

Утвержден

ЛРМТ.44469821.504105-01 34 01 - ЛУ

**ЗАКРЫТОЕ АКЦИОНЕРНОЕ ОБЩЕСТВО  
«ИТ В СУДОСТРОЕНИИ»**

**ТРЕХФАЗНЫЙ ДВУХСКОРОСТНОЙ НЕРАВНОВЕСНЫЙ  
ТЕПЛОГИДРАВЛИЧЕСКИЙ КОД  
«SERPENT»**

**РУКОВОДСТВО ОПЕРАТОРА**

**ЛРМТ.44469821.504105-01 34 01**

Листов 21

Инв. № подл.	Подл. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата
45-И-00	5	55-И-00		

1998

## АННОТАЦИЯ

В данном программном документе приведено руководство оператора по применению и эксплуатации теплогидравлического расчетного кода Serpent.

В данном программном документе в разделе «Назначение программы» указаны сведения о назначении программы и информация, достаточная для понимания функций программы и ее эксплуатации.

В разделе «Условия выполнения программы» указаны условия, необходимые для выполнения программы (минимальный состав аппаратных и программных средств и т.п.).

В разделе «Выполнение программы» указана последовательность действий оператора, обеспечивающих загрузку, запуск, выполнение и завершение программы, приведено описание функций, формата и возможных вариантов команд, с помощью которых оператор осуществляет загрузку и управляет выполнением программы, а также ответа программы на эти команды.

В разделе «Сообщения оператору» приведены тексты сообщений, выдаваемых в ходе выполнения программы, описание их содержания и соответствующие действия оператора (действия оператора в случае сбоя, возможности повторного запуска программы и т.п.).

Изм.	Лист	№ докум.	Подп.	Дата
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

## СОДЕРЖАНИЕ

1.	Назначение программы .....	4
2.	Условия выполнения программы.....	5
3.	Выполнение программы .....	6
3.1.	Установка программы .....	6
3.2.	Запуск программы .....	6
3.3.	Работа с программой .....	6
3.4.	Язык пользовательских файлов.....	7
3.4.1.	Правила при создании идентификаторов.....	7
3.4.2.	Запись численных констант.....	7
3.4.3.	Синтаксис файлов исходных данных .....	7
3.4.4.	Синтаксис управляющих файлов.....	9
3.5.	Построение расчетной схемы .....	15
3.5.1.	Принцип формирования расчетной схемы .....	15
4.	Сообщения оператору .....	17
5.	Перечень сокращений .....	20
6.	Лист регистрации изменений .....	21

Изм.	Лист	№ докум.	Подп.	Дата
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

## 1. НАЗНАЧЕНИЕ ПРОГРАММЫ

Теплогидравлический расчетный код (далее - РК) Serpent предназначен для проведения расчетов нестационарной гидродинамики двухфазных неравновесных потоков и ориентирован на моделирование динамики транспортных ЯЭУ.

РК основан на использовании негомогенной неравновесной модели. Для расчета применяется полунеявная схема. В расчете учитывается наличие нерастворимой газовой фазы.

Расчет нейтронно-физических процессов в коде производится на основе точечной модели кинетики реактора с учетом плотностного и температурного эффектов. Для моделирования транспортных установок предусмотрено формирование пространственного распределения объектов модели с возможностью динамической корректировки потенциальных силовых полей и углов поворота конструкции.

Формирование исходных данных осуществляется с использованием специально разработанного языка, который обрабатывается встроенным интерпретатором. В настоящем документе приводится описание языка и методов нодализации теплогидравлической схемы.

В РК Serpent предусмотрена возможность прерывания расчета с сохранением текущего состояния расчета и его последующим возобновлением, при этом допускается внесение изменений в логику управления. Состояние задачи записывается в файл сохранения по требованию пользователя в любой момент расчета и обозначается термином «Спасенное состояние».

РК Serpent позволяет производить обмен информацией в двухстороннем порядке с внешними программами в реальном времени, например, для моделирования отдельных элементов оборудования или моделирования пульта управления установкой.

Пользовательские файлы, содержащие условия расчета, представляют собой текстовые файлы, написанные в виде программы на языке кода Serpent в соответствии с разработанной пользователем нодализационной схемой. Файлы состоят из набора операторов, позволяющих описать связи между элементами, условия однозначности для каждого элемента нодализационной схемы и логику управления моделью – пользовательский сценарий. Расчетная информация выводится в файл результатов.

