

Обзор пакета программ SCADE

Краткий обзор средств проектирования
систем реального времени SCADE

Что такое SCADE?

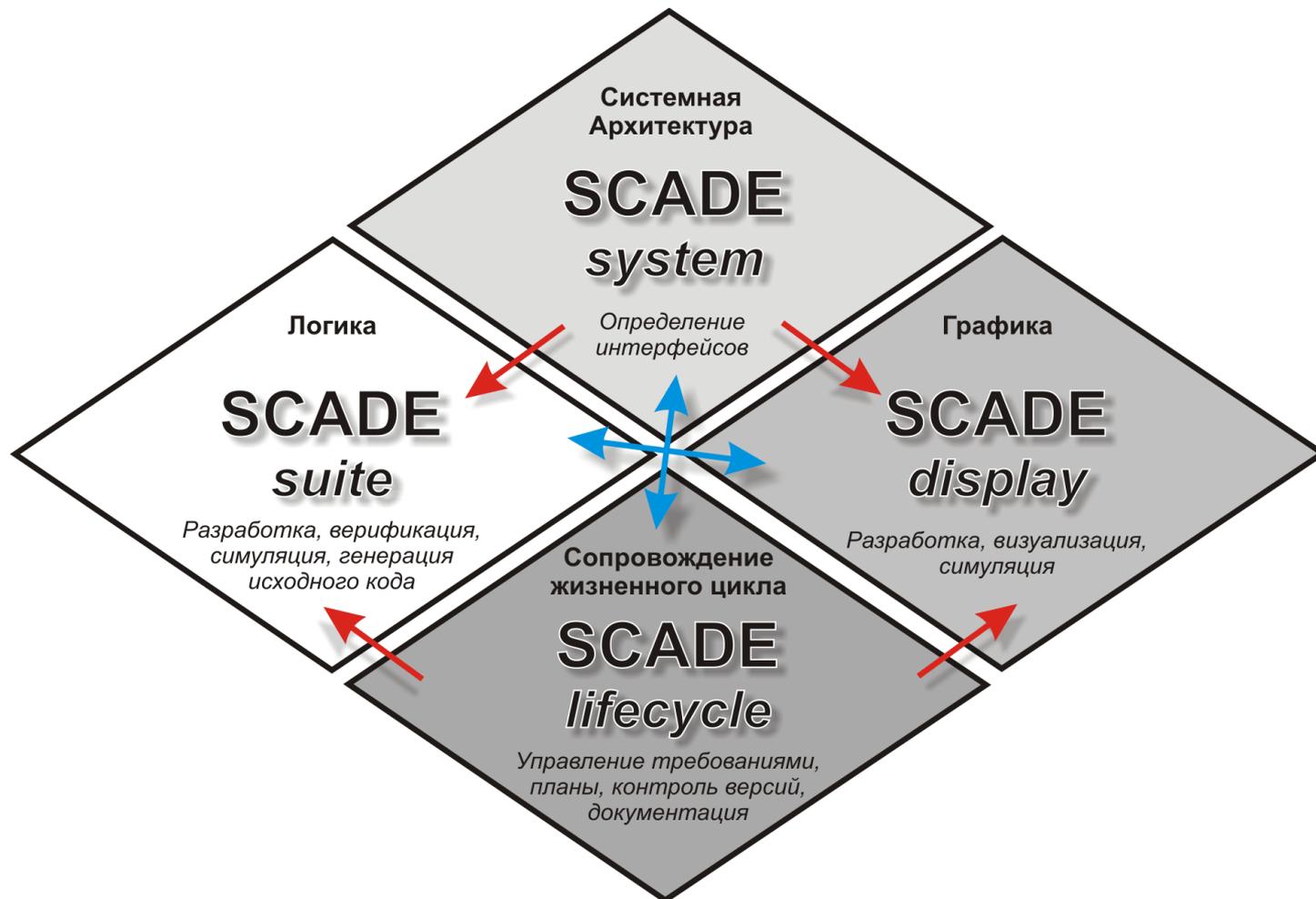
SCADE (**Safety-Critical Application Development Environment**) предназначена для проектирования систем со специальными требованиями к безопасности. SCADE - это среда визуального проектирования и автоматической генерации исходного кода. Код, сгенерированный SCADE, является сертифицированным по стандарту безопасности для авиации DO-178B.

Проектирование в SCADE складывается из построения диаграмм, которые затем используются для симуляции, верификации и генерации текстов программ на языке высокого уровня (Си или АДА).

SCADE используется ведущими отечественными авиастроителями, например, ОАО «Компания «Сухой», ГосНИИАС, ОАО «Концерн радиостроения «Вега», МИЭА («Авиаприбор»).

Состав SCADE

Компоненты среды разработки SCADE:



Кратко о SCADA System

SCADA System - это инструмент для разработки архитектуры встроенных систем, основанный на языке SysML. В SCADA System создается модель системы, включая подсистемы нижних уровней, затем выполняется их последующая коллективная разработка средствами SCADA Suite. SCADA System обеспечивает целостность данных, контроль версий и интеграцию со средствами управления требованиями и ведения проектов.

SCADA System встраивается в Eclipse посредством специального плагина (plug-in) и имеет функцию автоматической генерации документации на прикладную систему и ее компоненты, а также возможность открытия проектов компонентов в SCADA Suite, SCADA Lifecycle и SCADA Display. При этом проекты могут находиться как в репозитории, так и на удаленных компьютерах разработчиков.

