

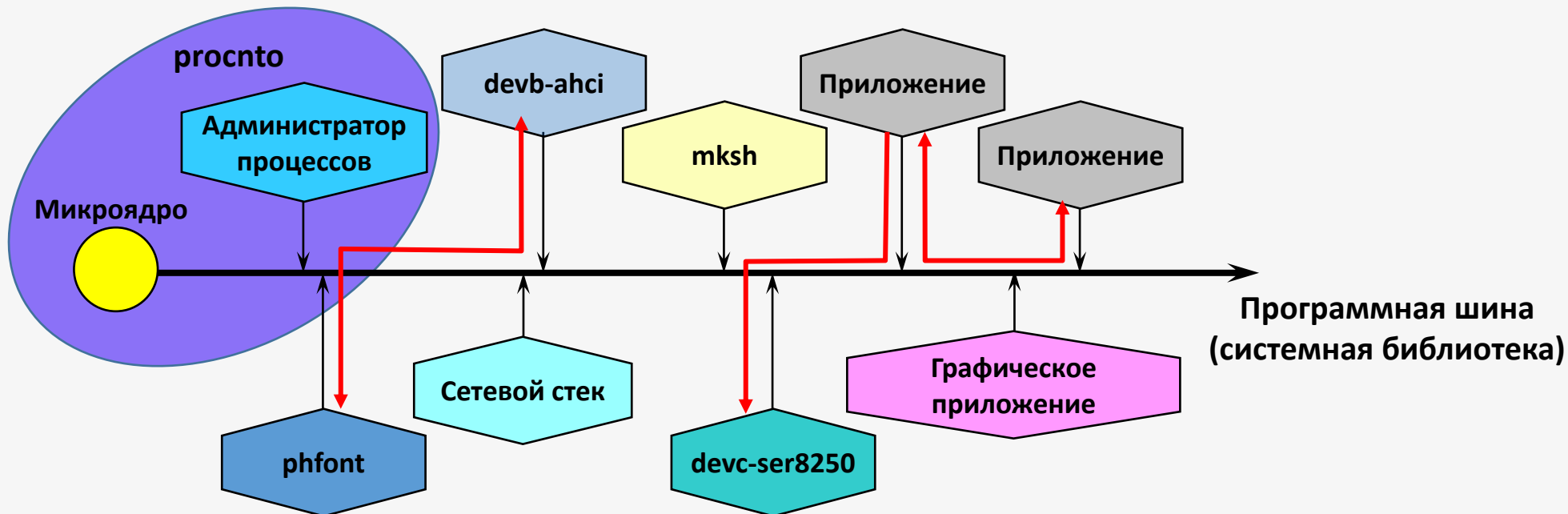
Принципы и инструменты разработки функционально безопасного ПО для ЗОСРВ «Нейтрино»

Сертификация по ГОСТ Р МЭК 61508

Подтверждение функциональной безопасности электрических, электронных и программируемых электронных (Э/Э/ПЭ) систем

- ◆ удостоверяет уровень полноты безопасности (УПБ) от УПБ 1 до УПБ 4
 - ◇ каждый УПБ обеспечивает определённые показатели снижения вероятности отказа системы
- ◆ функциональная безопасность ПО характеризуется уровнем стойкости к систематическим отказам (ССО) от ССО 1 до ССО 4
 - ◇ ПО с уровнем ССО n можно использовать в системах с УПБ вплоть до n
- ◆ требования к ПО изложены в ГОСТ IEC 61508-3—2018
- ◆ методы и средства описаны в ГОСТ Р МЭК 61508-7—2012

Микроядерная архитектура ЗОСРВ «Нейтрино»



- ♦ среда исполнения состоит из ядра (часть **procnto**), системной библиотеки (**libc.so**) и программ
- ♦ каждая программа, связанная с ФБ, должна проходить оценку соответствия требованиям ГОСТ IEC 61508-3 по соответствующему уровню ССО