Изм.	Лист	№ докум.	Подп.	Дата
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

## 2. УСЛОВИЯ ВЫПОЛНЕНИЯ ПРОГРАММЫ

Теплогидравлический расчетный код Serpent должен устанавливаться на ПЭВМ с операционной системой семейства Windows. В системе должны быть установлены библиотеки Qt 4.8 и openAL 1.0 (32 бита). Также должны быть установлены библиотеки среды исполнения Visual Studio 2010 (32 бита).

Для установки программы необходима следующая минимальная конфигурация оборудования:

- ПЭВМ с процессором не ниже Intel Core i5 4 ядра 2000 МГц;
- свободное ОЗУ не менее 512 Мб;
- монитор не менее 800x600 точек;
- свободное место на НЖМД не менее 100 Мб;
- устройство ввода типа «мышь»;
- клавиатура.

Требования к ресурсам жесткого диска определяются исходя из объемов дисковой памяти, требуемой для хранения исполняемых и временных файлов программы.

Защита информации программы от несанкционированного доступа обеспечивается специальными средствами ОС.

Изм.	Лист	№ докум.	Подп.	Дата
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

### **3. ВЫПОЛНЕНИЕ ПРОГРАММЫ**

#### **3.1. Установка программы**

Перед началом процедуры необходимо произвести установку библиотеки openAL 1.0 и новее (32 бита) и библиотек времени исполнения Visual Studio 2010 (32 бита). На цифровом носителе эти библиотеки не поставляются. Для установки теплогидравлического расчетного кода Serpent необходимо распаковать установочный архив с цифрового носителя в целевую папку. Программа готова к использованию.

#### **3.2. Запуск программы**

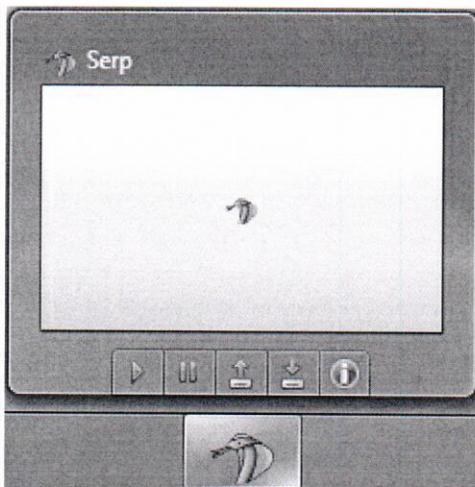
Запуск программы осуществляется с помощью файла agg.exe. При этом в рабочей директории должны содержаться все необходимые файлы комплекта, согласно Руководству системного программиста на теплогидравлический расчетный код Serpent

#### **3.3. Работа с программой**

Для работы теплогидравлического расчетного кода Serpent необходимо подготовить управляющие файлы и файлы исходных данных. Создание и правка этих файлов производится вручную с помощью текстового редактора.

При запуске файла agg.exe производится считывание пользовательских файлов и инициализация расчетной схемы. При этом создаются файлы-протоколы и файл результатов. Содержимое файла результатов формируется и дописывается в процессе расчета.

Управление кодом Serpent производится с помощью контекстного меню окна приложения (см. рисунок 3.1), доступного на панели задач с помощью наведения указателя мыши.



Изм.	Лист	№ докум.	Подп.	Дата
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

**Рисунок 3.1 - Интерфейс кода Serpent**

Запуск расчета производится с помощью кнопки «Пуск», при этом в названии окна начинает отображаться текущее время расчета. Остановка расчета производится с помощью кнопки «Стоп». Сохранение и восстановление состояния выполнения задачи производится с помощью кнопки «Спасти» и «Загрузить», соответственно.

### **3.4. Язык пользовательских файлов**

#### **3.4.1. Правила при создании идентификаторов**

- допустимые символы 0...9, буквы латинского алфавита a...z, A...Z, символ '\_';
- регистр символов учитывается;
- идентификатор может начинаться с любого допустимого символа, кроме цифры;
- длина идентификатора не должна превышать 18 символов.

#### **3.4.2. Запись численных констант**

Целочисленные значения

Пример допустимой записи числа 123:

123

Числа с плавающей точкой

Пример допустимой записи числа 123:

123.0

123.

