесения изменения в интерфейс.	$y_t = \begin{cases} 0 \text{ при } S_x = 0, f_y = 0 \\ 1 \text{ при } S_x = 1 \text{ и } f_y = 1 \\ 2 \text{ при } S_x = 1 \text{ и } f_y = 2 \end{cases}$ $f_y = \begin{cases} 0 \text{ при невыполнении} \\ 1 \text{ при } x_t = k_i \\ 2 \text{ при } x_t = k_p \end{cases}$ $S_y = \begin{cases} 0 \text{ при } f_y = 1, y_t = 0; f_y = 2, y_t = 0 \\ 1 \text{ при невыполнении} \end{cases}$

В табл. 1 приведены основные из разработанных программно-алгоритмических модулей, предназначенных для выявления возникающих отказов и сбоев системы. Представлены разработанные алгоритмы вычисления показателей, реализуемые каждым модулем, а также входные/выходные сигналы и параметры настройки.

**3. Результаты экспериментов.** С целью проверки разработанных программно-алгоритмических средств контроля и диагностики в тестовой среде выполнены экспериментальные исследования. Производилась имитация таких технических неисправностей, как работа аппаратного обеспечения с частичными сбоями и временное отключение аппаратных модулей. Выполнялось несколько экспериментов с формированием различных видов отказов и сбоев, для обнаружения которых использовались разработанные модули контроля.

Для иллюстрации и получения качественной характеристики результатов исследования в табл. 2 приведены результаты формирования и детектирования модулями контроля пяти видов отказов и сбоев.

Анализ результатов показал, что модули контроля и диагностики системы позволяют зафиксировать сформированные тестовой средой отказы и сбои, но не во всех случаях. Как можно видеть, только неисправность *E004* (некорректные данные) была обнаружена с помощью «Модуля контроля корректности данных» во всех случаях ее имитации. При проверке неисправности *E008* (внесение изменений в данные) с использованием «Модуля контроля вносимых изменений» не удалось детектировать отказы.

На основании этого сделан вывод, что для эффективного обнаружения отказов и сбоев следует разработать методику совместного использования модулей, позволяющих определять разные типы неисправностей. Использование каждым модулем контроля дополнительной информации о выявленных другими модулями технических неисправностях повысит эффективность контроля и диагностики.

**Заключение.** В результате выполненных исследований разработан комплекс показателей контроля основных неисправностей микроконтроллерной



## Результаты эксперимента

Тип отказа/сбоя (идентификатор)	Наименование модуля (номер модуля)	Обнаружение отказа/сбоя	Пропуск отказа/сбоя
1. Сбой тактовой частоты шины (E002)	Модуль контроля тактовой частоты шин (Модуль 2)	да	да
2. Некорректные данные (E004)	Модуль контроля корректности данных (Модуль 4)	да	нет
3. Нарушение прав доступа к памяти (E005)	Модуль контроля нарушения прав доступа к памяти (Модуль 5)	да	да
4. Превышение времени выполнения (E006)	Модуль контроля времени выполнения (Модуль 6)	да	да
5. Внесение изменений в данные (E008)	Модуль контроля вносимых изменений (Модуль 8)	нет	да

системы. Предложен подход к выявлению неисправностей программно-аппаратного комплекса с использованием программно-алгоритмических модулей контроля и диагностики. Экспериментально подтверждена эффективность данного подхода для контроля и диагностики сбоев и отказов в аппаратных компонентах при имитации неисправностей в тестовой среде.

Предполагается дальнейшее исследование способов формирования отказов в тестовой среде имитации неисправностей, нацеленной на поиск ошибок в готовом устройстве при моделировании реальных алгоритмов функционирования [10]. Кроме того, представляется целесообразным для проведения экспериментов по выявлению аппаратных неисправностей дополнительно разработать генератор сценариев имитации, позволяющий учесть возможные отказы аппаратных средств, тем самым повысить эффективность контроля и диагностики неисправностей программно-аппаратного комплекса.

## Библиографический список

1. Увалов Д. В. Бортовая микропроцессорная система управления с повышенной сбоеустойчивостью // Решетневские чтения. 2014. Т. 1, № 18. С. 248–250.
2. Herder J., Bos H., Gras B. [et al.]. Failure Resilience for Device Drivers // 37th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN'07). URL: <http://www.cs.vu.nl/~ast/Publications/Papers/dsn-2007.pdf> (дата обращения: 10.01.2018).
3. Sussman G. Building Robust Systems an Essay. URL: <http://groups.csail.mit.edu/mac/users/gjs/6.945/readings/robust-systems.pdf> (дата обращения: 12.01.2018).
4. Laprie J. C., Arlat J., Beounes C., Kanoun K. Hardware and software-fault tolerance // Commission of the European Communities Directorate-general Telecommunications, Information Industries and Innovation (eds) ESPRIT '90. 1990. Springer, Dordrecht. DOI: 10.1007/978-94-009-0705-8\_63.
5. Построение отказоустойчивой системы. URL: <https://habrahabr.ru/post/118496/> (дата обращения 20.01.18).
6. Копылов С. А. Обзор методов и подходов обнаружения отказов и отказоустойчивости системы управления для рабо-

тизированных морских подвижных объектов // Инженерный вестник Дона. 2014. № 3. С. 16–20.

7. Дегтярев А. Р., Киселев С. К. Отказоустойчивые реконфигурирующиеся комплексы интегрированной модульной авионики // Электротехнические и информационные комплексы и системы. 2016. № 1. С. 89–99.

8. Савкин Л. В. Мажоритирование логико-арифметических операций на низком аппаратном уровне в реконфигурируемой диагностической модели бортового комплекса управления космического аппарата // Universum: технические науки. 2014. № 12 (13). С. 1–10.

9. Львович И. Я., Тузиков А. А. Модель выбора вариантов резервирования в системе управления стендовыми испытаниями // Вестник Воронежского государственного технического университета. 2011. Т. 7, № 10. С. 47–50.

10. Панков Д. А., Денисова Л. А. Разработка и исследование алгоритма маршрутизации в многофункциональном комплексе связи // Омский научный вестник. 2017. № 6 (156). С. 143–145.

**ПАНКОВ Денис Анатольевич**, аспирант кафедры «Автоматизированные системы обработки информации и управления».

Адрес для переписки: [pankovDDD@yandex.ru](mailto:pankovDDD@yandex.ru)

**ДЕНИСОВА Людмила Альбертовна**, доктор технических наук, доцент (Россия), профессор кафедры «Автоматизированные системы обработки информации и управления».

SPIN-код: 4926-3449

AuthorID (РИНЦ): 512788

Адрес для переписки: [denisova@asoiu.com](mailto:denisova@asoiu.com)

## Для цитирования

Панков Д. А., Денисова Л. А. Контроль и диагностика неисправностей программно-аппаратного комплекса // Омский научный вестник. 2018. № 2 (158). С. 128–133. DOI: 10.25206/1813-8225-2018-158-128-133.

Статья поступила в редакцию 01.03.2018 г.

© Д. А. Панков, Л. А. Денисова