Ядро и системная библиотека

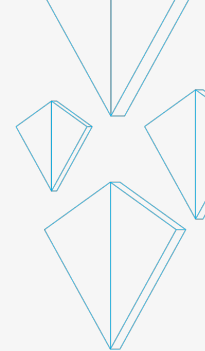
- ◆ обеспечивают работу программ в среде исполнения «Нейтрино»
- ◆ соответствуют требованиям ГОСТ IEC 61508-3-2018
 - ◇ по 3-му уровню стойкости к систематическим отказам программного обеспечения (CCO3/SC3)
 - ◇ этот уровень является максимальным для системного ПО



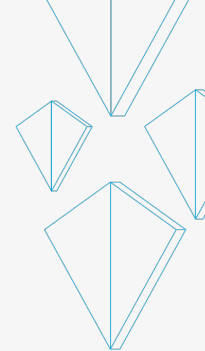
Функции ядра и системной библиотеки

В руководстве по функциональной безопасности ядра и системной библиотеки ЗОСРВ «Нейтрино» выделены пять классов функций:

- ◆ функции инициализации программ
- ◆ функции безопасности
- ◆ функции диагностики
- ◆ функции отладки
- ◆ функции интерфейсов к дополнительным модулям ЗОСРВ «Нейтрино»



Разработка программ

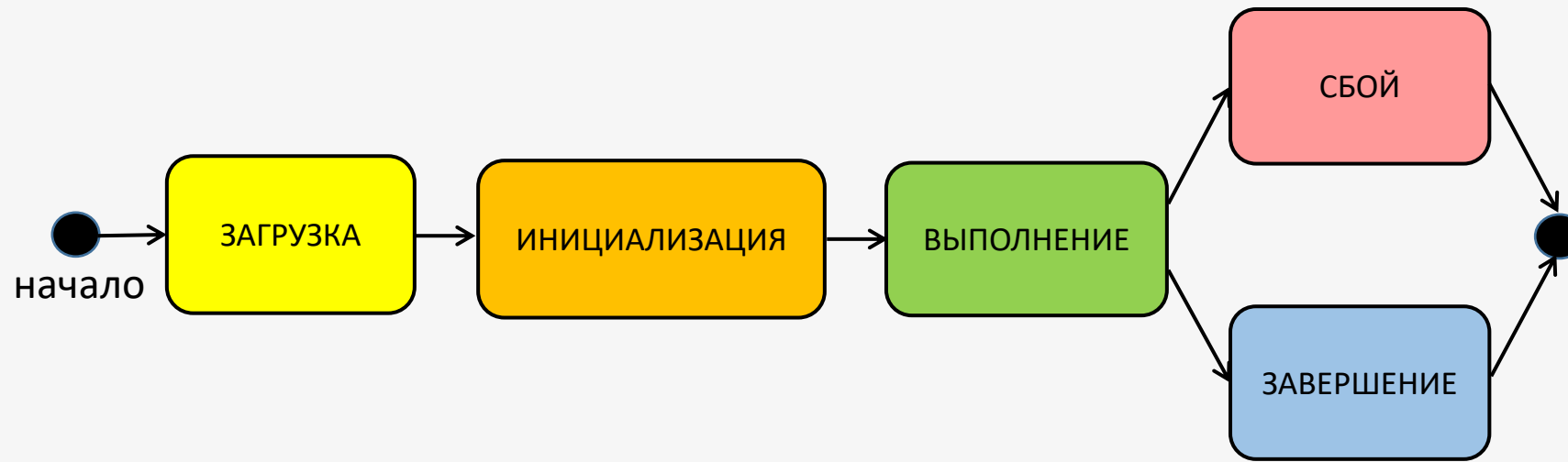


Для разработки ПО используется программный продукт «Комплект разработчика для ЗОСРВ «Нейтрино» редакции <...> », который содержит:

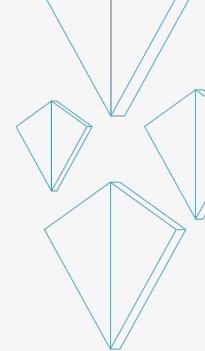
- ◆ компиляторы, компоновщики и прочие инструменты GNU для архитектур x86, ARM, PowerPC, MIPS, Эльбрус
- ◆ компоненты и инструменты для формирования сред исполнения ПО
- ◆ интегрированную среду разработки на основе Qt Creator
- ◆ средства интеграции с инструментами безопасной разработки ПО
- ◆ дополнительные возможности (поддержка различных сборочных систем, интерпретатора Python, библиотек Qt, ...)