Пример допустимой записи числа -0.4:

-0.4

-.4

Пример допустимой записи числа 123.4:

123.4

1234.e-1

.1234e3.

Операторы разделяются между собой символами: пробел, табуляция, перевод строки. В одной строке может быть записано несколько операторов.

Количество элементов массивов не должно превышать 100, если это не оговорено отдельно. Например, количество расчетных объемов (cell) в канале (ребро K) не должно превышать 100.

#### **3.4.3. Синтаксис файлов исходных данных**

Не допускается создание и использование пользовательских переменных, кроме объектов –

Изм.	Лист	№ докум.	Подп.	Дата
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

элементов расчетной модели.

Все типы элементов, доступных пользователю в коде, представляют собой классы. Классы имеют единый базовый элемент и принципиальную структуру.

Общее правило инициализации элемента расчетного кода:

*/ГруппаТипов/ТипОбъекта:ИмяОбъекта (Подключаемые объекты) Параметры {Вложенные объекты}*

Некоторые типы объектов не требует указания какого-либо параметра.

Если для выбранного типа элемента не требуется указание подключаемых объектов – пустые круглые скобки опускаются. Если для выбранного типа элемента не требуется указание вложенных объектов – следует оставить в конце пустые фигурные скобки. В некоторых случаях указывать группу типов не требуется, или требуется дополнительно определить тип объекта.

Для объектов разного типа допустимо использование совпадающего идентификатора.

Общее правило обращения к элементу:

*/ГруппаТипов/ТипОбъекта>ИмяОбъекта*

Если допустимо обращение к вложенным элементам, используется запись вида:

*/ГруппаТипов/ТипОбъекта>ИмяОбъекта<Номер>*

При указании констант значение берется в символы апострофа.

Все элементы перед использованием должны быть инициализированы.

Допустима инициализация при использовании. Инициализация объектов производится однократно.

Любая последовательность символов, не содержащая ни одного из перечисленных символов \ / \\$ [ ] { } '<>', игнорируется интерпретатором как комментарий. В данном руководстве для упрощения чтения листингов комментарии выделяются подчеркиванием волнистой линией.

Пример:

...

### Ячейка тип 1

```
/cell:Cell1 nchan '1' step '0.1' alfa '0.0' S '7.07e-4' Dg '0.03' tr '3.33e-4' xi '0.0' {}
```

...

### /array:Array1

```
{ Cell1[10] Перечисление других ячеек}
```

```
/rebro/K:RebroK1 (/uzel/D>UzelD1 /uzel/D>UzelD2 ) nchan 1
{ Array1 }
```

...

В приведенном примере последовательности символов «Ячейка тип 1», «nchan», «step» и т.д.

Изм.	Лист	№ докум.	Подп.	Дата
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

являются комментариями. Далее для сохранения читаемости кода будут всегда указаны комментарии перед параметрами («nchan», «step», ...). Их указание носит рекомендательный характер. Считывание параметров производится последовательно.

#### **3.4.4. Синтаксис управляющих файлов**

Операторы и операнды, в том числе символы скобок, должны быть разделены пробелами, табуляцией или переходом строки.

Разрешенные операторы приведены в таблицах 3.1 и 3.2.

Изм.	Лист	№ докум.	Подп.	Дата
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

Таблица 3.1 - Операторы сравнения

Бинарные операторы	
<	операция «меньше, чем»
>	операция «больше, чем»; операция обращения к объекту
<=	операция «меньше или равно»
>=	операция «больше или равно»
=	сравнение на равенство
+	логическая операция И
-	логическая операция ИЛИ
!=	операция сравнения на неравенство
&	операция побитового умножения (побитовое И). Если соответствующие биты операндов равны 1, результирующий бит 1, в противном случае 0. $  \begin{array}{r}  1\ 1\ 0\ 0 \\  \&\ 0\ 1\ 1\ 0 \\  =\ 0\ 1\ 0\ 0  \end{array}  $
^	операция побитового сложения (побитовое исключающее сравнение). Если соответствующие биты операндов равны 0 и 1, результирующий бит 1, в противном случае 0. $  \begin{array}{r}  1\ 1\ 0\ 0 \\  ^\ 0\ 1\ 1\ 0 \\  =\ 1\ 0\ 1\ 0  \end{array}  $
	операция побитового ИЛИ. Если хотя бы один из соответствующих битов операндов равен 1, результирующий бит 1, в противном случае 0. $  \begin{array}{r}  1\ 1\ 0\ 0 \\    0\ 1\ 1\ 0 \\  = 1\ 1\ 1\ 0  \end{array}  $
Унарные операторы	
!	операция логического отрицания

