

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ  
ВЫСШЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ

**“САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ  
УНИВЕРСИТЕТ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ,  
МЕХАНИКИ И ОПТИКИ”**

Факультет Компьютерных технологий и управления

Направление подготовки 211000.68 «Конструирование и технология электронных средств»

Магистерская программа 21100002 «Технологии и инструментальные средства проектирования электронных систем»

Квалификация (степень) магистр

Специальное звание магистр-инженер

Кафедра ПБКС Группа 6159

# **МАГИСТЕРСКАЯ ДИССЕРТАЦИЯ**

**на тему**

Разработка программного комплекса автоматизированного  
расчета теплового режима печатной платы

Автор магистерской диссертации Ягодаров А. Е. (подпись)  
( Фамилия, И., О. )

Научный руководитель доцент каф. ПБКС, к.т.н., Крылов Б. А. (подпись)  
( Фамилия, И., О. )

Руководитель магистерской программы профессор, д.т.н. Ткалич В.Л. (подпись)  
( Фамилия, И., О. )

**К защите допустить**

Зав. кафедрой профессор, д.т.н., Гатчин Ю. А. (подпись)  
( Фамилия, И., О. )

“        ”        2013 г.

Санкт-Петербург, 2013 г.

Магистерская диссертация выполнена с оценкой \_\_\_\_\_

Дата защиты “ \_\_\_\_ ” \_\_\_\_\_ 2013 г.

Секретарь ГАК \_\_\_\_\_ Лещикова Н.Г.

Листов хранения \_\_\_\_\_

**Чертежей хранения** \_\_\_\_\_

Федеральное государственное бюджетное образовательное учреждение  
высшего профессионального образования  
Санкт-Петербургский национальный исследовательский университет  
информационных технологий, механики и оптики

**АННОТАЦИЯ  
ПО МАГИСТЕРСКОЙ ДИССЕРТАЦИИ**

Студента Ягодарова Андрея Евгеньевича  
( Фамилия, И., О. )

Факультет Компьютерных технологий и управления

Кафедра ПБКС Группа 6159

Направление 211000.68 «Конструирование и технология электронных средств»

Академическая степень магистр техники и технологии

Наименование темы: Разработка программного комплекса автоматизированного  
расчета теплового режима печатной платы

Научный руководитель доцент каф. ПБКС, к.т.н., Крылов Б. А.  
( Фамилия, И., О., ученое звание, степень )

Консультант \_\_\_\_\_  
( Фамилия, И., О., ученое звание, степень )

**КРАТКОЕ СОДЕРЖАНИЕ МАГИСТЕРСКОЙ ДИССЕРТАЦИИ  
И ОСНОВНЫЕ ВЫВОДЫ**

объем 93 стр., графический материал 12 стр., библиография 12 наим.

⌚ Направление и задача исследований

Направление исследований: разработка программного комплекса автоматизированного  
расчета теплового режима печатной платы. Входными данными для программного  
комплекса является экспортированная из САПР Altium Designer 10 конструкторская  
документация печатной платы. В функции программного комплекса должны входить:  
обработка входных данных, расчет теплового режима печатной платы на основе входных  
данных, отображение рассчитанного теплового режима пользователю. Задача исследований:  
анализ методов расчета теплового режима печатной платы, анализ тепловых и  
математических моделей печатной платы; выбор способа расчета теплового режима  
печатной платы, автоматизация выбранного способа расчета.

⌚ Проектная или исследовательская часть (с указанием основных методов исследований,  
расчетов и результатов)

В данной магистерской диссертации (МД) приведены основные положения разработки  
программного комплекса для автоматизации процесса расчета теплового режима печатной

платы на основе документов конструкторской документации САПР Altium Designer 10 (документы PCB Editor, экспортированные в виде Altium PCB ASCII (\*.PcbDoc)). Итог работы программы — температурная карта элементов печатной платы. В процессе работы над МД были исследованы методы расчета теплового режима и виды тепловых моделей печатных плат. Проектирование программного комплекса проводилось с применением объектно-ориентированного подхода. Вследствие чего разработаны UML-диаграммы, на основе которых была написана программа. В ходе разработки программы использовалась среда разработки Eclipse. Программа написана на языке программирования Java.

⌚ Новизна полученных результатов \_\_\_\_\_

⌚ Является ли работа продолжением курсовых проектов (работ), есть ли публикации \_\_\_\_\_

**Практическая ценность работы. Рекомендации по внедрению** \_\_\_\_\_

Выпускник \_\_\_\_\_  
(подпись)

Научный руководитель \_\_\_\_\_  
(подпись)

“ \_\_\_\_\_ ” \_\_\_\_\_ 2013 г.

## **СОДЕРЖАНИЕ**

## **ОСНОВНЫЕ УСЛОВНЫЕ ОБОЗНАЧЕНИЯ**

РЭА — радиоэлектронная аппаратура;

ЭВМ — электронная вычислительная машина;

САПР — система автоматизированного проектирования;

ПП — печатная плата;

UML – unified modeling language – унифицированный язык моделирования;

IDE – integrated development environment – интегрированная среда разработки;

ПО — программное обеспечение.

## ВВЕДЕНИЕ

Каждый объект на нашей планете (включая саму планету) излучает и поглощает тепло. Процессы поглощения и излучения тепла определяют температуру объекта. Тепловой режим объекта — совокупность факторов, определяющих распределение тепловых потоков на объекте.

Печатная плата с размещёнными на ней элементами является РЭА. Каждый элемент такой печатной платы обладает своей температурой. Совокупность температур всех элементов, из которых собран РЭА, то есть его температурное поле, характеризует тепловой режим аппарата[1].

Нормальное функционирование РЭА возможно лишь при условии поддержания температур ее элементов в определенных пределах. Изменение теплового режима оказывает влияние на характеристики элементов и может привести к возникновению физико-химических процессов, выводящих элемент из строя. При этом дестабилизирующими тепловыми воздействиями являются рассеиваемые при работе элементов мощности, изменения температуры внешней среды и тепловые потоки от окружающих прибор объектов. Чтобы тепловой режим РЭА был нормальным, необходимо выполнение двух условий [1]:

- 1) температура всех элементов РЭА в условиях эксплуатации заключена в диапазоне допустимых температур этих элементов;

- 2) температура всех элементов РЭА такова, что будет обеспечена работа РЭА с заданной надёжностью.

Для обеспечения нормального теплового режима РЭА требуется рассчитать текущий тепловой режим. Анализ рассчитанного теплового режима позволит разработчикам внести коррективы в проектируемую плату с целью рационального размещения элементов и актуального выбора принудительного охлаждения/нагрева печатной платы.

В современных условиях многие предприятия не в состоянии содержать специальные подразделения для проведения расчетов теплового режима печатной платы, и выполнять их приходится разработчику вручную. Выполнение такого расчета вручную занимает много временных и трудовых затрат разработчика.

Исходя из вышесказанного, задача автоматизации процесса расчета теплового режима печатной платы актуальна. Решение этой задачи увеличит временные ресурсы разработчика печатных плат. Это позволит ему не задерживаться на рутинных расчетах, а уделять свое время на более творческие задачи. В итоге это позволит предприятиям выпускать более качественные продукты.

В первой главе рассмотрены методы расчета теплового режима печатной платы, проведен обзор существующих систем автоматизации анализа теплового режима печатной платы, приведена постановка задачи данной работы.

Во второй главе детально рассмотрено проектирование разрабатываемого программного комплекса автоматизированного расчета теплового режима печатной платы.

В третьей главе описано сравнительное тестирование разработанного программного комплекса.

В приложении предоставлено руководство пользователя по разработанному программному комплексу.



## **1. АНАЛИТИЧЕСКАЯ ЧАСТЬ**

### **1.1 Анализ методов расчета теплового режима РЭА**

В связи с наличием острой проблемы обеспечения теплового режима РЭА, были развернуты широкие исследования как за рубежом, так и в нашей стране для решения этой проблемы. За последнее время сформировалась целая отрасль науки и техники, в которой работает большое число специалистов, занимающихся обеспечением тепловых режимов различных приборных комплексов [3].

В последние годы исследования тепловых режимов приборов стали развиваться в направлении создания методов теплового проектирования, базирующихся на широком применении ЭВМ, и разработки соответствующих частей САПР. Проектирование проводится на основе моделирования теплового режима объекта. Для реализации моделирования необходимо разрабатывать физико-математические модели объектов, вычислительные методы, обеспечивающие возможность проведения расчетов, а также соответствующее программное обеспечение ЭВМ [3].

Конструирование элементной базы микроэлектронной аппаратуры выполняется отдельно от конструирования устройств. Поэтому на этом этапе определяются допустимые температуры корпусов элементов, которые в дальнейшем являются исходными данными для проектирования систем охлаждения устройств [3].

В процессе работы элемент или устройство подвергаются внешним и внутренним тепловым воздействиям. Внешние тепловые воздействия вызваны изменениями температуры окружающей среды и тепловым потоком, поступающим к границе объекта, внутренние — его собственными тепловыделениями. Под влиянием этих тепловых воздействий происходит изменение теплового режима объекта [3].

Источниками теплоты в РЭА являются различные электрические устройства и отдельные радиодетали. Потребляемая радиодетальными электрическая энергия преобразуется в них в электромагнитную, механическую, тепловую и другие виды энергии. Иными словами, часть потребляемой радиодетальными энергии превращается в полезные сигналы, остальная — в теплоту [5].

Процедура анализа теплового режима базируется на моделировании объекта. Для реализации моделирования необходимо разрабатывать физико-математическую модель объекта, вычислительные методы, обеспечивающие возможность проведения расчетов, а также соответствующее программное обеспечение для ЭВМ. Используемые модели должны быть реализуемыми математически и в то же время адекватными изучаемым процессам, протекающим в конкретных объектах [3].

### **1.1.1 Процессы тепломассообмена в природе**

Тепломассообмен — раздел физики, в котором рассматриваются процессы переноса теплоты (энергии) и массы (вещества).

Явления теплообмена связаны с необратимым переносом энергии из одной части пространства в другую и вызваны разностью температур, а явления массообмена — с перемещением вещества из одной части пространства в другую и вызваны разностью концентраций [5].

Если теплообмен и массообмен сопутствуют друг другу и их приходится рассматривать во взаимосвязи, то имеем дело с тепломассообменом. Когда явления теплообмена и массообмена мало влияют друг на друга, их можно рассматривать порознь; иногда имеет место только какое-нибудь одно явление. Соответствующие процессы в этих случаях называют процессами тепло- и массообмена, теплообмена, массообмена [5].

Различают три вида переноса энергии в виде теплоты:

теплопроводность, конвекцию и тепловое излучение.

Теплопроводность — молекулярный перенос теплоты в сплошной среде, вызванный разностью температур.

Конвекция — процесс переноса теплоты при перемещении макроскопических объемов жидкости или газа из области с одной температурой в область с другой; при этом перенос теплоты неразрывно связан с переносом вещества. Процессы конвекции сопровождаются теплопроводностью; этот совместный процесс называется конвективным теплообменом.

Тепловое излучение — процесс переноса теплоты, обусловленный превращением внутренней энергии вещества в энергию излучения, переносом ее в виде электромагнитных волн и поглощением веществом.

Перенос вещества происходит с помощью диффузии и конвективного массообмена. Диффузия — молекулярный перенос вещества в среде, вызванный разностью концентраций (концентрационная диффузия), температур (термодиффузия) или давлений (бародиффузия). Конвективный массообмен — перенос вещества, вызванный совместным действием конвективного переноса вещества и молекулярной диффузии.