Разработка программ

Конечный автомат состояний функционально безопасной программы:



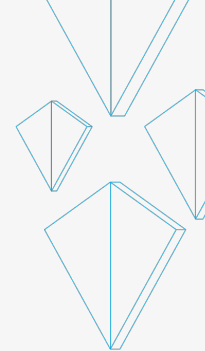
Состояния программы



◆ ЗАГРУЗКА

- ◆ размещение сегментов кода и данных программы в ОЗУ с последующим запуском:
 - ◆ модулем **procnto** из сценария загрузки
 - ◆ другой программой с помощью API системной библиотеки
- ◆ запущенная программа называется процессом
 - ◆ имеет **pid**, **uid**, **gid** и другие типовые UNIX-атрибуты
 - ◆ процесс может быть многопоточным

Состояния программы



◆ ИНИЦИАЛИЗАЦИЯ

- ◆ выделение и инициализация динамических ресурсов для программы:
 - ◆ выделение различных видов памяти
 - ◆ создание каналов, соединений, файловых дескрипторов и т. д.
 - ◆ создание мьютексов и других объектов синхронизации
 - ◆ инициализация всех видов данных, в том числе атрибутивных структур ресурсов
 - ◆ создание потоков и дочерних процессов
 - ◆ загрузка динамических библиотек

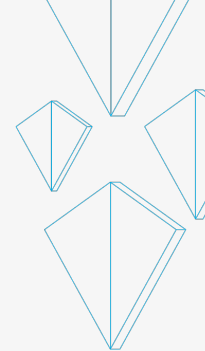
Состояния программы

♦ ВЫПОЛНЕНИЕ

- ♦ потоки программ используют выделенные ресурсы
- ♦ ядро планирует потоки
- ♦ запрещено вызывать функции системной библиотеки для динамического выделения ресурсов

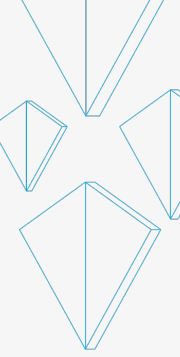
♦ ЗАВЕРШЕНИЕ и СБОЙ

- ♦ освобождение оперативной памяти и ресурсов:
 - ♦ по собственной инициативе программы
 - ♦ по инициативе другой программы с необходимыми правами
 - ♦ аварийно по инициативе средств диагностики ядра

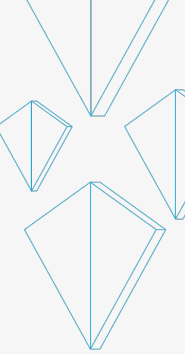


Реальное время и изоляция процессов

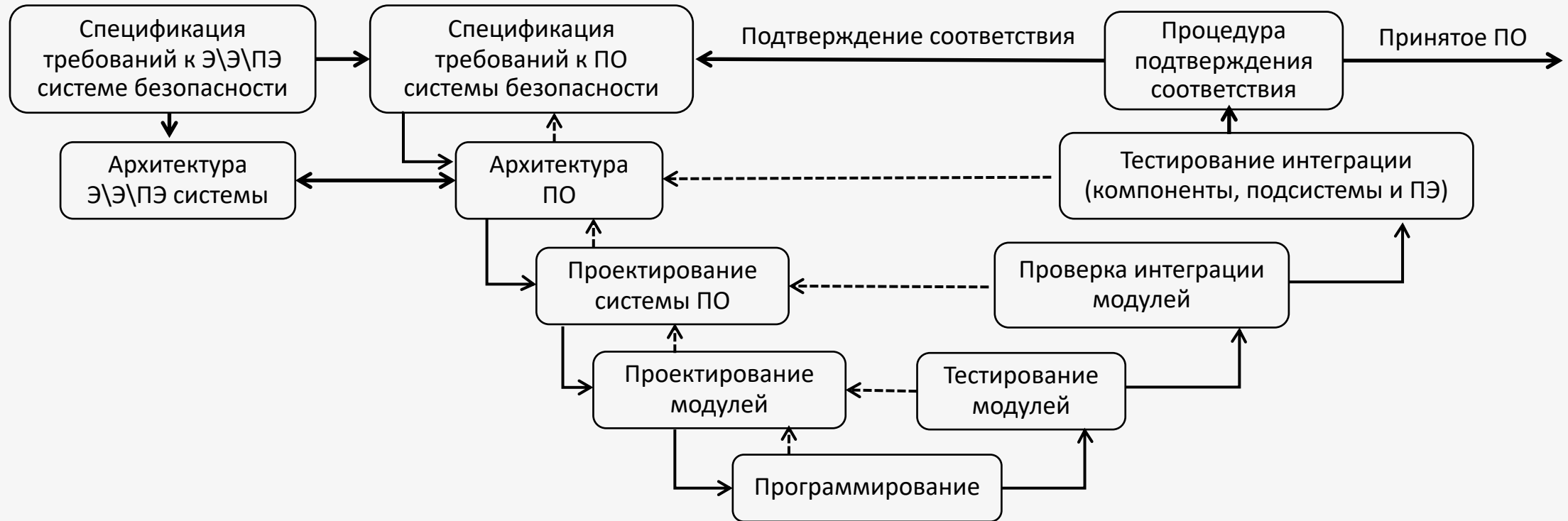
- ◆ для защиты от инверсии приоритетов поддерживается механизм наследования приоритетов при использовании синхронных сообщений и мьютексов
- ◆ для изоляции процессов во времени поддерживается механизм адаптивного партиционирования