Изм.	Лист	№ докум.	Подп.	Дата
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

Таблица 3.2 - Операторы, действие которых совмещено с присваиванием

Бинарные операторы	
<code>*=</code>	операция умножения, совмещенная с присваиванием
<code>+=</code>	операция сложения, совмещенная с присваиванием
<code>-=</code>	операция вычитания, совмещенная с присваиванием
<code>/=</code>	операция деления, совмещенная с присваиванием
<code>&amp;=</code>	операция побитового умножения (побитовое И), совмещенная с присваиванием.
<code>^=</code>	операция побитового сложения (побитовое исключающее сравнение), совмещенная с присваиванием.
<code> =</code>	операция побитового ИЛИ, совмещенная с присваиванием.
<code>=</code>	операция присваивания

В управляемом файле разрешено использование условных операторов без ограничения глубины вложения. Синтаксис:

```
if {УСЛОВИЕ}
ОПЕРАТОРЫ
endif
```

Если УСЛОВИЕ возвращает логическое значение ИСТИНА (любое ненулевое значение) выполняется блок ОПЕРАТОРЫ.

Для случая, если блок ОПЕРАТОРЫ содержит только один оператор, допустима запись условия в виде:

*ОПЕРАТОР {УСЛОВИЕ}*

Объявление переменных производится при первом использовании.

Перечень доступных функций приведен в таблицах 3.3 и 3.4. В качестве аргумента функции может быть использована только предварительно определенная переменная, но не константа.

Изм.	Лист	№ докум.	Подп.	Дата
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

Таблица 3.3 - Доступные математические функции

Функция	Описание
sign(x)	возвращает знак аргумента
abs(x)	возвращает абсолютное значение аргумента
cos(x)	функция косинуса. Угол (аргумент) задается в радианах
sin(x)	функция синуса. Угол (аргумент) задается в радианах
exp(x)	экспоненциальная функция
lg(x)	возвращает значение десятичного логарифма
ln(x)	возвращает значение натурального логарифма
Pow(x,y)	возвращает значение числа x, возведенного в степень y
sqrt(x)	возвращает неотрицательное значение квадратного корня
floor(x)	находит наибольшее целое, не превышающее значения x
max(x,y)	возвращает максимальное из двух значений
min(x,y)	возвращает минимальное из двух значений
mod(x,y)	возвращает остаток от деления нацело x на y

где x, y – численные переменные.

Таблица 3.4 - Встроенные функции теплофизических свойств воды и пара

Функция	Описание
Ew_P(P)	энталпия воды на линии насыщения, как функция от давления, кДж/кг
Ep_P(P)	энталпия пара на линии насыщения, как функция от давления, кДж/кг
PfEw(E)	давление, при котором пароводяная смесь превращается в воду, МПа
PfEp(E)	давление, при котором пароводяная смесь превращается в пар, МПа
T1fE(P,E)	температура воды, как функция давления и энталпии, °C
T2fE(P,E)	температура пара, как функция давления и энталпии, °C

где P – абсолютное давление, МПа; E – энталпия, кДж/кг.

Изм.	Лист	№ докум.	Подп.	Дата
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

Последовательность символов «\\» + «пробел», заставляет интерпретатор пропустить дальнейшее содержимое строки - комментарий. Первая и последняя строки в управляющем файле не должны быть комментариями.

Пример: На каждом шаге вычислить значение по следующей зависимости

$$\text{X} = \begin{cases} \frac{\text{Pi} \cdot \text{Diam}^2}{4}, & \text{X} \leq 1 \\ \frac{\text{Pi} \cdot \text{Diam}^2}{4} \cdot \left( \frac{(\text{X}-1)}{100} + 1 \right), & \text{X} > 1 \end{cases} \quad 0, \text{X} > 3,$$

где t – текущее расчетное время модели, Diam = 0.005.

```
...
\\ Инициализация переменных
Pi = 3.1416
Diam = 5.e-3
a1 = 2.
a2 = time
X = 0
if {time<3}
  X = Pi.
  X = Pow (Diam, a1)
  X /= 4
  if { time > 1. }
    a2 -= 1
    a2 /= 100
    a2 += 1
    X *= a2
  endif
endif
...