Кратко о SCADE Lifecycle

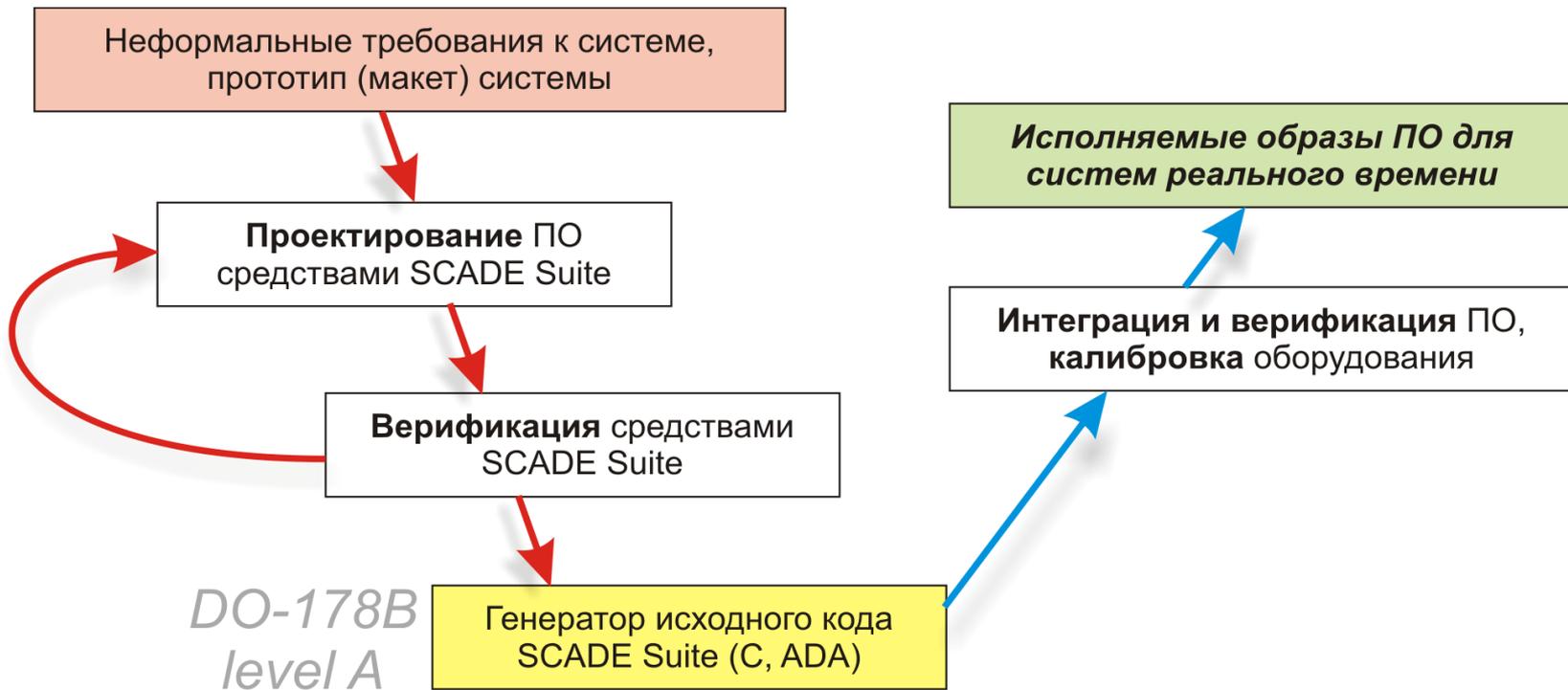
SCADE Lifecycle - это инструмент, в котором выполняются сбор и формулировка требований к системе (Requirements Management) и управление жизненным циклом проекта.

Основные функции SCADE lifecycle:

- протоколирование изменений, которые вносятся в проекты и документы;
- проверка вносимых изменений на соответствие стандартам и нормативным документам (например, проверка исходных текстов синтаксическим анализатором на предмет соответствия внутренним требованиям предприятия);
- извещение соответствующих групп пользователей и разработчиков о вносимых (внесенных) изменениях;
- корректировка планов и сроков в соответствии с вносимыми (внесенными) изменениями;