Некоторые примеры, связанные с переносом теплоты и массы в природе:

- ⌚ теплообмен человека со средой;
- ⌚ перенос теплоты из жилища в окружающую среду (через окна, двери, стены) и, наоборот, из среды в жилище;
- ⌚ перенос энергии от Солнца к Земле;
- ⌚ различные способы переработки вещества и продуктов — все эти процессы связаны с переносом энергии и вещества в пространстве (тепломассообмен).

Такие процессы, как испарение, сушка, образование облаков, представляют собой целый комплекс явлений тепломассообмена, сопровождающихся фазовыми превращениями.

### 1.1.2 Исследование теплового режима РЭА

РЭА представляет собой систему многих тел с внутренними источниками тепла. Температурное поле аппарата зависит от мощности и распределения источников тепла, конструкции, режима работы аппарата и его системы охлаждения, геометрических параметров, физических свойств материалов, из которых изготовлен аппарат, условий его эксплуатации.

Постановка задачи изучения теплового режима звучит следующим образом [1]: требуется найти температуру  $t_j$  какой-либо  $j$ -й области аппарата в зависимости от времени  $\tau$  и суммарной мощности  $P$  источников тепла при заданных условиях его эксплуатации (1.1).

$$t_j = t_j(\tau, P) \quad (1.1)$$

Температура  $t_j$  является аддитивной функцией температуры  $t_c$  окружающей среды РЭА и перегрева  $\vartheta_j$  по отношению к температуре среды, возникающего в результате действия всех источников тепла, сосредоточенных в аппарате. Для того, чтобы выделить влияние источников тепла на тепловой режим  $j$ -й области РЭА, зависимость (1.1) удобно представить в следующем виде:

$$\vartheta_j = t_j - t_c = \vartheta_j(\tau, P) \quad (1.2)$$

В установившемся тепловом режиме перегрев  $\vartheta_j$  не зависит от времени:

$$\vartheta_j = \vartheta_j(P) \quad (1.3)$$

Зависимость перегрева  $\vartheta_j$  от суммарной мощности  $P$  всех источников

тепла, действующих в аппарате, называется тепловой характеристикой  $j$ -й области аппарата.

Исследование тепловых режимов РЭА сводится к экспериментальному или аналитическому определению вида зависимостей (1.2) или (1.3).

Цель исследования теплового режима РЭА, что можно выделить из постановки задачи, состоит в определении температурного поля аппарата с учетом влияния различных конструктивных и физических параметров. При исследовании тепловых режимов РЭА следует придерживаться определенной последовательности. Исследование начинается с выявления признаков, определяющих класс изучаемого аппарата. Следующим этапом исследования является определение тепловой модели рассматриваемого класса аппаратов.

Тепловая модель РЭА или класса РЭА определяется в результате идеализации конструкций аппаратов рассматриваемого класса и протекающих в них процессов переноса тепла. При разработке тепловой модели учитываются наиболее существенные черты конструкции аппаратов и протекающих в них физических процессов. При переходе от класса аппаратов к тепловой модели принимается ряд допущений и ограничений, которые необходимо четко сформулировать и обосновать.

Математическая реализация тепловой модели (система уравнений, конечно-разностное представление и т. д.) называется математической моделью. Основное требование к тепловой модели может быть кратко сформулировано следующим образом: тепловая модель должна быть адекватна изучаемому явлению и реализуема математически.

Условия перехода от тепловой модели к математической должны быть тоже четко сформулированы и обоснованы. Необходимо, если это возможно на данном этапе исследования, выявить степень неточности, вносимую допущениями. Иногда такое обоснование возможно провести после того, как получено решение задачи.

В большинстве случаев математическая модель РЭА представляется в

виде уравнений, описывающих температурное поле модели. Решение этой системы может быть получено аналоговыми методами (электролитическая ванна, гидро- или электроинтеграторы, аналоговые вычислительные машины и т. д.), численными методами или аналитически. Выбор метода решения требует обоснования в каждом конкретном случае. Заметим, что при аналоговом или численном решении задачи производится последовательное изучение различных частных реализаций математической модели, отличающихся численными значениями исходных параметров. Для обобщения результатов исследования частных случаев и представления их в компактной графической или аналитической форме требуется применение определенных иногда весьма сложных приемов.

При аналитическом решении задачи окончательное выражение, определяющее температурное поле тепловой модели, получается в общем виде и содержит гораздо большую информацию об изучаемом явлении, чем численные данные для конкретного частного случая. Иными словами, отпадает необходимость обобщения результатов частных расчетов.

В большинстве случаев форма окончательного решения задачи оказывается громоздкой и мало пригодной для оценки влияния различных параметров аппарата на его тепловой режим или вообще неудобной для практического применения. Появляется необходимость в разработке более простой, хотя и менее точной методики. В этом случае полученная ранее методика может служить инструментом дальнейшего исследования.

Итак, исследование теплового режима РЭА можно упорядочить в следующую последовательность действий:

1. выявление признаков, определяющих класс изучаемого аппарата;
2. определение тепловой модели рассматриваемого класса аппаратов;
3. переход от тепловой модели к математической;
4. выбор метода решения математической модели;
5. решение математической модели.

### 1.1.3 Тепловые модели РЭА

РЭА можно рассматривать, как систему многих тел с сосредоточенными источниками тепловой энергии. Анализ температурных полей таких систем является весьма сложной задачей, решение которой выполняется приближенными методами. Исследователь пытается установить количественную зависимость между температурой ограниченного числа наиболее ответственных мест аппарата и существенными факторами, влияющими на процесс теплообмена. При экспериментальном решении задачи эта работа может проводиться непосредственно на радиоэлектронном аппарате. Аналитическое решение исключает такой подход, так как тепловые процессы в реальной конструкции аппарата, как правило, не поддаются математическому описанию из-за наличия большого числа основных и второстепенных факторов, влияющих на процесс. Поэтому необходим переход к тепловой модели РЭА.

В настоящее время получили развитие две группы тепловых моделей РЭА [1]. Характерной особенностью моделей первой группы является разделение всех поверхностей модели РЭА на отдельные условно изометрические участки. Например, при определении среднеповерхностной температуры нагретой зоны условно изотермическими считаются поверхность корпуса и вся поверхность нагретой зоны, состоящая из поверхностей элементов и части шасси, не занятой ими.

На рис. 1.1(а) схематически показан разрез радиоэлектронного аппарата и указаны значения температур, измеренных в разных точках корпуса, шасси и радиодеталей. На рис. 1.1(б) представлена тепловая модель того же аппарата, на основании которой определяются среднеповерхностные температуры его нагретой зоны и корпуса. При этом сложная по форме нагретая зона реального аппарата заменена прямоугольным параллелепипедом, поверхность которого рассматривается как изометрическая. Такое же допущение делается относительно температурного поля корпуса. Задача упрощается настолько, что

математическое описание процесса теплообмена в тепловой модели аппарата становится возможным и сравнительно несложным.

На рис. 1.2(а) схематически изображен разрез радиоэлектронного аппарата кассетной конструкции, на монтажных платах (кассетах) которого смонтированы модули, микросхемы, микромодули и т. д. Значения температур корпуса и отдельных точек нагретой зоны не указаны, но предполагается, как и в первом примере, что поля температур неравномерны. К такому радиоэлектронному аппарату можно применить тепловую модель рис. 1.2(б) и описать процессы переноса тепла от поверхности нагретой зоны к корпусу и далее в среду. В результате анализа получим средние поверхностные температуры корпуса и нагретой зоны.

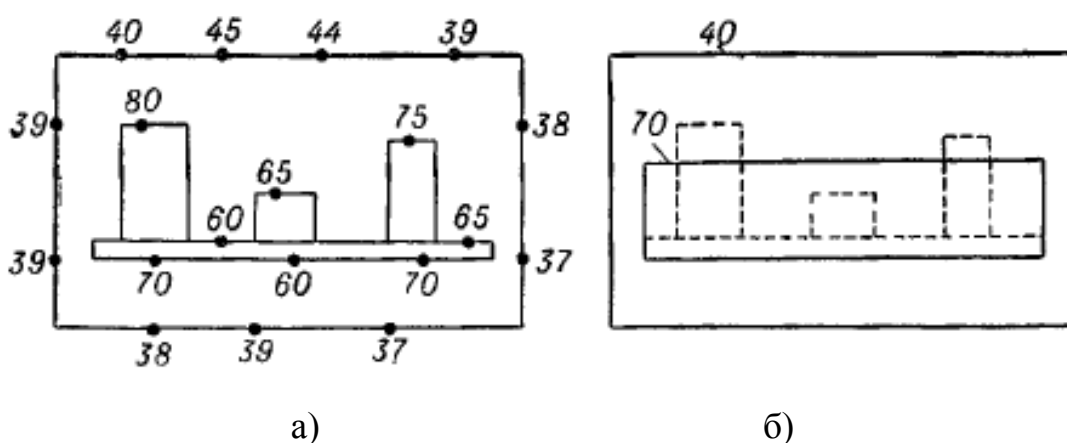


Рис. 1.1. Разрез простейшей конструкции РЭА (а) и его тепловая модель (б)





подробную разбивку, как это показано на рис. 1.2(в), и вместо семи изотермических поверхностей рис. 1.2(б) (одна поверхность — корпус, шесть — платы) ввести, например, 16 изотермических поверхностей. Информационные возможности тепловой модели рис. 1.2(в) значительно больше, чем модели рис. 1.2(б); изучая модель рис. 1.2(в), можно определить среднеповерхностные температуры шестнадцати поверхностей. Аналогичскую разбивку, при необходимости можно сделать в направлении, перпендикулярном плоскости чертежа, и тем самым еще более увеличить информационные возможности модели.

Процессы переноса тепла в тепловых моделях первой группы рассматриваются так, как если бы они протекали между изотермическими поверхностями.

В тепловых моделях, относящихся ко второй группе, нагретая зона РЭА, представляющая собой неоднородную систему многих тел, идеализируется в виде однородного тела. Свойства этого тела характеризуются значениями коэффициентов теплопроводности и теплоемкости.

На рис. 1.2(г) изображена тепловая модель второй группы для аппарата, изображение которого приведено на рис. 1.2(а). Нагретая зона аппарата представляет собой совокупность многих тел с дискретными источниками тепловой энергии. Информационные возможности такой тепловой модели весьма велики, так как ее исследование позволяет получить аналитическое выражение для поля температур нагретой зоны.

Особенности тепловых моделей РЭА определяют математический аппарат, применяемый для их анализа. Тепловые модели первой группы исследуются при помощи так называемого метода тепловых схем, которые позволяют описать процессы переноса тепла в РЭА при помощи системы неоднородных нелинейных алгебраических уравнений. Для изучения тепловых моделей второй группы применяются дифференциальные уравнения теплопроводности. При исследовании теплового режима РЭА сложных

конструкций тепловая модель аппарата может содержать в себе элементы обеих указанных групп моделей. При этом отдельные части сложного РЭА представляются в виде условно изотермических поверхностей, другие — в виде однородных тел.