```

Для взаимодействия с элементами расчетной модели могут использоваться как переменные, созданные с помощью управляющих элементов, так и прямое обращение к расчетным параметрам элементов.

Общее правило обращения к параметру:

*/ГруппаТипов/ТипОбъекта/ИмяОбъекта 'ИмяПараметра'*

К параметру вложенного элемента:

*/ГруппаТипов/ТипОбъекта/ИмяОбъекта<Номер> 'ИмяПараметра'*

Пример:

*G1 = /rebro/K/Pipe <1> 'Gwater'*

Изм.	Лист	№ докум.	Подп.	Дата
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

Значению переменной G1 присваивается значение расхода воды в первой расчетной ячейке ребра типа К с именем Pipe.

Запись вида

`# DEF1 DEF2`

называется директивой и указывает интерпретатору, при встрече в тексте последовательности символов DEF1 – заменить ее на последовательность символов DEF2.

Существует возможность создания пользовательских функций. При этом действуют следующие ограничения:

- все переменные имеют глобальную область определения;
- функция не имеет передаваемых параметров;
- функция не может возвращать значение.

Правило описания и вызова функции

*function ИмяФункции*

*ОПЕРАТОРЫ*

*endFunction*

Описание функции может быть расположено в произвольном месте кода вне операторов условия, до первого обращения к функции.

Вызов функции

*call ИмяФункции*

Изм.	Лист	№ докум.	Подп.	Дата
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

Пример:

Требуется в интервале времени от 0 до 2 секунд выполнять умножение переменных a, b и с на 2, а после на 4.

...

*// Определяем функцию*

*function F1*

*a \*= d*  
*b \*= d*  
*c \*= d*

*endFunction*

*if { time <= 2. }*  
*d = 2.*  
*endif*

*if { time > 2. }*  
*d = 4.*  
*endif*

*call F1*

...

### 3.5. Построение расчетной схемы

#### 3.5.1. Принцип формирования расчетной схемы

Для создания расчетной модели необходимо построить расчетные Графы (контуры). Расчетный Граф образован вершинами – узлами и ребрами, которые определяют связи между вершинами по массо- и теплопереносу. В дальнейшем условимся подразумевать под ребрами – элементы, формирующие связи по переносу теплоносителя.

Для формирования тепловых связей между элементами Графа используется элемент ТВЭЛ, моделирующий твердотельный элемент с энерговыделением, в частном случае, нулевым. Тепловая связь может соединять элементы, находящиеся в разных Графах.

Моделирование кинетики ядерного реактора для задания энерговыделения производится с помощью элемента Кинетика.

Для формирования элементов, образующих гидравлический тракт, таких как местное сопротивление, клапан, насос и т.п., используются гидравлические элементы, которые можно связывать с ребрами.

Для использования параметров вышеперечисленных элементов (например, получения расхода воды в элементе типа ребро, задания нового тепловыделения в элементе типа Кинетика или из-

Иzm.	Лист	№ докум.	Подп.	Дата
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

менения режима работы клапана) используются специальные управляющие типы, которые могут быть связаны с любым доступным параметром.

Для элементов группы типов Ребро и группы типов Узел учитывается тепловое объемное расширение. Расчетный объем элемента  $V$  принимается равным геометрическому объему  $V_0$ , определяемому при инициализации, при значении средней температуры теплоносителя, заключенного в элементе  $T_{gm} 20^{\circ}\text{C}$ . При отличной температуре расчетный объем вычисляется по формуле:

$$\Delta V = V_0 \cdot (1 + 1.2 \cdot 10^{-5} \cdot 3 \cdot (T - 20)),$$

которая описывает термическое изменение объема для стали.

Иzm.	Лист	№ докум.	Подп.	Дата
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

## 4. СООБЩЕНИЯ ОПЕРАТОРУ

При возникновении ошибок в процессе работы, выполнение программы прерывается и пользователю выдается сообщение о характере ошибки.

Некоторые из возможных сообщений об ошибках:

«*Model isn't created*» – не удается создать расчетную модель. Сообщение свидетельствует о наличии синтаксических ошибок.

«*Error in initModel*» – ошибка при инициализации расчетной модели. Например, выдается в случае присвоения различным параметрам одного номера информационного канала.

«*Error in model*» – расчетная ошибка в модели. «Развал» задачи по расчетным параметрам (достигнута температура кристаллизации воды и т.п.)