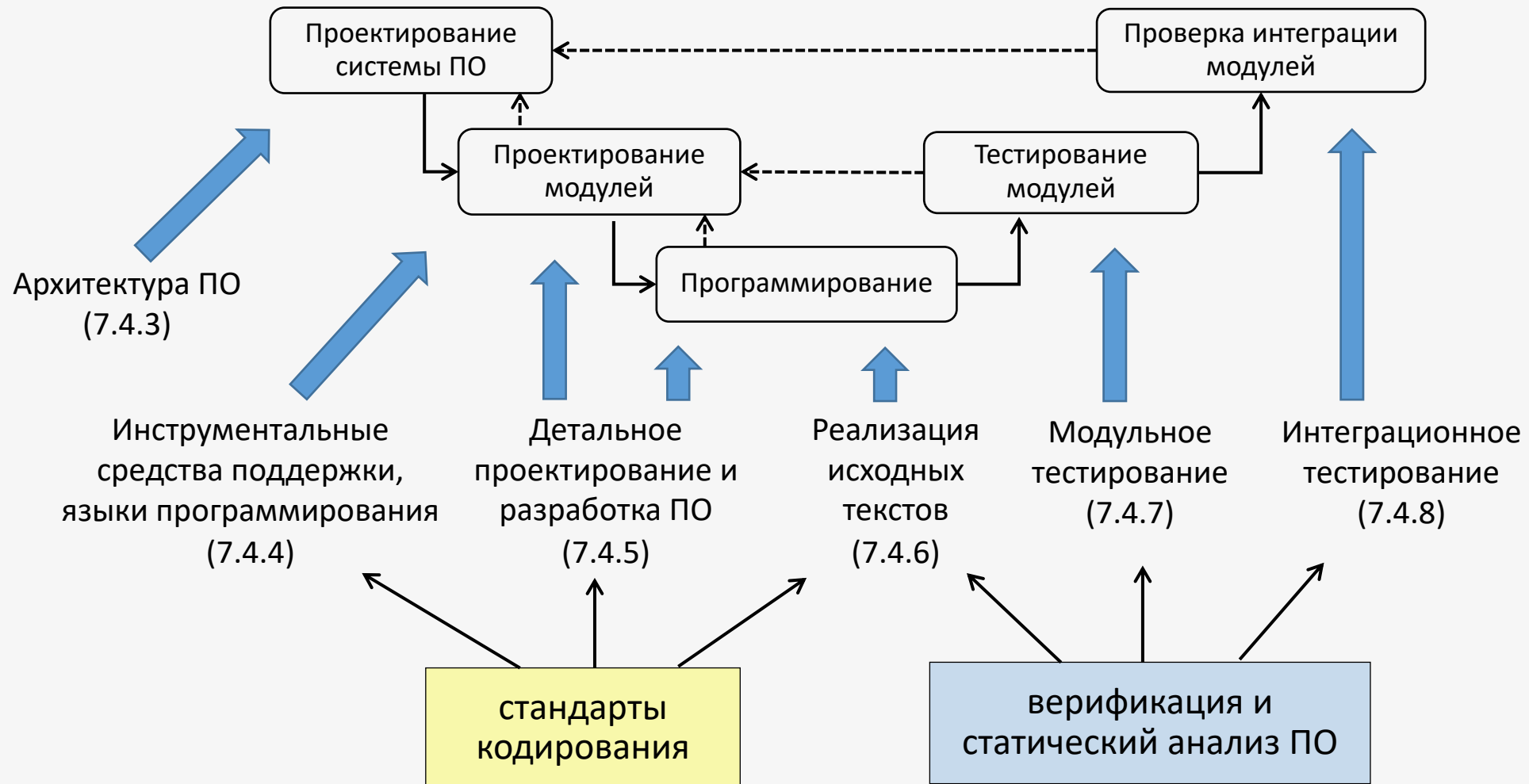
Жизненный цикл ПО системы безопасности



- ◆ должен быть выбран и специфицирован при планировании системы безопасности
- ◆ приемлемо адаптировать V-модель с учетом полноты безопасности и сложности проекта



Проектирование и реализация ПО системы безопасности



Языки программирования

Рекомендации по конкретным языкам программирования (ГОСТ Р МЭК 61508-7—2012, таблица С.1, фрагмент)

Язык программирования	УПБ 1	УПБ 2	УПБ 3	УПБ 4
Java	NR	NR	NR	NR
Java с подмножеством	R	R	NR	NR
FORTAN 77	R	R	R	R
FORTAN 77 с подмножеством	HR	HR	HR	HR
C/C++	R	-	NR	NR
C/C++ с подмножеством и стандартом кодирования, а также использование инструментов статического анализа	HR	HR	HR	HR
Ассемблер	R	R	-	-
Ассемблер с подмножеством и стандартом кодирования	R	R	R	R

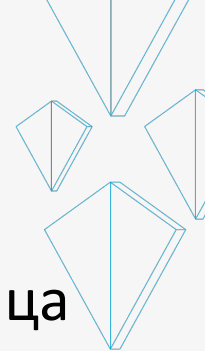
HR — настоятельно рекомендуется, R — рекомендуется, NR — не рекомендуется
По сути рекомендации являются требованиями различной степени строгости

Стандарты кодирования

- ◆ стандарт кодирования — набор правил, соглашений и рекомендуемых практик для написания исходного кода на некотором языке программирования