#### 1.1.4 Математические модели тепловых моделей РЭА

Отличительной особенностью моделирования процессов теплообмена, осуществляемого при проектировании, является необходимость рассмотрения процесса одной и той же физической природы для всего прибора или приборного комплекса. Тепловая модель системы тел в общем случае представляет собой комбинацию многочисленных областей сложной конфигурации с источниками и стоками теплоты, в которых движутся потоки теплоносителей (газов или жидкостей). Наиболее полная математическая модель теплового режима объекта записывается в виде системы многомерных нестационарных уравнений теплопроводности для твердых тел (1.4) и уравнений энергии для потоков теплоносителей (1.5).

$$c_i \rho_i \frac{\partial T_i}{\partial \tau} = \nabla(\lambda_i \nabla T_i) + q_{Vi}, i=1, \dots, I. \quad (1.4)$$

$$c_l \rho_l \left( \frac{\partial U_l}{\partial \tau} + \vec{v} \nabla U_l \right) = \nabla(\lambda_l \nabla U_l), l=1, \dots, L. \quad (1.5)$$

В уравнениях (1.4), (1.5) использованы следующие обозначения:  $T_i$ ,  $U_l$  — температуры твердых тел и теплоносителей;  $\tau$  — время;  $c$  — удельная теплоемкость;  $\rho$  — плотность;  $\lambda$  — теплопроводность;  $q_v$  — объемная плотность теплового потока;  $v$  — скорость движения теплоносителя;  $I$ ,  $L$  — число тел и потоков теплоносителей.

Реализация такой полной модели затруднительна даже с применением

современных ЭВМ, поскольку число элементов нижнего уровня иерархии составляет обычно от нескольких сотен до нескольких десятков тысяч. Возникающие трудности связаны как с проблемой выбора метода решения и объемами требуемых затрат машинного времени и памяти, так и с объемом исходной информации, входящей в полную модель.

На любом из уровней иерархии задач анализа является определение тех характеристик температурных полей подсистем данного уровня, которые необходимы для нахождения конструктивных параметров, соответствующих этому уровню (например, среднеобъемных температур отдельных областей, средних температур различных поверхностей). Внутренняя структура подсистем учитывается через их обобщенные интегральные характеристики, зависящие от конструктивных параметров данного уровня.

В случае, когда ограничиваются анализом только среднеобъемных и среднеповерхностных температур тел, среднерасходных температур теплоносителей, их расчет проводится на основе моделей с сосредоточенными параметрами. Кроме того, довольно часто для некоторых тел, групп тел или потоков теплоносителей бывает необходимо рассматривать одномерные распределения характерных температур (например, распределения по длине осредненных в поперечном сечении температур). В этой ситуации используются одномерные модели.

Обычно наступает момент, когда для увеличения степени детализации описания теплового режима приходится для некоторых компонентов системы рассматривать многомерные температурные поля и проводить анализ на основе многомерной модели. При определении пространственных температурных полей метод поэтапного моделирования используется в различных вариантах. Например, можно сначала преобразовать исходную многомерную модель в модель с сосредоточенными параметрами или одномерную модель и определить соответствующие осредненные характеристики температурных полей, а затем решать многомерные уравнения для отдельных тел, подставляя в

их граничные условия уже известные осредненные значения тепловых потоков или температур окружающих тел и теплоносителей. При таком подходе многомерность модели фактически обеспечивается путём решения многомерных задач для одиночных тел. Особенностью других вариантов является использование тех или иных промежуточных моделей, в которых для нескольких тел одновременно рассматриваются многомерные температурные поля. Таким образом, в этом случае приходится решать многомерные задачи для систем тел.

В качестве укрупненной модели системы неупорядоченно расположенных тел и потоков теплоносителей обычно применяют модель с сосредоточенными параметрами в которой состояние элементов характеризуется не пространственным распределением температуры, а одним или несколькими средними значениями температуры. Рассмотрим переход от «полной» модели (1.4), (1.5) к математической модели, представляющей систему алгебраических (для стационарных задач) или обыкновенных дифференциальных (для нестационарных) уравнений.

Применим к уравнениям (1.4), (1.5) оператор осреднения по объему

$$I[f(\vec{x})] = \frac{1}{V} \int_V f(\vec{x}) dV = f_V, \vec{x} = (x, y, z) \quad (1.6)$$

Используя теорему Грина[3], получаем для твердых тел

$$C_i \frac{dT_{iV}}{d\tau} = \int_{S_i} \lambda_i \text{grad } T_i \vec{n} dS + P_i, i=1, \dots, I, \quad (1.7)$$

а для потоков теплоносителей

$$C_l \frac{dU_{lv}}{d\tau} = \int_{S_l} \lambda_l \text{grad } U_l \vec{n} dS - \int_{S_l} c_l \rho_l U_l \vec{v} \vec{n} dS, l=1, \dots, L, \quad (1.8)$$

где  $C_i, C_l$  - полные теплоемкости;  $P_i$  - мощность;  $S_i, S_l$  - площади поверхностей, ограничивающих объемы  $V_i, V_l$ ;  $\vec{n}$  - орт внешней нормали к поверхности.

Введем обозначения:  $S'_l$  - площадь поверхности твердых тел, омываемых  $l$ -м потоком;  $S_l^{ex}, S_l^{exx}$  - площади поверхностей входного и выходного сечений  $l$ -го потока ( $S_l = S'_l + S_l^{ex} + S_l^{exx}$ ). Тогда, полагая, что кондуктивным переносом теплоты через  $S_l^{ex}$  и  $S_l^{exx}$  можно пренебречь по сравнению с конвективным, и вводя среднерасходные температуры теплоносителя  $U_l^{ex}, U_l^{exx}$  входящего и выходящего потоков, преобразуем (1.8) к виду

$$C_l \frac{dU_{lv}}{d\tau} = \int_{S_l} \lambda_l \text{grad } U_l \vec{n} dS - c_l G_l (U_l^{exx} - U_l^{ex}), \quad (1.9)$$

где  $G_l$  - массовый расход теплоносителя.

Уравнения (1.7) и (1.9) выражают в интегральной форме закон сохранения энергии для  $i$ -го тела и  $l$ -го потока теплоносителя. Следующий шаг перехода к модели с сосредоточенными параметрами заключается в задании выражений для тепловых потоков, проходящих через поверхности  $S_i$  и  $S'_l$ . Во многих случаях оказывается возможным описать теплообмен между телами и между телами и потоками с помощью тепловых проводимостей.

Разобьем поверхность каждого  $i$ -го тела на участки  $S_{i,n}$  ( $n=1, \dots, N_i$ ) таким образом, чтобы тепловой поток между  $n$ -м участком поверхности  $i$ -го тела и  $m$ -м участком поверхности  $j$ -го тела мог быть выражен в виде

$$P_{ij} = \sigma_{ij} (T_{iS,n} - T_{jS,m}), \quad (1.10)$$

а тепловой поток  $P_{il}$ , передаваемый теплоносителю, записан так

$$P_{il} = \sigma_{il} (T_{iS,n} - U_{lV}), \quad (1.11)$$

где  $T_{iS,n}$ ,  $T_{jS,m}$  — средние температуры рассматриваемых участков поверхностей;  $\sigma_{ij}$ ,  $\sigma_{il}$  — тепловые проводимости между соответствующими участками поверхностей тел  $i$  и  $j$ , между участком поверхности тела  $i$  и потоком теплоносителя  $l$ .

Заметим, что тепловая проводимость  $\sigma_{il}$  может быть определена также по отношению к начальной ( $T_{iS,n} - U_l^{ex}$ ) разности температур тела и потока.

Введение тепловых проводимостей и запись тепловых потоков в форме (1.10), (1.11) является допущением, погрешность которого можно снизить, увеличивая число выделяемых участков поверхностей.

Подставив в (1.7), (1.9) выражения для тепловых потоков на границах, придем к системе обыкновенных дифференциальных уравнений относительно осредненных температур

$$C_i \frac{dT_{iV}}{d\tau} = \sum_{n=1}^{N_i} \left[ \sum_{j=1}^I \sigma_{ij} (T_{jS,m} - T_{iS,n}) + \sum_{l=1}^L \sigma_{il} (U_{lV} - T_{iS,n}) \right] + P_i, \quad i=1, \dots, I; \quad (1.12)$$

$$C_l \frac{dU_{lV}}{d\tau} = \sum_{i=1}^I \sigma_{il} (T_{iS,n} - U_{lV}) - c_l G_l (U_l^{ex} - U_l^{ex}), \quad l=1, \dots, L. \quad (1.13)$$

В уравнения (1.12), (1.13) входят следующие неизвестные величины: для твердых тел — среднеобъемная температура  $T_{iV}$  и средние температуры  $T_{iS,n}$  участков поверхности, выделенных для задания тепловых потоков к

окружающим телам и теплоносителям; для теплоносителей — среднеобъемная температура  $U_{iv}$  и среднерасходная температура в выходном сечении  $U_l^{ex}$ .

Температура  $U_l^{ex}$  либо задается, если поток входит в систему извне, либо рассчитывается по температурам  $U_k^{ex}$  выходящих потоков, смешением которых образуется данный поток:

$$G_l^{ex} U_l^{ex} = \sum G_k^{ex} U_k^{ex} . \quad (1.14)$$

#### 1.1.4.1 Модели с сосредоточенными параметрами

Укрупнённые модели позволяют рассчитывать интегральные характеристики процессов теплообмена (значения среднеобъемных и среднеповерхностных температур, средних тепловых потоков) с учетом взаимодействия между всеми подсистемами. Математическое описание процессов теплообмена осуществляют системами алгебраических и обыкновенных дифференциальных уравнений. В дальнейшем эти модели для расчета средних температур выделенных областей будем называть моделями с сосредоточенными параметрами, в отличие от моделей с распределенными параметрами, которые учитывают пространственные распределения температурных полей.

Практика показала, что для расчета средних температур по моделям с сосредоточенными параметрами целесообразно строить достаточно общие модели этого вида и разрабатывать для каждой из них универсальное программное обеспечение, позволяющее решать широкий круг конкретных технических задач.

Рассмотрим модель системы тел и потоков теплоносителей, находящихся во взаимном теплообмене. В исследуемом объекте выделим



области трех видов:  $N_T$  — твердых тел, температурные поля которых считаются равномерными ( $T_i(\tau), i=1, \dots, N_T$ );  $N_{\mathcal{K}}$  — объемов с протекающими в них жидкими или газообразными теплоносителями, имеющими среднерасходные температуры на входе  $U_l^{ex}(\tau)$  и на выходе  $U_l^{ex}(\tau)$  и осредненную по объему температуру  $U_l(\tau), l=1, \dots, N_{\mathcal{K}}$ ;  $N_{CP}$  — сред с известными температурами  $T_k^{CP}(\tau), k=1, \dots, N_{CP}$ .

В твердых телах и в объемах с теплоносителями могут действовать источники теплоты с полными мощностями  $P_i^T$  и  $P_i^{\mathcal{K}}$  соответственно. Твердые тела находятся в теплообмене друг с другом, а также с теплоносителями и со средами. Тепловые потоки  $P_{ij}^{T.T.}$ , приходящие к данному телу  $i$  от соседних тел, от теплоносителей  $P_{il}^{T.\mathcal{K}}$  и от сред  $P_{ik}^{T.CP.}$ , можно выразить через разности их средних температур в виде:

$$\begin{aligned} P_{ij}^{T.T.} &= \sigma_{ij}^{T.T.} (T_j - T_i), \\ P_{il}^{T.\mathcal{K}} &= \sigma_{il}^{T.\mathcal{K}} (U_l - T_i), \\ P_{ik}^{T.CP.} &= \sigma_{ik}^{T.CP.} (T_k^{CP} - T_i), \end{aligned} \quad (1.15)$$

где  $\sigma_{ij}^{T.T.}$ ,  $\sigma_{il}^{T.\mathcal{K}}$ ,  $\sigma_{ik}^{T.CP.}$  — тепловые проводимости между телами  $i$  и  $j$ , телом  $i$  и теплоносителем  $l$ , телом  $i$  и средой  $k$  соответственно.

Корректность зависимостей (1.15) во многом определяется выбранным способом определения величин тепловых проводимостей.