«*Error in initCommand*» – синтаксическая ошибка в управляющем файле.

«*Буфер %размер\_буфера% мал для трубы Data*» – выдается, если размер объекта для обмена данными недостаточен.

«*Error in Main Thread*» – системная ошибка

«*Command isn't created*» – отсутствует описание объекта command.

«*Error in command*» – расчетная ошибка в управляющем файле.

«*Caught Math Error!*» – ошибка при выполнении математических действий.

«*Cannot allocate memory!*» – системная ошибка

«*массив arrayToView уже существует и номер %номер\_канала% для него слишком велик*» – ошибка выдается в случае, если номер\_канала управляющего элемента, присоединенного к объекту в command, больше максимального номера канала управляющего элемента, используемого в модели.

Изм.	Лист	№ докум.	Подп.	Дата
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

«Вектор направления не нормирован» – вектор направляющих косинусов не нормирован

- сумма квадратов направляющих косинусов не равна единице. Ошибка не является критической, и расчет будет продолжен.

При незапланированном прекращении выполнения программы в большинстве случаев описание ошибки выводится в файл serpent.out, либо command.out, причем анализ содержимого последних строк файла позволяют установить место нахождения указанной ошибки.

Некоторые ошибки, выводимые в serpent.out:

«error[встретился неожиданный символ] %символ%» – ошибка синтаксиса: использован недопустимый символ.

«error[объект/%тип\_объекта>имя\_объекта% не существует]» – попытка использовать неинициализированный объект.

«error[объект/%тип\_объекта:имя\_объекта% уже существует]» – попытка повторной инициализации объекта.

«Existing 1 opened '{' in end of file» – отсутствует закрывающая фигурная скобка.

«B ArrayCapacity rebro/K/%имя\_ребра% нет вектора ячеек» – ошибка инициализации элемента типа ребро К – не найден элемент типа array.

«viewNum= номер\_канала existing twice» – номер канала view совпал с номером зарезервированного информационного канала.

«error :Already exist view{%номер\_канала%}» – повторное использование номера информационного канала.

«error :Already exist sign{%номер\_канала%}» – повторное использование номера информационного канала.

«интегральное ребро вставляется только в кинетику» – при перечислении подключенных элементов для элементов типа Граф и model ошибочно указан элемент типа ребро.

«error:parameter %имя\_параметра% don't exist» – при инициализации элемента управляющего типа указано не существующее для связываемого типа имя параметра. Сообщение носит информационный характер, номеру информационного канала не сопоставляется никакое значение параметра.

«error %номер\_ячейки% ячейки твэла присоединяется с внутренней стороны к крайней (виртуальной) ячейке» – количество участков в твэле превышает количество ячеек связываемого ребра.

«/model>%имя\_модели% не имеет объектов в своем составе» – модель должна иметь в сво-

Изм.	Лист	№ докум.	Подп.	Дата
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

ем составе хотя бы один объект типа graf/G.

Некоторые ошибки, выводимые в command.out:

«*error sim = %символ% nString = %номер строки%*» – в тексте управляющего файла на строке с указанным номером обнаружена синтаксическая ошибка после указанного символа.

«*error! Слишком длинное название параметра→%имя\_параметра%*» – указанное имя параметра превышает 18 символов.

«*error! error→лишний endif*» сообщение о лишнем операторе endif.

«*error ! У функции должна быть обратная скобка*» – отсутствует закрывающая круглая скобка.

Изм.	Лист	№ докум.	Подп.	Дата
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

## 5. ПЕРЕЧЕНЬ СОКРАЩЕНИЙ

ЯЭУ	ядерная энергетическая установка
ПЭВМ	персональная электронно-вычислительная машина (персональный компьютер)
ОЗУ	оперативное запоминающее устройство (оперативная память)
НЖМД	накопители на жестких магнитных дисках (жёсткий диск)
ОС	операционная система
ТВЭЛ	тепловыделяющий элемент

Изм.	Лист	№ докум.	Подп.	Дата
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

## **6. ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ**

<i>Изм.</i>	<i>Лист</i>	<i>№ докум.</i>	<i>Подп.</i>	<i>Дата</i>
<i>Инв. № подл.</i>	<i>Подп. и дата</i>	<i>Взам. инв. №</i>	<i>Инв. № дубл.</i>	<i>Подп. и дата</i>