- ◆ предназначение стандартов кодирования с точки зрения требований ФБ (ГОСТ IEC 61508-3-2018, раздел 7.4.5 + таблица C.11):
 - ◇ отсутствие собственных ошибок проекта
 - ◇ простота и ясность
 - ◇ предсказуемость поведения
 - ◇ верифицируемость и тестируемость
- ◆ языки программирования для разработки всего ПО, связанного с безопасностью, должны использоваться в соответствии со стандартами составления программ для таких языков (п. 7.4.4.12)

Стандарты кодирования в ГОСТ Р МЭК 61508

Стандарты для проектирования и кодирования (ГОСТ IEC 61508-3—2018, таблица В.1, конкретизация — в ГОСТ Р МЭК 61508-7—2012, раздел С.2.6)



Метод/средство	УПБ 1	УПБ 2	УПБ 3	УПБ 4
Использование стандартов кодирования для сокращения вероятности ошибок	HR	HR	HR	HR
Не использовать динамические объекты	R	HR	HR	HR
Не использовать динамические переменные	—	R	HR	HR
Проверка создания динамических переменных в неавтономном режиме	—	R	HR	HR
Ограниченное использование прерываний	R	R	HR	HR
Ограниченное использование указателей	—	R	HR	HR
Ограниченное использование рекурсий	—	R	HR	HR
Не использовать неструктурированное управление в программах, написанных на языках высокого уровня	R	HR	HR	HR
Не использовать автоматическое преобразование типов	R	HR	HR	HR