Используя выражения (1.9), (1.10) и (1.15), запишем в более простом виде уравнения теплового баланса для твердых тел и теплоносителей. Поскольку мощность  $P_i^T$ , выделяющаяся в теле  $i$ , расходуется на его нагрев и передается окружающим телам (Т.Т.), теплоносителям (Т.Ж.) и средам (Т.СР.),

то уравнение теплового баланса имеет вид

$$P_i^T = C_i^T \frac{dT_i}{d\tau} + \sum_{j=1}^{N_T} \sigma_{ij}^{T.T.} (T_i - T_j) + \sum_{l=1}^{N_{\text{Ж}}} \sigma_{il}^{T.\text{Ж}} (T_i - U_l) + \sum_{k=1}^{N_{\text{CP}}} \sigma_{ik}^{T.CP.} (T_i - T_k^{CP}), \quad (1.16)$$

$$i=1, \dots, N_T, l=1, \dots, N_{\text{Ж}}, k=1, \dots, N_{\text{CP}}$$

где  $C_i^T$  — полная теплоемкость тела  $i$ .

Тепловые потоки  $\sigma_{il}^{T.\text{Ж}} (T_i - U_l)$ , переданные от тел  $i$  к теплоносителю в объеме  $l$ , и мощность  $P_l^{\text{Ж}}$  расходуются на нагрев движущегося теплоносителя. Поэтому уравнение теплового баланса для потоков теплоносителей записывается в виде

$$P_l^{\text{Ж}} + \sum_{i=1}^{N_T} \sigma_{il}^{T.\text{Ж}} (T_i - U_l) = C_l^{\text{Ж}} \frac{dU_l}{d\tau} + c_l^{\gamma\delta} G_l (U_l^{\text{вых}} - U_l^{\text{вх}}), \quad i=1, \dots, N_T, l=1, \dots, N_{\text{Ж}}. \quad (1.17)$$

Здесь первое слагаемое в правой части соответствует тепловому потоку, идущему на изменение температуры теплоносителя в объеме  $l$  с полной теплоемкостью  $C_l^{\text{Ж}}$ , а второе — тепловому потоку, выносимому из объема при прохождении теплоносителя с массовым расходом  $G_l$ . Для газов, как правило, первое слагаемое пренебрежимо мало в большинстве систем охлаждения.

В общем случае теплоноситель в объеме  $l$  может поступать из других объемов ( $m=1, \dots, N_{\text{Ж}}, m \neq l$ ) и из сред с известной температурой ( $k=1, \dots, N_{\text{CP}}$ ), т. е. на входе в  $l$ -й объем может происходить смешение потоков теплоносителей с температурами  $U_m^{\text{вых}}$  и  $T_k^{CP}$ . Тогда температура  $U_l^{\text{вх}}$  рассчитывается из следующего соотношения:

$$c_l^{\gamma\delta} G_l U_l^{\text{вх}} = \sum_{m=1}^{N_{\text{Ж}}} c_m^{\gamma\delta} G_{ml} U_m^{\text{вых}} + \sum_{k=1}^{N_{\text{CP}}} c_k^{\gamma\delta} G_{kl} T_k^{CP}; \quad (1.18)$$

$$G_l = \sum_{m=1}^{N_{\text{ж}}} G_{ml} + \sum_{k=1}^{N_{\text{ср}}} G_{kl}, l=1, \dots, N_{\text{ж}},$$

где  $G_{ml}$ ,  $G_{kl}$  — массовые расходы теплоносителей, втекающих в  $l$ -й объем из  $m$ -го объема и  $k$ -й среды.

Для замыкания системы ( $N_T + 2N_{\text{ж}}$ ) уравнений (1.16) — (1.18) с неизвестными температурами  $T_i$  ( $i=1, \dots, N_T$ ),  $U_l$ ,  $U_l^{\text{ex}}$ ,  $U_l^{\text{блх}}$  ( $l=1, \dots, N_{\text{ж}}$ ) следует добавить соотношения, связывающие температуры  $U_l$  с температурами  $U_l^{\text{ex}}$ ,  $U_l^{\text{блх}}$ . Эти соотношения можно получить на основе тех или иных допущений о характере пространственного распределения температуры теплоносителя в объеме. Например, при линейном изменении температуры по длине канала справедливо равенство  $U_l = (U_l^{\text{ex}} + U_l^{\text{блх}})/2$ , а при интенсивном перемешивании теплоносителя в объеме  $U_l = U_l^{\text{блх}}$ . Обобщением этих соотношений является выражение

$$U_l = \psi_l U_l^{\text{блх}} + (1 - \psi_l) U_l^{\text{ex}}, 0 \leq \psi_l \leq 1, \quad (1.19)$$

которое и будем использовать в дальнейшем.

Величину  $\psi_l$  обычно оценивают из рассмотрения конкретных модельных задач. Например, для одномерного течения в канале длиной  $L$  с омываемым периметром  $f$  при постоянной температуре стенок  $T$  и местном коэффициенте теплоотдачи  $\alpha$  средняя массовая температура  $U(x)$  жидкости в сечении  $x$  равна

$$U(x) = T(U^{\text{ex}} - T) \exp(-bx); b = (\alpha f / cG)^{1/2}, \quad (1.20)$$

где  $c$  и  $G$  - удельная теплоемкость и массовый расход жидкости. Определив из этого выражения значение среднеобъемной температуры

$U_v = \int_0^L U dx / L$  жидкости и температуры  $U^{6yx} = U(L)$  на выходе из канала, найдем из (1.20) коэффициент

$$\psi_l = (1 - \exp(-bL))^{-1} - 1/bL. \quad (1.21)$$

Таким образом, модель системы тел и потоков теплоносителей описывается системой  $(N_T + N_{\mathcal{K}})$  обыкновенных дифференциальных уравнений (1.16), (1.17) и  $2N_{\mathcal{K}}$  алгебраических уравнений (1.18), (1.19), содержащей искомые температуры  $T_i$ ,  $U_l$ ,  $U_l^{6x}$ ,  $U_l^{6yx}$ . Для полной постановки задачи задаются значения температур в начальный момент

$$T_i|_{\tau=0} = T_{i0}; U_l|_{\tau=0} = U_{l0}. \quad (1.22)$$

В стационарном режиме, когда температуры твердых тел и теплоносителей неизменны во времени ( $dT_i/d\tau = 0, dU_l/d\tau = 0$ ), приходим к системе  $(N_T + 3N_{\mathcal{K}})$  алгебраических уравнений. Число уравнений можно уменьшить, если подставить выражения (1.19) для  $U_l$  и (1.18) для  $U_l^{6x}$  в уравнения (1.16), (1.17). В результате получим систему  $(N_T + N_{\mathcal{K}})$  алгебраических уравнений относительно температур  $T_i$  и  $U_l^{6yx}$ :

уравнения для твердых тел ( $i = 1, \dots, N_T$ )

$$\begin{aligned} & \left( \sum_{j=1}^{N_T} \sigma_{ij}^{T.T.} + \sum_{l=1}^{N_{\mathcal{K}}} \sigma_{il}^{T.\mathcal{K}} + \sum_{k=1}^{N_{CP}} \sigma_{ik}^{T.CP.} \right) T_i - \sum_{j=1}^{N_T} \sigma_{ij}^{T.T.} T_j - \sum_{l=1}^{N_{\mathcal{K}}} \sigma_{il}^{T.\mathcal{K}} (1 - \psi_l) \left( \sum_{m=1}^{N_{\mathcal{K}}} c_m^{y\partial} G_{ml} U_m^{6yx} \right) / c_l^{y\partial} G_l - \\ & - \sum_{l=1}^{N_{\mathcal{K}}} \sigma_{il}^{T.\mathcal{K}} \psi_l U_l^{6yx} = P_i^T + \sum_{k=1}^{N_{CP}} \sigma_{ik}^{T.CP.} T_k^{CP} + \sum_{l=1}^{N_{\mathcal{K}}} \sigma_{il}^{T.\mathcal{K}} (1 - \psi_l) \left( \sum_{k=1}^{N_{CP}} c_k^{y\partial} G_{kl} T_k^{CP} \right) / c_l^{y\partial} G_l; \end{aligned} \quad (1.23)$$

уравнения для теплоносителей ( $l=1, \dots, N_{\text{ж}}$ )

$$\begin{aligned} & \left( \sum_{i=1}^{N_T} \sigma_{il}^{T, \text{ж}} \psi_l + c_l^{y0} G_l \right) U_l^{\text{бых}} - \sum_{i=1}^{N_T} \sigma_{il}^{T, \text{ж}} T_i + \left( \sum_{i=1}^{N_T} \sigma_{il}^{T, \text{ж}} (1 - \psi_l) - c_l^{y0} G_l \right) \left( \sum_{m=1}^{N_{\text{ж}}} c_m^{y0} G_{ml} U_m^{\text{бых}} \right) / c_l^{y0} G_l = \\ & = P_l^{\text{ж}} + (c_l^{y0} G_l - \sum_{i=1}^{N_T} \sigma_{il}^{T, \text{ж}} (1 - \psi_l)) \left( \sum_{k=1}^{N_{\text{CP}}} c_k^{y0} G_{kl} G_{kl} T_k^{\text{CP}} \right) / c_l^{y0} G_l. \end{aligned} \quad (1.24)$$

При записи системы (1.23), (1.24) в левых частях уравнений сгруппированы коэффициенты при искомым температурах  $T_i$  и  $U_l^{\text{бых}}$ , а в правых записаны члены, содержащие заданные температуры  $T_k^{\text{CP}}$  и мощности. Видно, что по сравнению с исходной формой записи системы ( $N_T + 3N_{\text{ж}}$ ) уравнений в (1.23), (1.24) усложнилась структура матрицы системы уравнений, но зато сократилось число уравнений, что в ряде случаев весьма существенно при численной реализации модели. Заметим, что для частного случая  $\psi_l = 1$  (идеальное перемешивание теплоносителя в объеме  $l$ ) структура системы уравнений сильно упрощается.

Лучистые и конвективные тепловые проводимости зависят от искомым температур взаимодействующих тел. Значения теплоемкостей, мощностей и расходов теплоносителей также могут зависеть от температур. Поэтому в общем случае получающиеся системы алгебраических (для стационарного режима) или обыкновенных дифференциальных (для нестационарного) уравнений являются нелинейными. Однако при решении систем нелинейных алгебраических уравнений теплового баланса обычно организуют итерационный процесс, при котором определение очередного приближения проводится путем решения системы линейных уравнений, в которой проводимости, теплоемкости и мощности рассчитаны по значениям температур, найденным на предыдущей итерации. При численном решении систем обыкновенных дифференциальных уравнений, описывающих нестационарный режим системы тел, вычисляют значения зависящих от температуры

коэффициентов по температурам на предыдущем шаге по времени или же используют известные методы решения задачи Коши для систем нелинейных уравнений.

#### **1.1.4.2 Одномерные модели**

В случае, когда применение модели с сосредоточенными параметрами не позволяет получить необходимую проектировщику информацию о тепловом режиме исследуемого объекта, целесообразно сделать следующий шаг: усложнить анализ и перейти к одномерной модели. Применение одномерных моделей может быть вызвано рядом причин.

Во-первых, часто встречаются элементы приборов, у которых температурное поле имеет существенную неравномерность только по какой-то одной пространственной координате, и эта неравномерность влияет на их функциональные характеристики. Такая ситуация особенно характерна для различных оптических элементов. Например, неравномерность температуры по оси зеркала приводит к его изгибу, который вызывает появление термических aberrаций в оптической системе [3]. Для иллюминатора, линзы, активного элемента твердотельного лазера с жидкостным охлаждением, электрооптического затвора существенна неравномерность температурного поля по радиальной координате. Из-за этой неравномерности в иллюминаторе и активном элементе возникает тепловая линза [3], в линзе — термическая сферическая aberrация и изменение фокусного расстояния [3], у электрооптического затвора изменяется коэффициент пропускания [3].