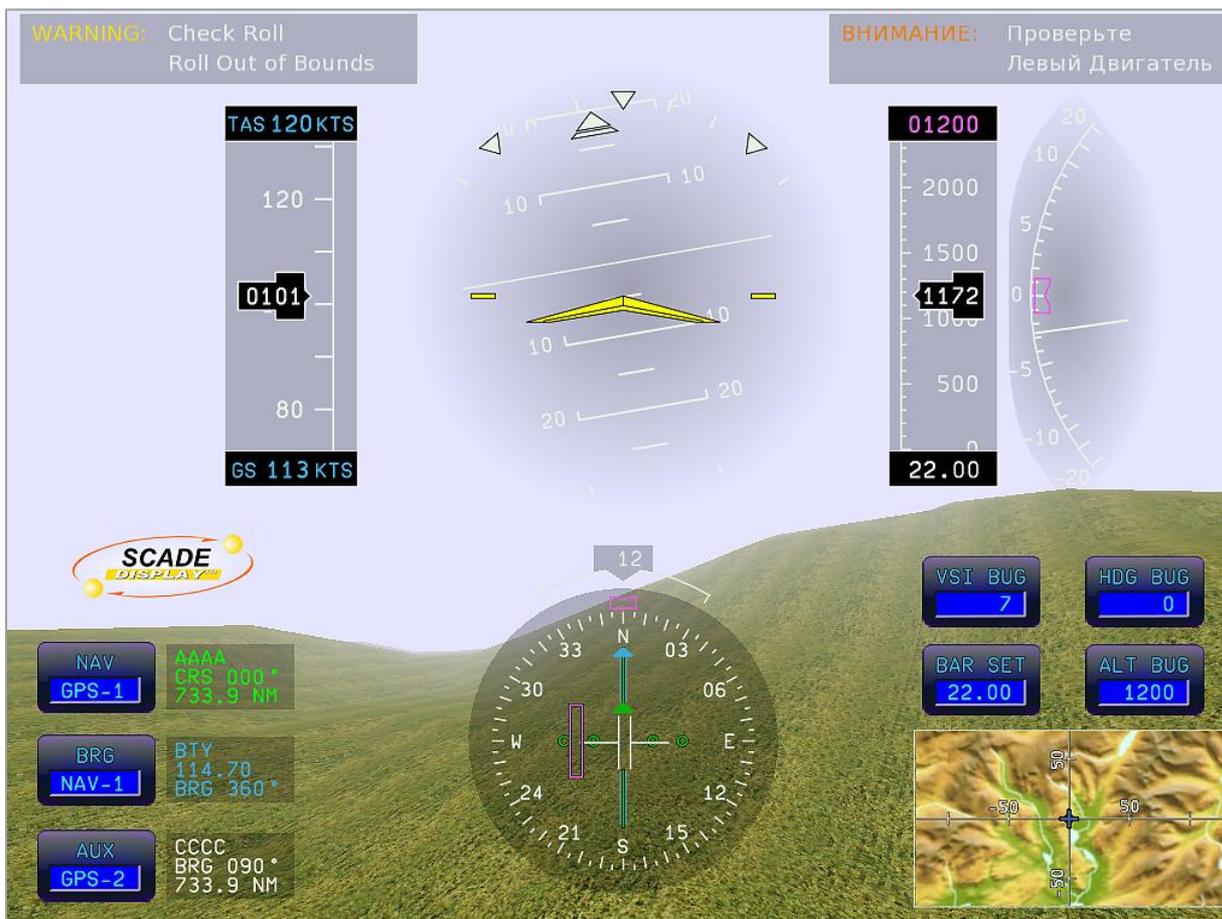
Кратко о SCADE Suite

SCADE Suite - средство разработки, моделирования и генерации исходного кода на языке высокого уровня.



Кратко о SCADE Display

SCADE Display - средство разработки графических интерфейсов и визуализации проектов SCADE Suite.



Модели в SCADE Suite

Модели в SCADE Suite представляются в графическом и текстовом виде.

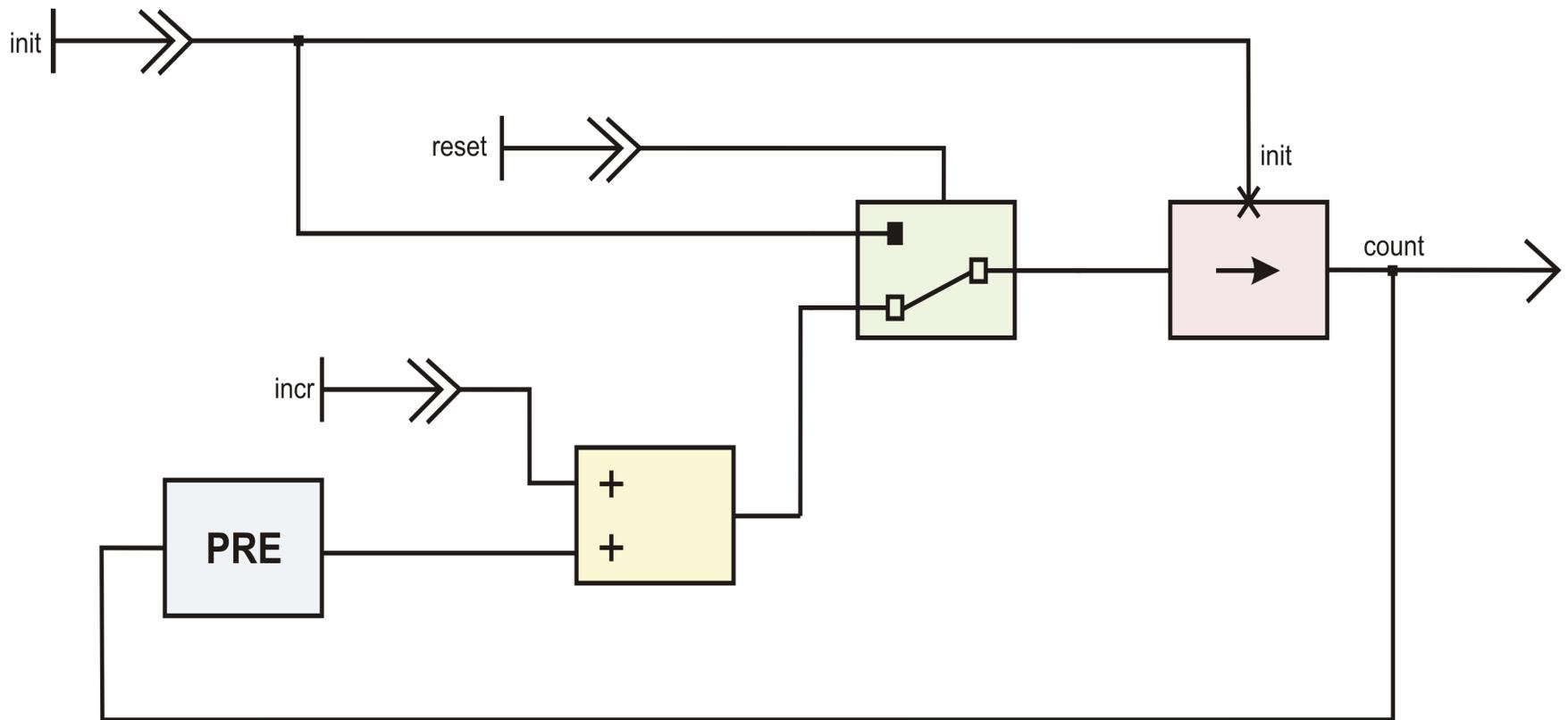
Графическое представление модели - это диаграммы и операторы. Операторы соединяются между собой связями. Текстовое представление модели - это представление модели (диаграммы или оператора) на языке Lustre.



```
node counter (init, incr: int; reset: bool)
returns (count: int );
```

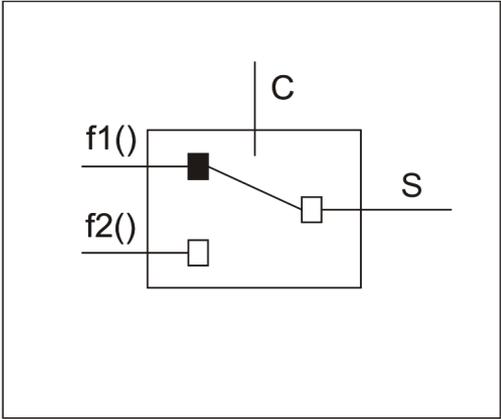
Модель «Счетчик»

Рассмотрим модель счетчика. Диаграмма модели счетчика, построенная средствами SCADE Suite:



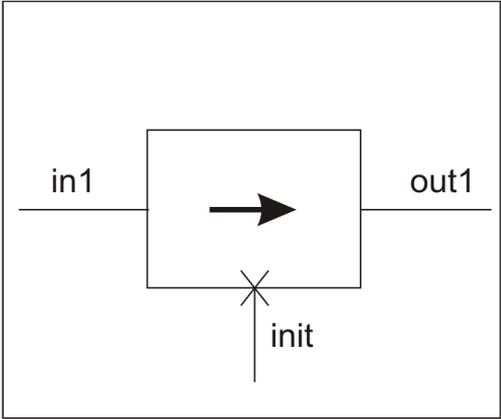
Операторы модели

На модели счетчика используется оператор выбора по условию, работа которого подобна конструкции **if-then-else**:

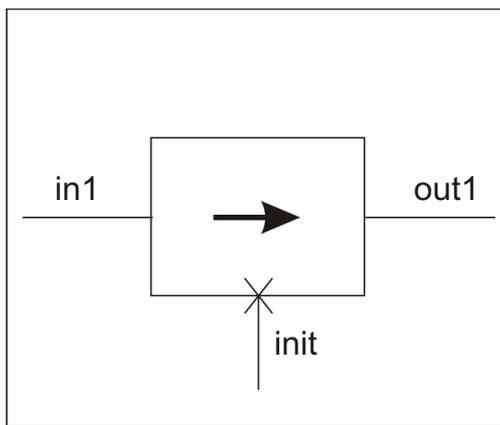
<i>Графическое представление</i>	<i>Текстовое представление</i>	<i>Комментарий</i>
	$S = \text{if } C \text{ then } f1() \text{ else } f2()$	Оператор <i>if-then-else</i> . Выбор определяется условием C.

Операторы модели

Оператор инициализации («Инициализатор»):

<i>Графическое представление</i>	<i>Текстовое представление</i>	<i>Комментарий</i>
	<pre>out1 = init → in1</pre>	<p>На первой итерации выходное состояние не определено. Инициализация приводит к появлению на выходе начального значения.</p>

Операторы модели

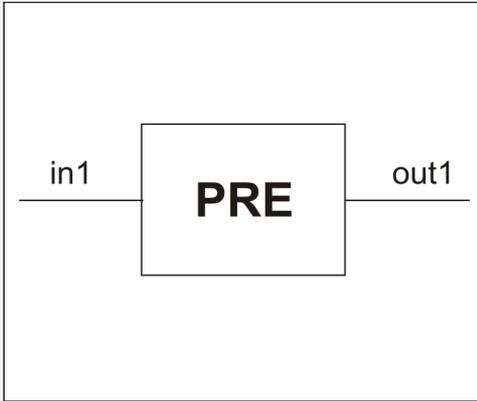


Оператор инициализации (\rightarrow) на первой итерации выдает на выходе значение входа **init**, а на последующих итерациях - значение на входе **in1**.
Таблица значений оператора инициализации:

Название входа/выхода	Значение входа/выхода			
	Итерация №1	Итерация №2	Итерация №3	Итерация №4
init	E1	E1	E1	E1
in1	F1	F2	F3	F4
out1 = init \rightarrow in1	E1	F2	F3	F4

Операторы модели

Оператор задержки:

Графическое представление	Текстовое представление	Комментарий
 <p>A block diagram of a delay operator. It consists of a large outer rectangle containing a smaller inner rectangle. The inner rectangle is labeled "PRE" in bold capital letters. A horizontal line enters the left side of the inner rectangle, labeled "in1". A horizontal line exits the right side of the inner rectangle, labeled "out1".</p>	<pre>out1 = pre (in1)</pre>	<p>Блок задержки. На первой итерации (первое обращение к <i>pre</i>, итерация $i=1$) возвращаемое значение не определено, далее возвращаемое значение равно $in1_{i-1}$.</p>

Операторы модели

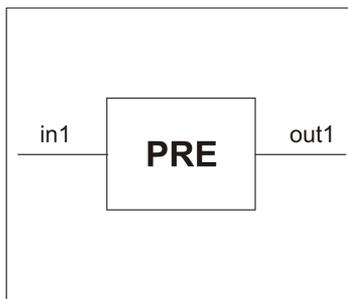


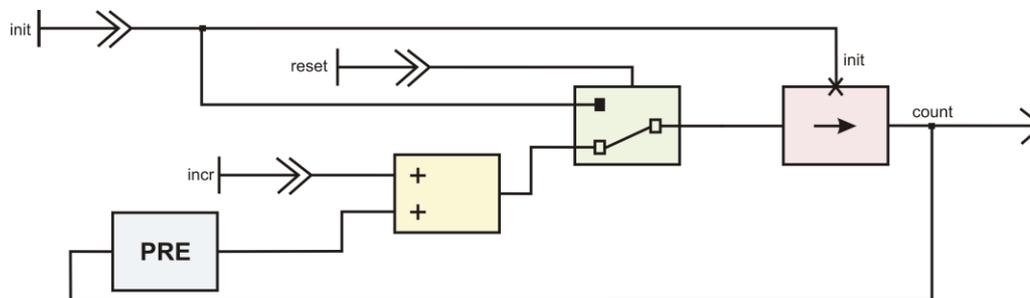
Таблица значений оператора задержки **Pre**:

Название входа/выхода	Значение входа/выхода			
	Итерация №1	Итерация №2	Итерация №3	Итерация №4
in1	E1	E2	E3	E4
out1	Не определено	E1	E2	E3

Оператор суммирования (**++**) выполняет арифметическое суммирование входных значений **in1** и **in2**: **sum = in1 + in2**

Модель «Счетчик» в действии

Таблица значений и текстовое представление модели «Счетчик»:



Название входа/ выхода	Значение входа/выхода								
	Итерация №1	Итерация №2	Итерация №3	Итерация №4	Итерация №5	Итерация №6	Итерация №7	Итерация №8	Итерация №9
init	0	0	0	0	0	0	0	5	5
reset	False	False	False	True	False	False	False	True	False
incr	1	1	1	1	2	3	4	1	1
count	0	$0+1=1$	$1+1=2$	0	$0+2=2$	$2+3=5$	$5+4=9$	5	$5+1=6$

```
count = init → if reset then init
else pre(count) + incr ;
```

Операторы времени SCADE Suite

В среде SCADE Suite часы позволяют операторам выполняться с разными частотами. Готовность выходного потока данных определяется внутренними часами этого потока, т. е. частотой, с которой вырабатываются его значения. Оператор выполняется в тот момент, когда готовы все его входные данные. В этот момент времени инициализируются базовые часы оператора.