Стандарт MISRA C/C++

- ◆ Стандарты разработки ПО на языках C/C++, разработанные организацией MISRA (Motor Industry Software Reliability Association)
- ◆ Цель — улучшение безопасности, переносимости и надёжности ПО для встраиваемых систем
- ◆ Правила MISRA C/C++ пересекаются с требованиями ГОСТ Р МЭК 61508

Требование ГОСТ IEC 61508-3	Правила MISRA C 2012
Не использовать автоматическое преобразование типов	Rule 10.x
Ограниченное использование указателей	Rule 11.x, 18.x
Не использовать неструктурированное управление в программах, написанных на языках высокого уровня	Rule 15.x
Ограниченное использование рекурсий	Rule 17.2
Не использовать динамические объекты	Rule 21.3

Статический анализ ПО в КР для ЗОСРВ «Нейтрино»

◆ штатный статический анализатор `cppcheck`

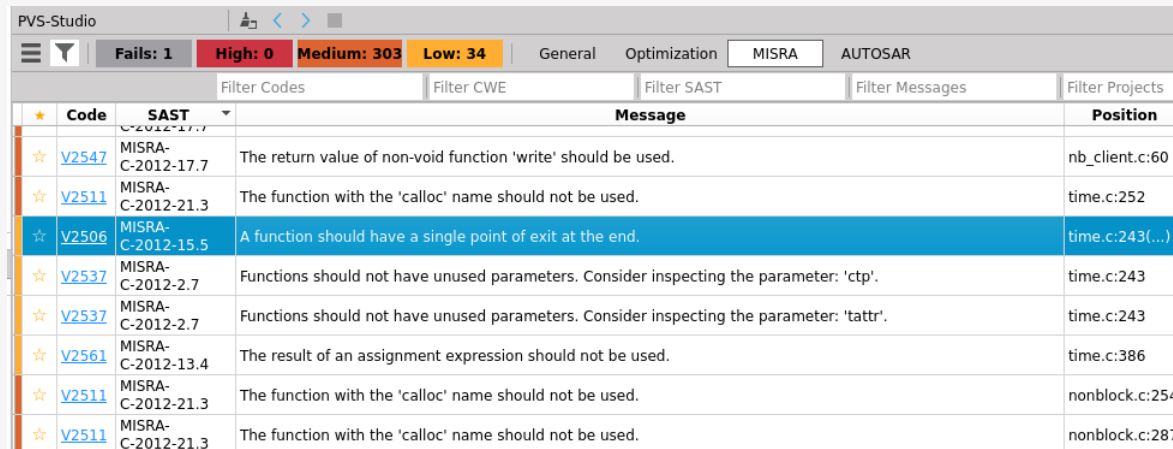
- ◆ лёгок в использовании
- ◆ обнаруживает типичные ошибки в коде на C/C++
- ◆ не поддерживает стандарт MISRA
- ◆ не учитывает межмодульные связи

◆ статический анализатор PVS-Studio

- ◆ поддерживает MISRA и другие стандарты кодирования
- ◆ учитывает межмодульные связи
- ◆ интегрируется в КР для ЗОСРВ «Нейтрино»
 - ◆ может использоваться в командно-строковом и графическом интерфейсах

Демонстрация

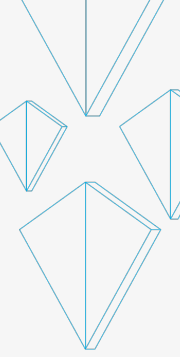
- ◆ статический анализ проекта в КР для ЗОСРВ «Нейтрино» с помощью PVS-Studio
 - ◇ в командной строке
 - ◇ в подключаемом модуле для Qt Creator редакции КПДА
- ◆ обработка отчёта о предупреждениях
 - ◇ интерпретация предупреждений
 - ◇ отображение различных категорий предупреждений
 - ◇ поиск и устранение нарушений правил стандарта MISRA



The screenshot shows the PVS-Studio application window. At the top, there's a status bar with 'Fails: 1', 'High: 0', 'Medium: 303', and 'Low: 34'. Below this are tabs for 'General', 'Optimization', 'MISRA', and 'AUTOSAR'. The 'MISRA' tab is selected. Below the tabs are filters for 'Filter Codes', 'Filter CWE', 'Filter SAST', 'Filter Messages', and 'Filter Projects'. The main area displays a table of warnings with columns for 'Code', 'SAST', 'Message', and 'Position'. The table lists several MISRA-C-2012-17.7, MISRA-C-2012-21.3, and MISRA-C-2012-15.5 violations. The row for MISRA-C-2012-15.5 is highlighted in blue.