Во-вторых, в случае, когда целью анализа является получение информации о характерных температурах — среднеповерхностных, среднеобъемных, минимальных и максимальных, применение одномерных моделей позволяет уменьшить погрешность их определения. В рамках моделей с сосредоточенными параметрами такую информацию можно получить только с

использованием различных коэффициентов неравномерности, точность определения которых может быть недостаточной. Подобные одномерные модели могут использоваться для канонических тел при преимущественном изменении температуры по какой-то одной пространственной координате (например, сфера и сферическая оболочка, цилиндр и цилиндрическая оболочка при условии зависимости температурного поля только от радиальной координаты).

Наконец, в случае системы тел использование одномерных моделей позволяет уменьшить погрешность расчета ее теплового режима за счет уточнения математического описания теплового взаимодействия между телами. Такая ситуация возникает, например, в задаче расчета теплового режима блока на печатных платах с кондуктивным отводом теплоты на охлаждаемое основание.

Рассмотрим тепловые модели обобщенной криволинейной стенки, обобщенного сплошного тела и одномерного потока хладагента (теплоносителя), а также соответствующих одномерных уравнений.

Модели обобщенной криволинейной стенки и обобщенного одномерного сплошного тела применяются для описания теплового режима элементов, у которых температура изменяется преимущественно по толщине или по мере удаления от центра элемента к его поверхности. К простейшим из них относятся элементы, имеющие каноническую форму: плоские стенки с температурным полем, изменяющимся только по толщине, сплошные или полые круговые цилиндры и шары с температурным полем, зависящим только от радиальной координаты. Соответствующие уравнения имеют вид:

для плоской стенки

$$c_p \frac{\partial T}{\partial \tau} = \frac{\partial}{\partial x} \left( \lambda \frac{\partial T}{\partial x} \right) + q_v ; \quad (1.25)$$

для цилиндра

$$c\rho \frac{\partial T}{\partial \tau} = \frac{1}{r} \frac{\partial}{\partial r} \left( r\lambda \frac{\partial T}{\partial r} \right) + q_v ; \quad (1.26)$$

для шара

$$c\rho \frac{\partial T}{\partial \tau} = \frac{1}{r^2} \frac{\partial}{\partial r} \left( r^2 \lambda \frac{\partial T}{\partial r} \right) + q_v ; \quad (1.27)$$

Остановимся на модели обобщенной криволинейной стенки подробнее. Рассмотрим замкнутую оболочку, внутренняя и внешняя поверхности которой с площадями  $S_1$  и  $S_2$  являются изотермическими и имеют температуры  $T_1$  и  $T_2$ . Внутри оболочки любая изотермическая поверхность располагается между ее внешней и внутренней поверхностями и имеет площадь  $S$ . Основная идея предлагаемой методики заключается во введении зависимости  $S(\rho)$  площади изотермической поверхности  $S$  от некоторого единственного параметра  $\rho$ , называемого обобщенной координатой. Подчеркнем, что, несмотря на такой «геометрический» термин, параметр  $\rho$  в общем случае непосредственно не связан с «обычными» пространственными координатами декартовой или цилиндрической системы. Он изменяется в диапазоне  $[0, L]$ , причем значение  $\rho=0$  соответствует внутренней поверхности оболочки, а  $\rho=L$  - внешней поверхности. Величина  $L$  называется определяющим размером геометрической модели объекта (характерной толщиной криволинейной стенки).

Поскольку предполагается, что площади изотермических поверхностей зависят от единственного параметра  $\rho$ , то и соответствующие этим поверхностям температуры зависят только от обобщенной координаты  $\rho$  и времени  $\tau$ , и можно рассматривать одномерное распределение температуры  $T(\rho, \tau)$ . Подчеркнем, что это распределение вводится не с целью определения



пространственного распределения температуры в реальном элементе, так как параметр  $\rho$ , как было уже отмечено, обычно не привязан к какой-либо реальной пространственной координате. Целью рассмотрения зависимости  $T(\rho, \tau)$  является получение характерных температур реального объекта: среднеповерхностных  $T_1(\tau)$ ,  $T_2(\tau)$  - их можно положить равными  $T(0, \tau)$  и  $T(L, \tau)$  соответственно, среднеобъемной  $T_v(\tau)$ , которую можно найти по формуле

$$T_v(\tau) = \frac{1}{V} \int_0^L T(\rho, \tau) S(\rho) d\rho; \quad V = \int_0^L S(\rho) d\rho, \quad (1.28)$$

а также максимальной или минимальной температур и перепада температуры в объекте.

При сделанных допущениях о зависимости площади изотермических поверхностей от обобщенной координаты одномерное уравнение для распределения  $T(\rho, \tau)$  имеет вид

$$c\gamma \frac{\partial T}{\partial \tau} = \frac{1}{S(\rho)} \frac{\partial}{\partial \rho} \left( \lambda S(\rho) \frac{\partial T}{\partial \rho} \right) + q_v(\rho, \tau), \quad 0 < \rho < L. \quad (1.29)$$

Граничные условия для уравнения (1.29) записываются с учетом того, что тепловые потоки  $q_{S1}$  и  $q_{S2}$  с единицы поверхностей оболочки  $S_1$  и  $S_2$  определяются по формулам:

$$q_{S1, S2} = \mp \lambda \frac{\partial T}{\partial \rho} \Big|_{\rho=0, L}. \quad (1.30)$$

Перейдем к рекомендациям по выбору характерного размера  $L$  и зависимости  $S(\rho)$ . Очевидно, что для зависимости  $S(\rho)$  должны выполняться

условия

$$S(0)=S_1 ; \quad S(L)=S_2 . \quad (1.31)$$

Кроме того, целесообразно потребовать, чтобы объем «модельной» криволинейной стенки был бы равен объему реальной стенки  $V$  :

$$\int_0^L S(\rho) d\rho = V . \quad (1.32)$$

Условия (1.31) и (1.32) оставляют достаточно широкие возможности для выбора зависимости  $S(\rho)$  .

Обычно зависимость  $S(\rho)$  представляется в виде полинома второй степени [3]

$$S(\rho)=a+b\rho+d\rho^2 , \quad (1.33)$$

у которого для определения трех коэффициентов  $a$  ,  $b$  ,  $d$  используются три условия (1.31), (1.32). Отсюда получаются следующие соотношения:

$$S(\bar{\rho})=S_1+\frac{6V}{L}(\tilde{b}\bar{\rho}-\tilde{d}\bar{\rho}^2) ;$$

$$\bar{\rho}=\frac{\rho}{L} ; \quad \tilde{b}=1-\frac{2}{3}\varphi_1-\frac{1}{3}\varphi_2 ; \quad \tilde{d}=1-(\varphi_1+\varphi_2)/2 ; \quad (1.34)$$

$$\varphi_i=\frac{LS_i}{V}, i=1,2 .$$

Отметим, что для оболочек канонической формы: цилиндрической,

шаровой, а также для плоской стенки — зависимость (1.34) является точной. Для них определяющий размер  $L$  совпадает с реальной толщиной стенки. Для оболочек неканонической конфигурации определяющий размер  $L$  рассчитывается по формуле

$$L = 2V/(S_1 + S_2) \quad \text{при} \quad S_2/S_1 < 4. \quad (1.35)$$

Рассмотренный прием перехода к одномерному приближению для температурного поля можно применять не только для элементов типа оболочек и стенок, но и для сплошных однородных тел типа выпуклых многогранников. В этом случае также вводится обобщенная координата  $\rho$ , но, в отличие от оболочки, значение  $\rho=0$  считается соответствующим «центральной» точке реального элемента. Для тел правильной формы, имеющих оси или плоскости симметрии, центральная точка совпадает с центром тяжести тела.

Такое одномерное приближение целесообразно применять только в случае, когда в центральной точке элемента его температура достигает экстремального значения, а ее изменение происходит в основном по мере удаления от центральной точки к поверхности тела. Очевидно, что подобный характер распределения температурного поля требует наличия соответствующей симметрии в условиях теплообмена на поверхности реального тела. Уравнение для одномерного приближения  $T(\rho, \tau)$  температурного поля сплошного тела также имеет вид (1.29). Граничное условие на поверхности  $\rho=L$  записывается так же, как и для криволинейной стенки, а граничное условие в центральной точке при  $\rho=0$  следует из условия экстремума температуры

$$\left. \frac{\partial T}{\partial \tau} \right|_{\rho=0} = 0. \quad (1.36)$$

Для сплошных тел используется аппроксимация зависимости площади изотермической поверхности от обобщенной координаты  $S(\rho)$ , отличная от (1.33). Для них целесообразно применять аппроксимацию вида

$$S(\bar{\rho}) = S \bar{\rho}^n, \bar{\rho} = \rho/L, 0 < \rho < L, \quad (1.37)$$

где  $S$  - площадь поверхности реального тела;  $n$  - коэффициент, учитывающий его форму и называемый поэтому фактором формы тела. Выражение для фактора формы можно найти из условия сохранения объема модельного и реального тел

$$n = SL/V - 1. \quad (1.38)$$

Для плоской стенки, цилиндра и шара  $n=0,1,2$  соответственно.

По выбору определяющего размера  $L$  предлагается несколько рекомендаций. Если реальное тело близко по форме к шару, цилиндру (при изменении температуры в основном по радиальной координате), то в качестве определяющего размера целесообразно взять радиус соответствующего равновеликого по площади поверхности канонического тела. Равновеликость объема при этом будет обеспечена выбором фактора формы согласно (1.38).

Таким образом, для тел класса шара или цилиндра справедливы соотношения

$$L = \frac{1}{2} \sqrt{\frac{S}{\pi}}; \quad L = \frac{S}{2\pi h}, \quad (1.39)$$

где  $h$  - длина тела типа цилиндра.

Для тел, которые плохо вписываются в приближения канонических форм, определяющий размер  $L$  следует вычислять по формуле

$$\frac{2}{L} = \frac{1}{L_{min}} + \frac{1}{L_{max}}, \quad (1.40)$$

где  $L_{min}$ ,  $L_{max}$  - минимальное и максимальное расстояния от центра тела до его поверхности.

Перейдём к одномерному потоку хладагента (теплоносителя). Рассмотрим поток хладагента, движущийся в канале в направлении  $x$  и омывающий поверхности  $N$  тел с температурами  $T_1, \dots, T_N$ . При описании в одномерном приближении теплового режима хладагента вводится среднерасходная в сечении  $x$  температура  $U(x)$

$$U(x) = \frac{1}{G} \int_S v_x \rho U dS; \quad G = \int_S v_x \rho dS \quad (1.41)$$

и принимается, что тепловой поток между телом и хладагентом в сечении  $x$  может быть выражен через температуру тела, среднерасходную температуру хладагента и соответствующий коэффициент теплоотдачи. Это допущение позволяет записать уравнение энергии для распределения среднерасходной температуры хладагента  $U(x, \tau)$  по длине канала в виде

$$cG \left( \frac{1}{v_x} \frac{\partial U}{\partial \tau} + \frac{\partial U}{\partial x} \right) = q_v(x, \tau) S(x) - \sum_{n=1}^N \alpha_n(x, U, T_n) \xi_n(x) [A_n(U) - B_n(T_n)] \quad (1.42)$$

где  $G$  - массовый расход;  $v_x$  — средняя по сечению канала скорость движения жидкости;  $\alpha_n$  — коэффициент теплоотдачи;  $\xi_n$  — омываемый периметр  $n$ -го тела;  $T_n$  — температура поверхности  $n$ -го омываемого тела;  $A_n$ ,  $B_n$  — некоторые функционалы от температурных полей;  $q_v$  — мощность

внутренних источников в единице объема жидкости;  $S$  — площадь поперечного сечения канала.