В среде SCADE Suite часы поддерживаются операторами **WHEN**, **CURRENT** и **CONDUCT**.

Оператор **WHEN** позволяет уменьшать, а оператор **CURRENT** позволяет увеличивать частоту выполнения.

Операторы WHEN и CURRENT

Таблица значений для **WHEN** и **CURRENT**:

$Z = \text{CURRENT}(Y)$	x1	x1	x3	x4	x4	x4	x7
$Y = X \text{ WHEN } C$	x1		x3	x4			x7
CLOCK C	true	false	true	true	false	false	true
X	x1	x2	x3	x4	x5	x6	x7
	t1	t2	t3	t4	t5	t6	t7

Входные переменные **X** определены для каждого тика базовых часов оператора (**t1...t7**), а переменная **Y** определена только в том случае, если **C=True**. Переменная **Z** определена для каждого тика базовых часов оператора: если переменная **Y** определена, то **Z=Y**, иначе переменная **Z** хранит последнее определенное значение **Y**.

Операторы WHEN и CURRENT

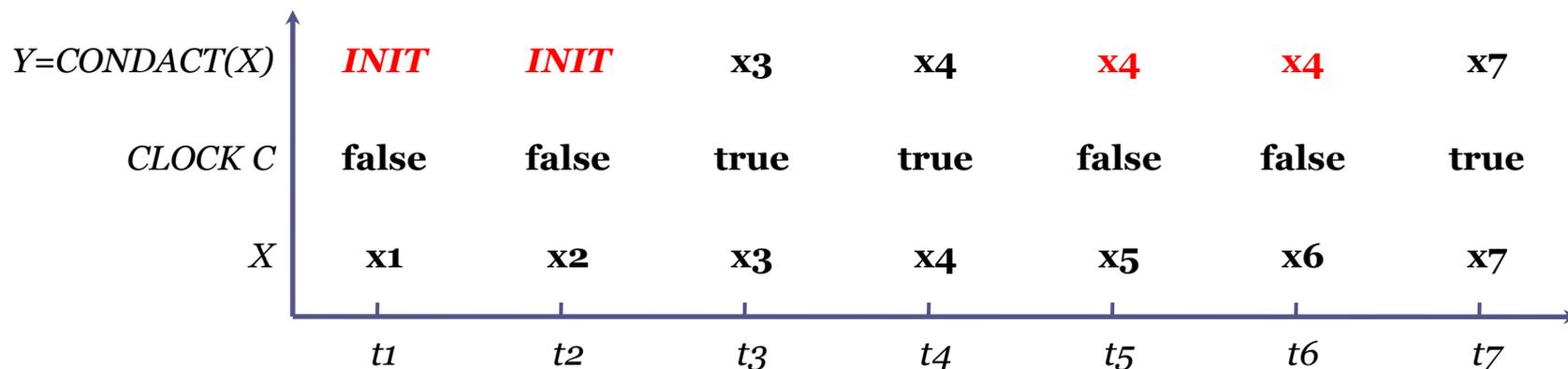
Если переменная Y не определена на первой итерации выполнения оператора, то таблица значений выглядит следующим образом:

$Z = \text{CURRENT}(Y)$	nil	nil	x3	x4	x4	x4	x7
$Y = X \text{ WHEN } C$			x3	x4			x7
$\text{CLOCK } C$	false	false	true	true	false	false	true
X	x1	x2	x3	x4	x5	x6	x7
	t_1	t_2	t_3	t_4	t_5	t_6	t_7

Значение **nil** означает, что переменная не определена в данной итерации.