Code	SAST	Message	Position
V2547	MISRA-C-2012-17.7	The return value of non-void function 'write' should be used.	nb_client.c:60
V2511	MISRA-C-2012-21.3	The function with the 'calloc' name should not be used.	time.c:252
V2506	MISRA-C-2012-15.5	A function should have a single point of exit at the end.	time.c:243(...)
V2537	MISRA-C-2012-2.7	Functions should not have unused parameters. Consider inspecting the parameter: 'ctp'.	time.c:243
V2537	MISRA-C-2012-2.7	Functions should not have unused parameters. Consider inspecting the parameter: 'tattr'.	time.c:243
V2561	MISRA-C-2012-13.4	The result of an assignment expression should not be used.	time.c:386
V2511	MISRA-C-2012-21.3	The function with the 'calloc' name should not be used.	nonblock.c:254
V2511	MISRA-C-2012-21.3	The function with the 'calloc' name should not be used.	nonblock.c:287

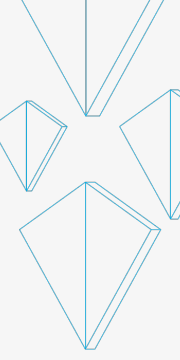
Требования к компетентности персонала



В ГОСТ Р МЭК 61508-1—2012 установлены требования к компетентности персонала:

- ◆ все лица, участвующие в жизненном цикле всей системы безопасности, её Э/Э/ПЭ или ПО, должны иметь компетентность, относящуюся к их обязанностям
- ◆ в руководстве по ФБ для ЗОСРВ «Нейтрино» определены следующие требования к компетентности персонала:
 - ◇ высшее / среднее специальное образование в области вычислительной техники / промышленной автоматизации
 - ◇ знание руководства по функциональной безопасности ядра и системной библиотеки ЗОСРВ «Нейтрино» и эксплуатационной документации
 - ◇ повышение квалификации по следующим направлениям (доступно в учебном центре ООО «СВД ВС» — см. следующий слайд)

Повышение квалификации



- ◆ разработка технических средств и программного обеспечения согласно требованиям ГОСТ Р МЭК 61508 (24 ак.ч.)
- ◆ разработка программного обеспечения для ЗОСРВ «Нейтрино» (32 ак.ч.)
- ◆ администрирование и диагностика ЗОСРВ «Нейтрино» (32 ак.ч.)
- ◆ инструментальные средства ЗОСРВ «Нейтрино» (32 ак. ч)
- ◆ РБПО для промышленных систем управления

Сертификация персонала



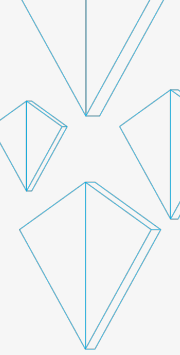
<https://fs.center>

ООО «НИЦ ФБ» — независимый орган по сертификации персонала

- ◆ подтверждение квалификации сотрудников в соответствии со стандартом ГОСТ Р ИСО/МЭК 17024
- ◆ схемы сертификации:
 - ◇ разработчик встроенного программного обеспечения
 - ◇ специалист по функциональной безопасности
- ◆ уровни квалификации:
 - ◇ I уровень - специалист
 - ◇ II уровень - профессионал
 - ◇ III уровень - эксперт
 - ◇ IV уровень - преподаватель

Заключение

- ◆ Ядро и системная библиотека ЗОСРВ «Нейтрино» позволяют выполнять программы, связанные с ФБ, в системах с уровнем полноты безопасности до УПБ 3 включительно
- ◆ Комплект разработчика для ЗОСРВ «Нейтрино» совместно со статическим анализатором PVS-Studio позволяет реализовывать требования ГОСТ Р МЭК 61508 к проектированию и реализации ПО
- ◆ Учебный центр ООО «СВД ВС» даёт специалистам возможность получать необходимую квалификацию в соответствии с требованиями ГОСТ Р МЭК 61508



Спасибо за внимание!