Для газообразных хладагентов при описании нестационарных процессов, как правило, можно пренебречь в левой части (1.42) локальной производной  $\partial U / \partial \tau$  по сравнению с конвективной производной  $v_x \partial U / \partial x$ .

Математическая одномерная модель теплового режима устройства представляет собой систему уравнений вида (1.29), (1.42). Кроме того, в эту систему могут входить и обыкновенные дифференциальные уравнения, соответствующие моделям с сосредоточенными параметрами для отдельных тел и хладагентов.

### 1.1.4.3 Многомерные модели

Многомерные математические модели обычно используют либо когда проводится определение температурных полей отдельных элементов, либо когда анализируется тепловой режим узла, состоящего из неделимых элементов.

Причина применения этих моделей заключается в необходимости получения используемой при проектировании приборов информации либо о пространственном распределении температуры в элементах, либо об их некоторых характерных температурах (максимальных, осредненных по некоторым локальным участкам поверхности или объема и т. д.). При определении многомерных температурных полей возникают два типа задач - многомерные задачи для одиночных тел и для системы тел.

Задачи первого типа, называемые далее многомерными задачами для одиночных тел, заключаются в решении какого-либо многомерного уравнения теплопроводности для отдельной области при заданных граничных условиях. Эта область может соответствовать либо одному элементу, либо узлу, который может быть представлен в виде сплошного тела с кусочно-непрерывными теплофизическими свойствами, либо блоку более высокого уровня иерархии

при использовании для него модели квазиоднородного тела с эффективными теплофизическими свойствами.

Рассмотрим печатную плату с установленными на ней элементами, температуры корпусов которых требуется найти. Для анализа стационарного теплового режима платы с элементами необходимо решить вспомогательную задачу о температурном поле платы с одиночным локальным источником теплоты на ее поверхности. При ее решении перепадом температуры по толщине платы пренебрегают и используют тепловую модель пластины с двумерным температурным полем  $T(x, y)$ , теплообменом на боковой поверхности и локальным источником, занимающим область  $(x_1 < x < x_2, y_1 < y < y_2)$ . Таким образом, для определения двумерного распределения перегрева  $\vartheta(x, y)$ , созданного источником, решают уравнение

$$\frac{\partial}{\partial x} \left( \lambda \frac{\partial \vartheta}{\partial x} \right) + \frac{\partial}{\partial y} \left( \lambda \frac{\partial \vartheta}{\partial y} \right) - \frac{\alpha}{\delta} \vartheta + \frac{q(x, y)}{\delta} = 0, \quad (1.43)$$

в котором распределение плотности теплового потока  $q(x, y)$  задается в виде

$$q(x, y) = \begin{cases} P/(x_2 - x_1)(y_2 - y_1), & x_1 < x < x_2, y_1 < y < y_2; \\ 0, & \text{вне области источника.} \end{cases} \quad (1.44)$$

На торцах платы задаются либо граничные условия, либо условия отсутствия теплообмена.

Задачи второго типа, называемые далее многомерными задачами для систем тел, заключаются в решении системы уравнений, в которую наряду с одним или несколькими многомерными уравнениями теплопроводности входят одномерные уравнения теплопроводности или энергии, а также обыкновенные дифференциальные уравнения, соответствующие моделям элементов с

сосредоточенными параметрами. Особенностью математических постановок этих задач является то, что в граничные условия для многомерного уравнения, описывающего температурное поле  $T_i(x, y, z, \tau)$   $i$ -го тела, входят неизвестные пространственные распределения  $T_j(x, y, z, \tau)$  или средние температуры других тел или потоков хладагента. В случае уравнений для пластин и дисков неизвестные температуры других тел могут входить в члены уравнений, описывающие боковой теплообмен.

В наиболее сложном случае проводится решение системы многомерных уравнений для всех элементов модели, которую в достаточно общей постановке можно записать следующим образом:

$$c_i \rho_i \frac{\partial T_i}{\partial \tau} = \nabla(\lambda_i \nabla T_i) + q_{vi}(\vec{x}_i, \tau); \quad (1.45)$$

$$\begin{aligned} \lambda_i \frac{\partial T_i}{\partial n} \Big|_{\tau_i} = & \sum_{j=1}^m \alpha_{ij}(\vec{x}_i, T_i, T_j)(T_j - T_i) + \sum_{j=1}^m \int_{A_j} \sigma \beta_{ij}(\vec{x}_i, \vec{x}_j)(T_j^4 - T_i^4) dA_j + q_s(\vec{x}_i, \tau) + \\ & + \sum_{j=m+1}^M \alpha_{ij}(\vec{x}_i, T_i, \tilde{T}_j)(\tilde{T}_j - T_i), \end{aligned} \quad (1.46)$$

где индексы  $j=1, \dots, m$  соответствуют элементам модели; индексы  $j=m+1, \dots, M$  соответствуют телам и средам с найденными на предыдущих этапах расчета температурами;  $\alpha_{ij}$  — локальные коэффициенты теплоотдачи;  $\sigma$  — постоянная Стефана—Больцмана;  $\beta_{ij}$  — коэффициенты, описывающие лучистый теплообмен. Коэффициенты зависят от геометрии системы, коэффициентов черноты, зеркального и диффузного отражений. Первая сумма в правой части соотношения (1.46) описывает конвективный и контактный теплообмен между элементами модели, вторая — лучистый теплообмен, а третья — внешние тепловые воздействия со стороны тел и сред с найденными на предыдущих этапах расчета температурами.



## **1.2 Особенности существующих решений автоматизации процесса расчёта теплового режима печатной платы**

Для моделирования тепловых процессов могут использоваться различные программы теплового проектирования РЭА: Mentor Graphics, Beta Soft, TAS, Thermal Designer 98, FLOTHERM, COSMOS, PRAC, АСОНИКА-Т, ТРиАНА и другие. Объединяет их одно: по исходным данным в автоматическом режиме синтезируется модель тепловых процессов, по которой проводится моделирование теплового режима.

В качестве примера, рассмотрим проведение анализа тепловых режимов печатных плат с использованием программного обеспечения фирмы Dynamic Soft Analysis, Inc - Beta Soft. При расчете используется более 50 уравнений, позволяющих провести полноценное трехмерное моделирование явлений теплопередачи на плате.

При проведении моделирования используются численные методы, основанные на продвинутых методах конечных разностей с адаптивными сетками, которые генерируют адаптивные местные сетки для того, чтобы не допускать снижения точности на мелких деталях проекта, не попадающих в основную сетку платы. Моделируются процессы теплообмена, теплопроводности, конвекции и излучения. Особое внимание уделено моделированию воздушной конвекции с учетом трехмерного расположения компонентов на плате.

Программа BetaSoft имеет специальный конвертер, обеспечивающий интерфейс с САПР печатных плат различных производителей. На рис. 1.4 показано окно программы BetaSoft с загруженной печатной платой.

Имеется возможность устанавливать систему единиц измерения и пределы изменения температуры.

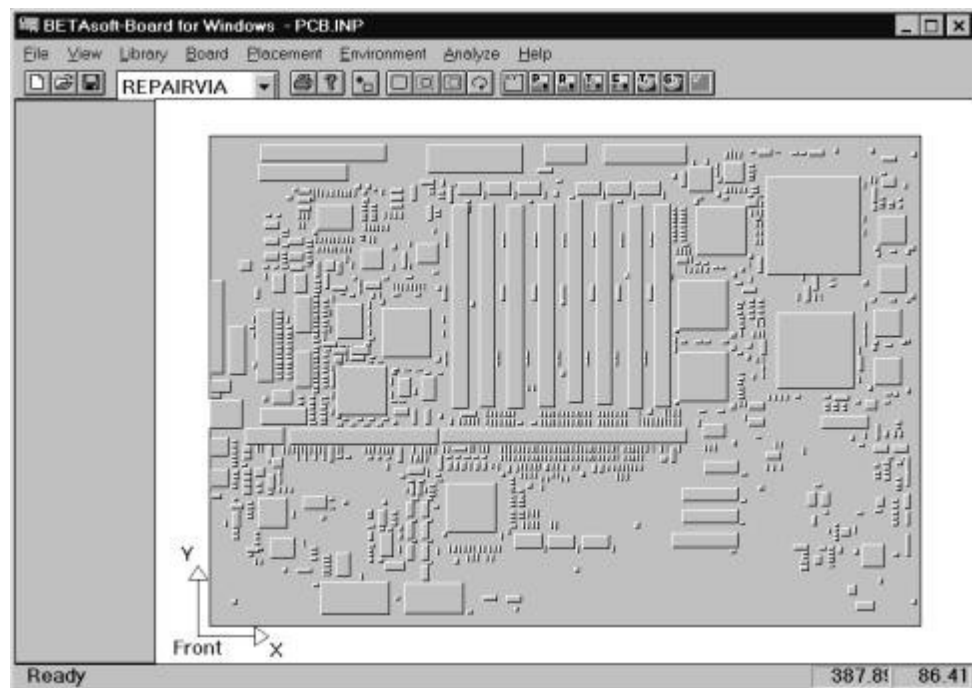


Рис. 1.4. Окно программы BetaSoft с загруженной печатной платой

Обеспечен просмотр верхней или нижней стороны платы, а также послойный просмотр. Пользователь может изменять масштаб изображения и вращать плату в пространстве (рис. 1.5).

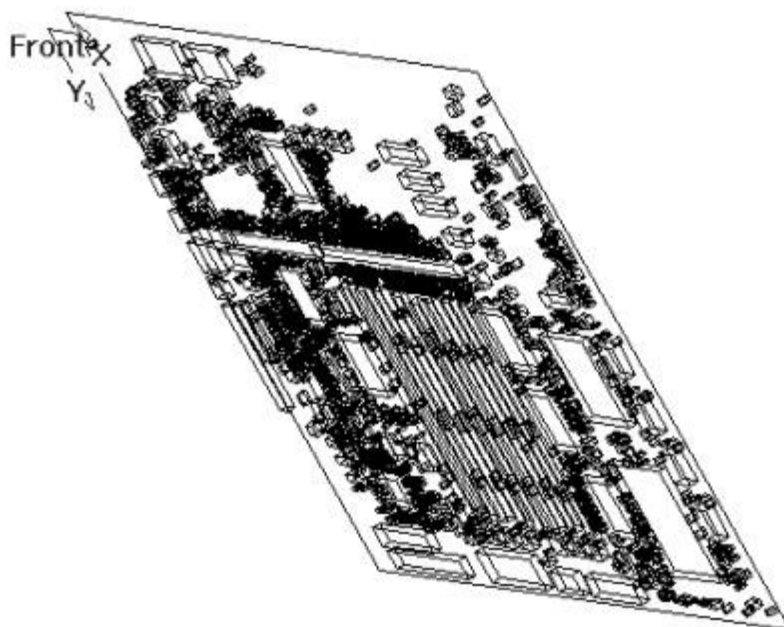


Рис. 1.5. Вращение платы в пространстве

На рис. 1.6 и 1.7 приведены примеры экранов программы BetaSoft при анализе мощности рассеяния и построении карты распределения температур печатной платы.