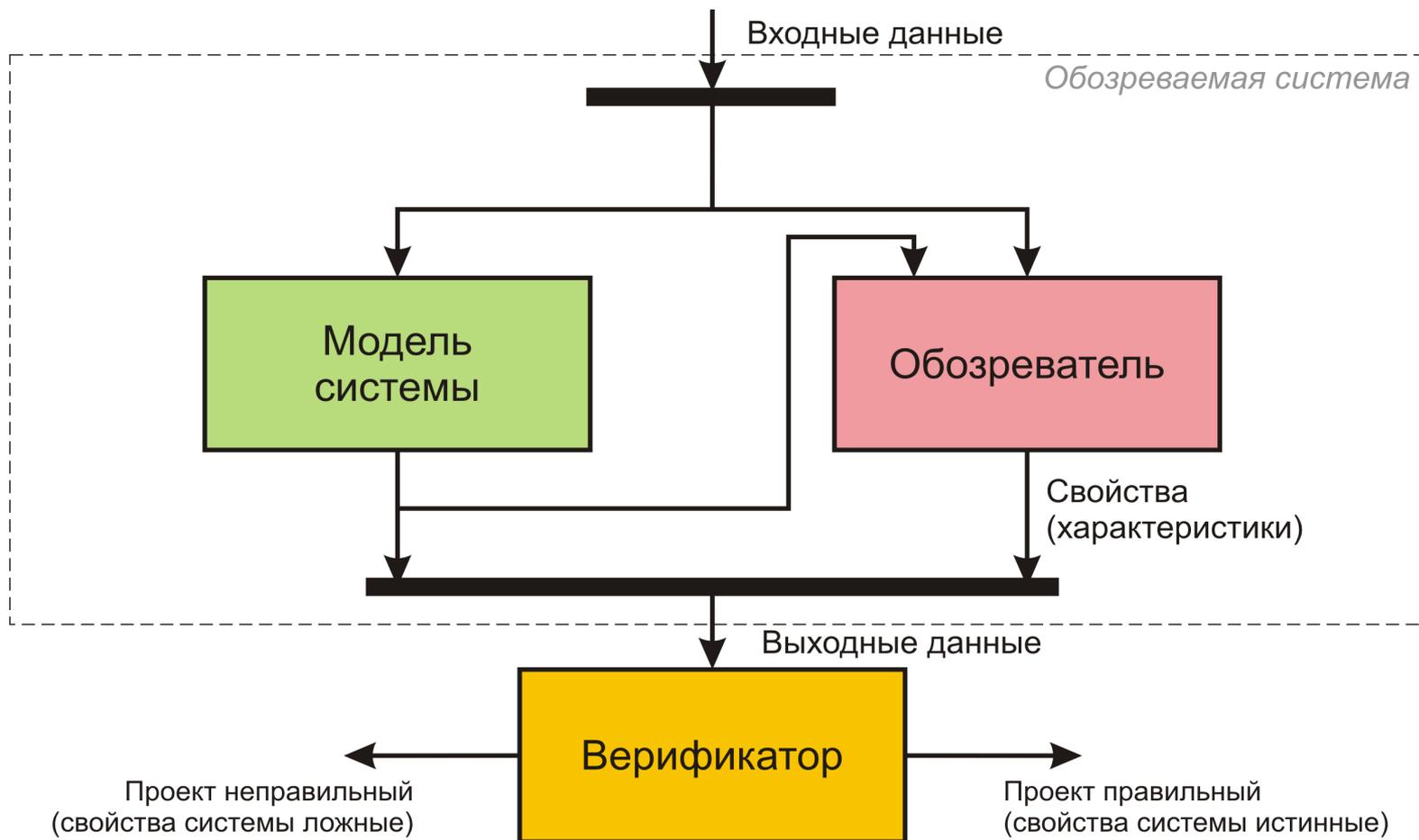
Оператор CONDACT

Примитив **CONDACT** реализован на основании **WHEN** и **CURRENT**. Он выполняет инициализацию выходных данных на первой итерации базовых часов оператора, если значение **CLOCK** является ложным.



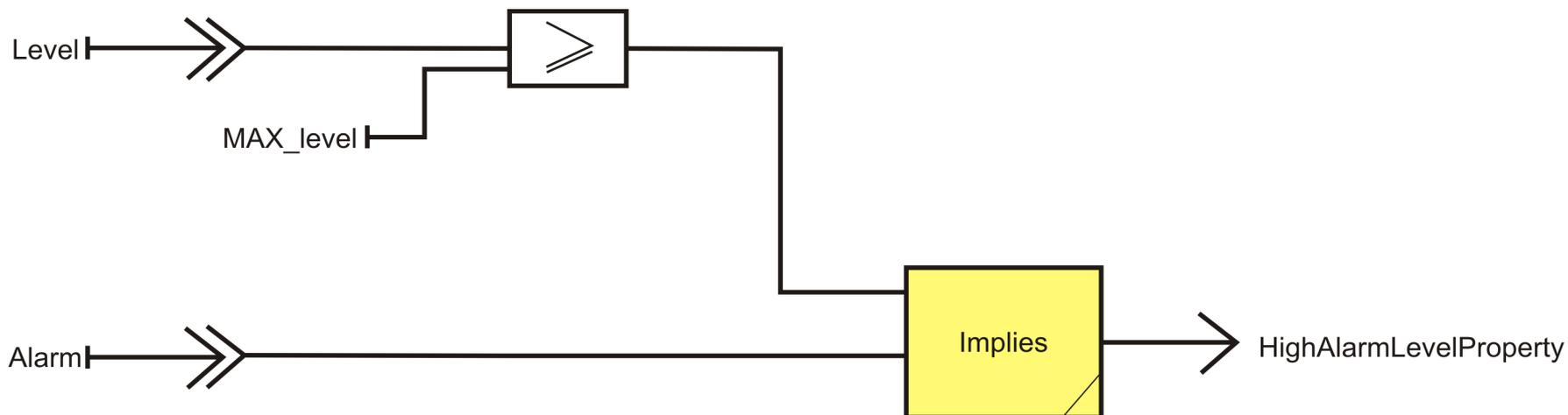
Верификатор SCADE Suite

Обобщенная схема верификации модели в SCADE Suite:



Верификатор SCADE Suite

Свойства системы описываются моделью в SCADE Suite:



Если $Level \geq MAX_level$, то выход `HighAlarmLevelProperty` оператора `Implies` будет установлен в значение **TRUE**.

При верификации или симуляции системы значение `HighAlarmLevelProperty` будет протоколироваться средой SCADE Suite.

Симулятор SCADe Suite

Симулятор позволяет выполнять имитационное моделирование и отладку моделей SCADe посредством выполнения Си кода, сгенерированного генератором исходного кода SCADe Suite. Симуляция модели выполняется следующим образом:

- выбирается оператор или диаграмма;
- строится бинарный код (исполняемый образ) модели;
- выполняется запуск бинарного кода;
- анализируются результаты выполнения бинарного кода.

Симулятор позволяет пользователю указывать значения входных данных, количество циклических прогонов модели, параметры остановки и возобновления прогона, параметры повторного запуска.

Симулятор SCAD E Suite

На симулируемых диаграммах можно задавать так называемые «**зонды**» - новые переменные, ассоциируемые со входными или выходными потоками данных. «**Зонды**» задаются до запуска процесса симулирования. После запуска процесса симулирования «**зонды**» будут изменять свои значения.

Остановы - это специальные утверждения, которые вставляются в диаграмму перед запуском процесса симулирования. В случае если значение такого утверждения оказалось ложным, процесс симулирования останавливается.

Пакетная симуляция выполняется на основании сценариев. Сценарии разрабатываются на языке TCL (Tool Command Language – «командный язык инструментов»).

Генератор кода SCAD E Suite

В состав SCAD E Suite входят следующие генераторы исходного кода:

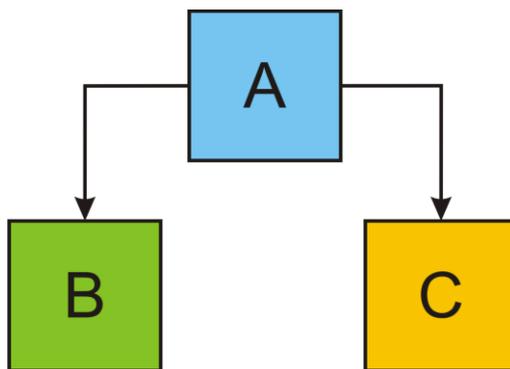
- Генератор исходного кода (ANSI C) общего использования;
- Сертифицированный генератор (C, DO-178 level A);
- ADA-генератор исходного кода (ADA95 и Spark ADA).