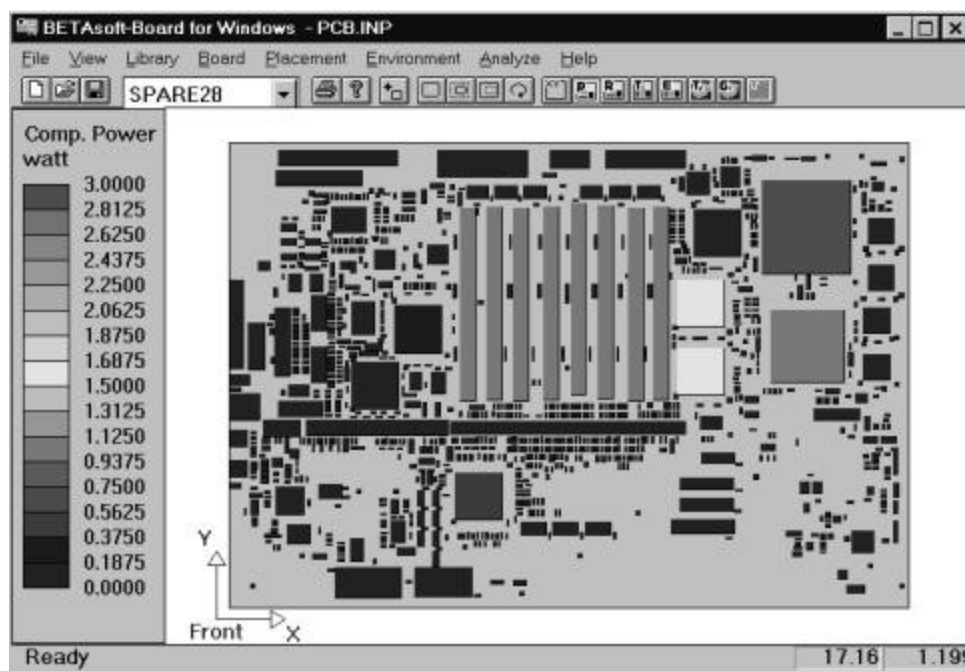


Рис. 1.6. Мощности рассеяния компонентов

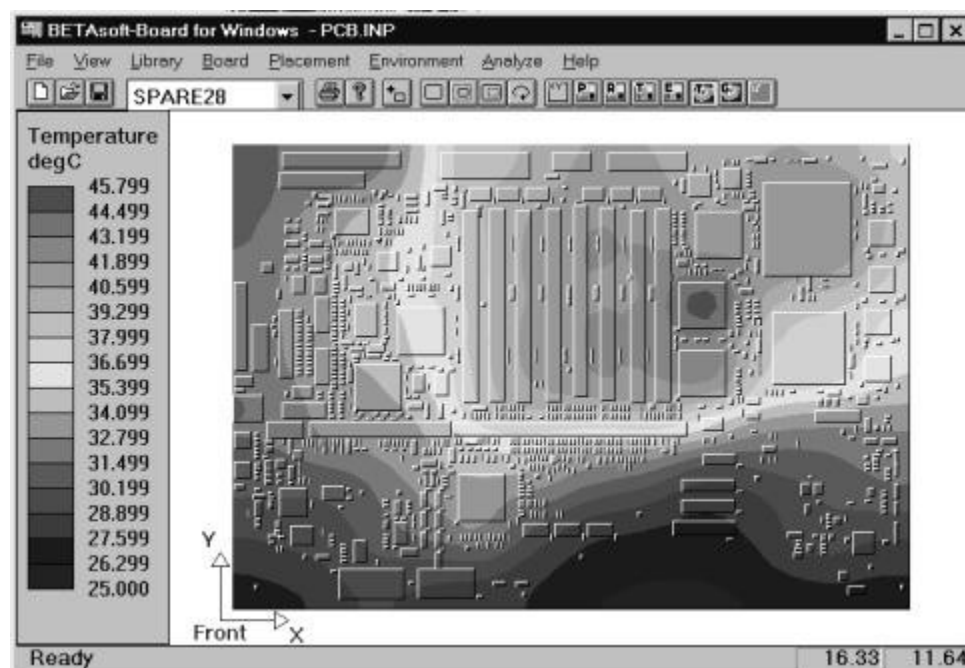


Рис. 1.7. Карта распределения температур платы

Существует инструментарий предназначенный для работы с библиотеками компонентов - создавать новые компоненты, добавлять их с новым корпусом или параметрами, копировать, удалять их и т. п.

В меню редактирования компонента основной библиотеки задаются параметры, характеризующие геометрию используемого компонента, такие как тип корпуса, его размеры, величина воздушного зазора между компонентом и платой, размеры и число выводов, их теплопроводность; приводятся параметры, характеризующие тепловые свойства компонента, такие как мощность рассеяния, тепловое сопротивление между кристаллом и корпусом компонента, тепловые сопротивления между воздушной средой и корпусом компонента при различных скоростях обдува, коэффициент теплообмена излучением, свойства теплопроводности зазора, предельные значения температур корпуса и кристалла компонента.

Задавая соответствующие значения в диалоговом режиме, разработчик достаточно легко может настроить свою собственную библиотеку компонентов, учитывающую все особенности проекта.

Имеются меню, позволяющее задать необходимые параметры платы: геометрические размеры и толщины слоев платы, задать их тепловые свойства.

Возможно послойное задание теплопроводности как металлизированных, так и диэлектрических участков платы, долю металлизированных участков во всей поверхности платы, а также предельных температур корпусов и кристаллов компонентов.

Существует меню для задания свойств среды анализа. Возможно задание температуры и давления окружающего воздуха, условия принудительного охлаждения, ориентацию плат, коэффициенты эффективности теплоотдачи и т. п. Кроме того, в этом же меню определяется характеристика алгоритма расчета.

### **1.3 Постановка задачи автоматизации процесса расчета теплового режима печатной платы**

Сформулируем постановку задачи автоматизации процесса расчета теплового режима печатной платы (в рамках данной работы) следующим образом: требуется реализовать программный комплекс автоматизированного расчета стационарного теплового режима печатной платы. Исходными данными к этому расчету являются файлы конструкторской документации САПР печатных плат с информацией о размерах печатной платы и элементов на ней, а также с информацией о параметрах размещённых элементов. Потоки теплоносителей не учитывать, так как, в случае с печатной платой, они мало влияют на общую картину теплового режима.

В настоящее время наиболее популярна САПР Altium Designer. Эту популярность САПР Altium Designer набрала не просто так. Эта САПР имеет достаточно мощные средства конструирования печатных плат, что подтвердят многие инженеры. К сожалению, расчета теплового режима в ней не предусматривается. Исходя из этого, в качестве САПР печатных плат необходимо использовать САПР Altium Designer.

Реализуемый программный комплекс должен работать следующим образом:

- ⌚ пользователь задаёт исходные данные (файлы конструкторской документации САПР);
- ⌚ исходные данные обрабатываются программным комплексом;
- ⌚ пользователю отображаются обработанные исходные данные и предлагается вручную дополнить исходные данные (например, температура окружающей среды);
- ⌚ строится тепловая модель печатной платы с последующим переводом в математическую модель;

- Ⓟ производится решение математической модели с выводом результата пользователю.

Тепловой моделью платы с элементами является прямоугольная пластина, на которой с двух сторон расположены элементы с равномерными температурными полями и мощностями.

В качестве математической модели необходимо использовать модель с сосредоточенными параметрами. Хотя эта модель и не строго учитывает взаимное влияние элементов на плате и теплообмен с окружающей средой, её использование даёт представление о тепловом режиме печатной платы с использованием минимальных затрат ресурсов ЭВМ.

Рассмотрим численное решение задачи определения стационарных температур по моделям с сосредоточенными параметрами без учета потоков теплоносителей.

В общем случае необходимо решить систему нелинейных алгебраических уравнений вида (1.23). Однако если считать тепловые проводимости  $\sigma_{ij}^{T.T.}$ ,  $\sigma_{ik}^{T.CP.}$ , а также мощности источников  $P_i^T$  известными, то система превращается в систему  $N_T$  линейных алгебраических уравнений относительно температур тел  $T_i$  ( $i=1, \dots, N_T$ ). Такую систему удобно записать в матричной форме

$$AX = B, \quad (1.47)$$

где  $X = (x_1, x_2, \dots, x_{N_T})^T = (T_1, \dots, T_{N_T})^T$  – вектор-столбец из  $N_T$  последовательно записанных неизвестных температур тел;  $B = (b_1, \dots, b_{N_T})^T$  – вектор-столбец свободных членов;  $A = \|a_{nm}\|$  – квадратная матрица коэффициентов размерности  $N_T \times N_T$ .

При любом подходе основной частью алгоритма решения системы стационарных уравнений модели с сосредоточенными параметрами является

процедура решения соответствующей линеаризованной системы.

Методы решения систем линейных алгебраических уравнений разделяются на две группы: прямые и итерационные методы. В прямых методах решение находится за конечное число действий. В итерационных методах решения линейных систем сначала задается некоторое начальное приближение, а затем реализуется повторяющийся процесс определения последовательных приближений, в котором каждое последующее приближение находится по одинаковому алгоритму на основе предыдущего.

При наличии сильных температурных зависимостей тепловых проводимостей, расходов или мощностей итерационный процесс может не обладать сходимостью. Конечно, существуют способы решения этой проблемы (например искусственное «торможение» изменения искомого решения от итерации к итерации), но прямые методы не требуют дополнительных затрат для достижения результата. Исходя из этого будем использовать прямые методы.

Наиболее распространенным прямым методом является метод последовательного исключения Гаусса [3]. Он основан на весьма простом приеме и заключается в последовательном исключении неизвестных  $x_n$  из всех уравнений с номерами  $m > n$ , начиная с  $n=1$ , и получении в конечном итоге линейного уравнения с одним неизвестным  $x_{N_T}$ . Далее из преобразованной системы уравнений с треугольной матрицей в обратном порядке находятся остальные неизвестные  $x_{N_T-1}, \dots, x_1$ .

Таким образом, система (1.47) принимает вид:

$$\begin{bmatrix} \left( \sum_{j=1}^{N_T} \sigma_{1j}^{T.T.} + \sum_{k=1}^{N_{CP}} \sigma_{1k}^{T.CP.} \right) & -\sigma_{12}^{T.T.} & \dots & -\sigma_{1N_T}^{T.T.} \\ -\sigma_{21}^{T.T.} & \left( \sum_{j=1}^{N_T} \sigma_{2j}^{T.T.} + \sum_{k=1}^{N_{CP}} \sigma_{2k}^{T.CP.} \right) & \dots & -\sigma_{2N_T}^{T.T.} \\ \vdots & \dots & \dots & \vdots \\ -\sigma_{N_T 1}^{T.T.} & -\sigma_{N_T 2}^{T.T.} & \dots & \left( \sum_{j=1}^{N_T} \sigma_{N_T j}^{T.T.} + \sum_{k=1}^{N_{CP}} \sigma_{N_T k}^{T.CP.} \right) \end{bmatrix} \times \begin{bmatrix} T_1 \\ T_2 \\ \vdots \\ T_{N_T} \end{bmatrix} = \begin{bmatrix} P_1^T + \sum_{k=1}^{N_{CP}} \sigma_{1k}^{T.CP.} T_k^{CP} \\ P_2^T + \sum_{k=1}^{N_{CP}} \sigma_{2k}^{T.CP.} T_k^{CP} \\ \vdots \\ P_{N_T}^T + \sum_{k=1}^{N_{CP}} \sigma_{N_T k}^{T.CP.} T_k^{CP} \end{bmatrix}$$

(1.48)

Матрица  $A$  этой системы является симметричной, так как  $a_{nm} = a_{mn} = -\sigma_{nm}^{T.T.} = -\sigma_{mn}^{T.T.}$  и для ее решения следует это учитывать.