Сгенерированный средствами SCAD E Suite исходный код обладает следующими свойствами:

- форматированный, пригодный для анализа и контроля;
- переносимый, то есть независимый от целевой архитектуры (отсутствуют системные вызовы ОС и драйверов);
- модульный;
- сгенерированный код не содержит функций динамического распределения памяти;
- конечное время выполнения (без тупиков и зацикливаний);
- исходный код является оптимизированным

Режимы работы генератора

Рассмотрим генерацию кода для следующей модели SCADE:



Генерация выполняется в следующих режимах:

- **режим вызовов** - операторы SCADE Suite генерируются в виде функций, а затем эти функции вызываются из головной функции программы;
- **режим встраивания** - операторы SCADE Suite генерируются в виде последовательности лексем, а затем эти лексемы встраиваются в головную функцию программы.

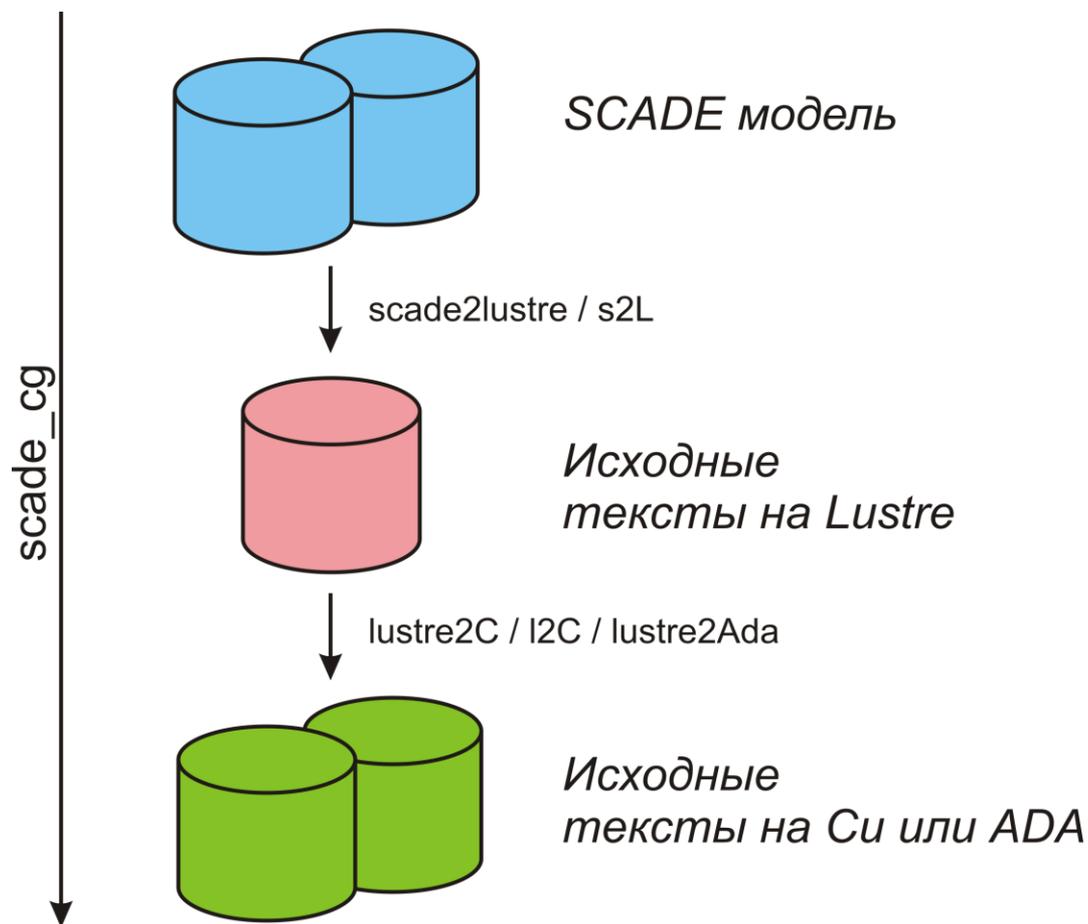
Режимы работы генератора

Особенности режимов генерации:

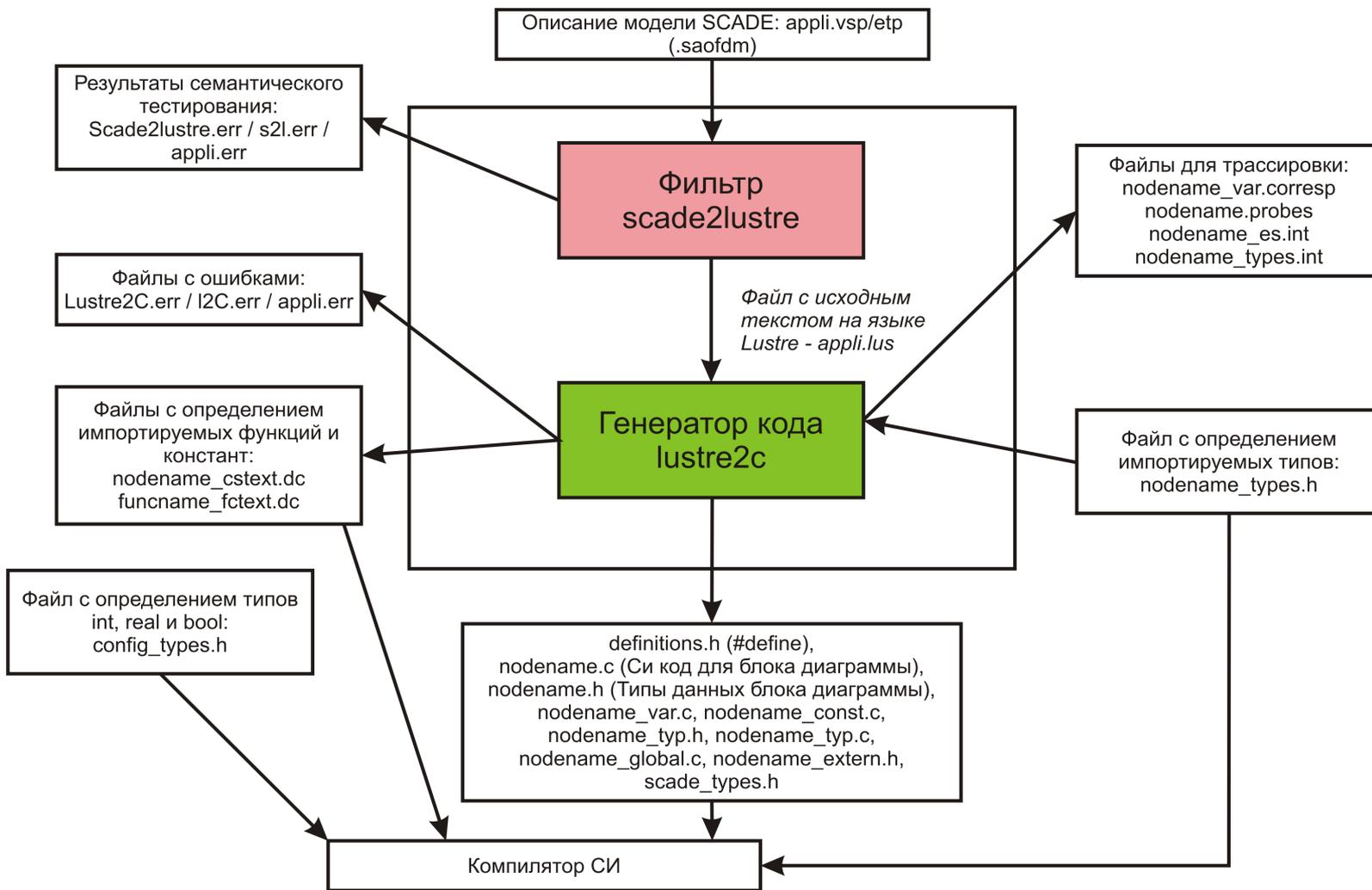
<i>Режим вызовов</i>	<i>Режим встраивания</i>	<i>Режим встраивания без возможности трассировки</i>
<pre> B() {...} C() {...} A() { ... B(); ... C(); ... } </pre>	<pre> A(){ ... /* begin of B */ ... /* end of B */ ... /* begin of C */ ... /* end of C */ ... } </pre>	<pre> A(){ ... lines of A, B and C ... } </pre>

Модель → Lustre → Си

Механизм генерации исходного кода из модели SCADE:



Процесс генерации кода



SCADE Display

SCADE Display используется совместно со SCADE Suite. В SCADE Display строится графическая часть (спидометры, панели отображения), а затем выполняется связывание с операторами модели SCADE Suite.

Особенности SCADE Display:

- Поддержка технологии OpenGL;
- Поддержка стандарта ARINC 661, который служит для стандартизации систем бортовой индикации (CDS Cockpit Display Systems) и взаимодействия этих систем с прикладными программами, управляющими бортовым оборудованием;
- Возможность сборки модели как stand-alone приложения, выполняющего симуляцию и графическую индикацию.

SCADE Display

SCADE Display позволяет выполнять симуляцию графических детализаций без написания сложных сценариев в интерактивном режиме.

Система проверки требований эргономики **SCADE Display Ergonomics Checker** предоставляет возможность проверки степени соответствия графических детализаций некоторым эргономическим правилам проектирования еще на этапе графического макетирования. Данный модуль позволяет пользователю оптимизировать графические детализации, а также производительность управляющего исходного кода.

SCADE Display содержит шлюз **Configuration Management (CM) Gateway** к системам управления конфигурацией. Этот шлюз поддерживает **CM**-средства, совместимые с Microsoft SCCI API.

SCADE и TCAГ СПО: отличия

TCAГ СПО имеет следующие отличительные особенности:

- Для построения моделей используются язык UML;
- Для представления функций используются диаграммы программной деятельности, из которых генерируются исходные тексты;
- В библиотеке TCAГ СПО присутствуют все функции ОС РВ Багет 2.0, ППМ, подмножество конструкций языка Си; библиотека может быть расширена примерами программ, реализующих стандартные для систем реального времени функции (обмен по МКИО, работа с общей памятью, организация нескольких потоков управления и т.д.);
- В состав TCAГ СПО входит интегрирующая среда, в которой выполняется управление проектами (коллективная разработка), компиляция исходных текстов, конфигурирование и построение образов ОС РВ Багет 2.0.

ТСАГ СПО: плюсы и минусы

ТСАГ СПО имеет следующие **достоинства**:

- Отечественная разработка, прошедшая сертификацию;
- Полная поддержка ОС РВ Багет 2.0 и ППМ;
- Использование открытых стандартов (UML, XML);
- Выполнение в среде Linux-совместимой ОС (Fedora, Debian).

... и **недостатки**:

- Отсутствие подсистемы имитационного Моделирования (симуляция моделей);
- «Линейность» диаграмм программной деятельности;
- Отсутствие поддержки ОС РВ Багет 3.0;
- Низкая эргономика интерфейсов (редакторы, формы, окна).

ТСАГ СПО: перспективы

Основные **перспективы** развития ТСАГ СПО:

- Разработка подсистемы имитационного моделирования;
- Разработка подсистемы анализа исходного кода (анализатор Си) и создание верификатора моделей;
- Интеграция в Eclipse;
- Поддержка стандартов SysML и SystemC;
- Проработка пользовательских интерфейсов;
- Поддержка ОС РВ Багет 3.0;
- Реинжиниринг исходного кода (исходный код → модель ТСАГ СПО).

Вместо заключения

SCADE Suite и TСАГ СПО решают задачу автоматизации разработки систем реального времени. Эти программы позволяют автоматизировать процесс моделирования, повысить надежность конечного исходного кода за счет верификации моделей высокого уровня, сгенерировать готовый для компиляции и сборки исходный код.

TСАГ СПО и SCADE Suite могут конкурировать, для этого в TСАГ СПО необходимо реализовать возможность симуляции (имитационного моделирования) моделей.

СПАСИБО ЗА ВНИМАНИЕ