Неопределёнными величинами остаются значения тепловой проводимости и мощности источника теплоты. Определим эти величины.

Выразим каждый элемент ПП и пространство между этими элементами через параллелепипеды. Данные параллелепипеды представляют габариты элемента ПП или описанного пространства. Тогда, согласно закону теплопроводности Фурье (записанному в интегральной форме), тепловую проводимость можно выразить в виде:

$$\sigma_{ij}^{T.T.} = \lambda \frac{S_{ij}^{T.T.}}{l_{ij}^{T.T.}}, \quad (1.49)$$

$$\sigma_{ik}^{T.CP.} = \lambda \frac{S_{ik}^{T.CP.}}{l_{ik}^{T.CP.}},$$

где  $\lambda$  — теплопроводность среды ПП;  $S_{ij}^{T.T.}$  и  $S_{ik}^{T.CP.}$  - площадь сечения параллелепипеда пространства между телами  $i$  и  $j$ , телом  $i$  и средой  $k$  соответственно;  $l_{ij}^{T.T.}$  и  $l_{ik}^{T.CP.}$  - длина параллелепипеда пространства между телами  $i$  и  $j$ , телом  $i$  и средой  $k$  соответственно.

В случае, если мощность источника теплоты  $P_i^T$  тела  $i$  ПП неизвестна, то ее можно рассчитать как мощность постоянного тока согласно закону Ома, в виде:

$$P_i^T = \frac{U^2}{R_i^T} = \frac{U^2 S_i^T}{\rho_i^T l_i^T}, \quad (1.50)$$

где  $U$  — напряжение в цепи;  $S_i^T$  — площадь сечения параллелепипеда элемента  $i$  ПП;  $\rho_i^T$  — удельное сопротивление вещества элемента  $i$  ПП;  $l_i^T$  - длина параллелепипеда элемента  $i$  ПП.



## 2. ПРОГРАММНАЯ ЧАСТЬ

### 2.1 Перечень исходных документов

Исходя из постановки задачи, для осуществления автоматизации выбрана САПР Altium Designer 10. Данная САПР поддерживает экспортирование всех типов документов, используемых при проектировании печатной платы [4]. А именно следующие типы:

- ⌚ Schematic Editor;
- ⌚ PCB Editor;
- ⌚ Schematic Library Editor;
- ⌚ PCB Library Editor.

Документы типа PCB Editor содержат в себе сборочный чертеж сконструированной платы, с элементами из документов-библиотек типа *PCB Library Editor* и *Schematic Library Editor*. Документы типа *Schematic Editor* не представляют особого интереса в рамках данной работы, так как они содержат в себе лишь информацию о принципиальной схеме печатной платы. Таким образом необходимым и достаточным типом документов решаемой задачи является тип *PCB Editor*.

Документы типа PCB Editor можно экспортировать в виде:

2. Altium PCB binary (\*.PcbDoc);
3. Altium PCB binary V3.0, V4.0 (\*.PCB);
4. Altium PCB binary V5.0 (\*.PcbDoc);
5. Altium PCB ASCII (\*.PcbDoc)
6. другие виды экспортирования для других САПР.

Для последующего программного разбора документа типа *PCB Editor*, его следует экспортировать в виде *Altium PCB ASCII (\*.PcbDoc)*. Именно в этот вид, так как такой вид позволяет программно читать документ в текстовом виде,

в отличие от бинарных (*binary*) видов документов *PCB Editor*.

## 2.2 Анализ структуры исходных документов

Как было сказано ранее - для последующего программного разбора документа типа *PCB Editor*, его следует экспортировать в виде *Altium PCB ASCII (\*.PcbDoc)*. Проанализируем структуру данного вида документов *PCB Editor*.

Если открыть документ вида *Altium PCB ASCII (\*.PcbDoc)* на чтение, то можно увидеть следующее (рис. 2.1): каждая информационная единица документа разбита символом «|». Совокупности информационных единиц разбиты на группы с идентификатором «RECORD». Обозначим группы с идентификатором «RECORD» как «записи».

```
00`CFG3D.SHOWMECHANICALLAYERS=FALSE`CFG3D.MECHANICALLAYERSOPACITY=1.000000
|RECORD=EngineeringChangeOrderOptions|ECOISACTIVE=FALSE|ECOFILNAME=PCB.EC
|RECORD=OutputOptions|DRILLGUIDEHOLESIZE=30mil|DRILLDRAWSYMBOLSIZE=50mil|DI
=0000000000000000000000000000000000000000000000000000000000000000000000
|RECORD=PrinterOptions|DEVICE=|DRIVER=|OUTPUT=|SHOWHOLES=FALSE|SCALETOFITP
|PP20=1|PP21=1|PP22=1|PP23=1|PP24=1|PP25=1|PP26=1|PP27=1|PP28=1|PP29=1|PP3
402|PM40=7697781|RECORD=PrinterOptions|PM41=7895160|PM42=8092539|PM43=8289
PC22=16711935|PC23=32896|PC24=65535|PC25=8421504|PC26=16777215|PC27=838873
|RECORD=GerberOptions|SORTOUTPUT=FALSE|USESOFTWAREARCS=TRUE|CENTERPHOTOPLO
|RECORD=AdvancedPlacerOptions|PLACELARGECLER=50mil|PLACESMALLCLER=20mil|I
|RECORD=DesignRuleCheckerOptions|DOMAKEDRCFILE=TRUE|DOMAKEDRCERRORLIST=TRUE
|RECORD=PinSwapOptions|QUIET=FALSE|APPROXIMATEPINPOSITIONS=FALSE|ALLOWPART
|RECORD=TestpointOptions|TESTPOINTSEARCHORDERSMDPADSBTM=0|TESTPOINTSEARCHOI
|RECORD=Class|INDEXFORSAVE=0|SELECTION=FALSE|LAYER=MULTILAYER|LOCKED=FALSE
|RECORD=C]ass|INDEXFORSAVE=1|SELECTION=FALSE|LAYER=MULTILAYER|LOCKED=FALSE
```

Рис. 2.1. Фрагмент документ вида *Altium PCB ASCII (\*.PcbDoc)* открытый на чтение.

Каждая запись, в свою очередь, имеет свой идентификатор («RECORD=<идентификатор записи>») и набор параметров, разделенных символом «|». Записи разделяются переводом строки. Может существовать несколько записей с одинаковыми идентификаторами записи, но с разными параметрами.

Далеко не все записи требуется разобрать для решения задачи теплового

расчета. Необходимые записи имеют следующие идентификаторы записи:

7. «Board» - информация о печатной плате;
8. «Component» - информация об элементе печатной платы;
9. «ComponentBody» - информация о корпусе элемента печатной платы;

Область конструирования *PCB Editor* привязана к координатным осям X и Y размерности мил (1 мил = 1/1000 дюйма = 0,0254 мм). Каждый элемент конструирования (лист, область печатной платы, элементы на печатной плате, надписи на листе и др.) имеет собственные координаты.

Область печатной платы строится по вершинам с заданными координатами. Количество и координаты каждой вершины можно узнать из записи «Board» по параметрам «VX<номер вершины>=<координата X, мил>» и «VY<номер вершины>=<координата Y, мил>». Зная координаты вершин области печатной платы, можно легко найти её размеры.

Печатная плата, как правило, является многослойной. САПР Altium Designer 10 поддерживает конструирование печатных плат с максимально возможными 82 слоями (не считая самый верхний и самый нижний слой). Чтобы узнать толщину печатной платы, необходимо сложить все толщины используемых слоев печатной платы. Параметры записи «Board», необходимые для данного подсчета:

- ⌚ «LAYER<номер слоя>MECHENABLED=<TRUE или FALSE>» - определяет, используется («TRUE») или нет («FALSE») механический слой под заданным номером;
- ⌚ «LAYER<номер слоя>DIELHEIGHT=<толщина, мил>» - толщина слоя под заданным номером;
- ⌚ «LAYER<номер слоя>COPTHICK=<толщина, мил>» - толщина от слоя под заданным номером до следующего слоя;
- ⌚ «TOPHEIGHT=<толщина, мил>» - толщина самого верхнего слоя;
- ⌚ «BOTTOMHEIGHT=<толщина, мил>» - толщина самого низкого слоя.

Следующие параметры записи «Board» определяют материал, из

которого сделан слой печатной платы:

- ⌚ «LAYER<номер слоя>DIELMATERIAL=<название материала>» - материал, из которого сделан слой под заданным номером;
- ⌚ «TOPMATERIAL=<название материала>» - материал, из которого сделан самый верхний слой;
- ⌚ «BOTTOMMATERIAL=<название материала>» - материал, из которого сделан самый нижний слой.

Каждая запись «Component» имеет параметр «ID=<номер>». Этот параметр служит для выделения уникальности каждого элемента печатной платы. Каждая запись «ComponentBody», в свою очередь имеет параметр «COMPONENT=<номер>». Параметры «ID» записи «Component» и «COMPONENT» записи «ComponentBody» тождественны.

Определение длины и ширины корпуса элемента печатной платы аналогично нахождению размеров печатной платы (по параметрам вершин записи «ComponentBody» - «VX<номер вершины>=<координата X, мил>» и «VY<номер вершины>=<координата Y, мил>»).

За толщину элемента печатной платы отвечает параметр записи «ComponentBody» - «OVERALLHEIGHT=<толщина, мил>».

Следующие параметры записи «Component» представляют необязательную, но полезную информацию об элементе печатной платы:

- ⌚ «PATTERN=<название>» - название используемого шаблона элемента;
- ⌚ «SOURCEDESIGNATOR=<указатель>» - указатель, обозначающий элемент печатной платы (например, значение для резистора: R1, R2);
- ⌚ «SOURCELIBREFERENCE=<название>» - библиотечное название элемента печатной платы;
- ⌚ «SOURCEDESCRIPTION=<описание>» - библиотечное описание элемента печатной платы;
- ⌚ «FOOTPRINTDESCRIPTION=<описание>» - описание элемента печатной

платы для монтажа.

Подведём итоги. Чтобы произвести автоматизированный расчет теплового режима печатной платы, необходимо программно разобрать документ типа *PCB Editor*, экспортированный в виде *Altium PCB ASCII (\*.PcbDoc)*. При разборе следует сохранить в памяти компьютера значения следующих параметров:

3. запись «Board»:

1. «VX<номер вершины>»;
2. «VY<номер вершины>»;
3. «LAYER<номер слоя>MECHENABLED»;
4. «LAYER<номер слоя>DIELHEIGHT»;
5. «LAYER<номер слоя>COPTHICK»;
6. «TOPHEIGHT»;
7. «BOTTOMHEIGHT»;
8. «LAYER<номер слоя>DIELMATERIAL»;
9. «TOPMATERIAL»;
- 10.«BOTTOMMATERIAL»;

4. запись «Component»:

1. «ID»;
2. «PATTERN»;
3. «SOURCEDESIGNATOR»;
4. «SOURCELIBREFERENCE»;
5. «SOURCEDESCRIPTION»;
6. «FOOTPRINTDESCRIPTION»;

5. запись «ComponentBody»:

1. «COMPONENT»;
2. «VX<номер вершины>»;
3. «VY<номер вершины>»;
4. «OVERALLHEIGHT».

## 2.3 Проектирование программного комплекса

До этого параграфа производился анализ задачи автоматизации процесса расчета теплового режима печатной платы под САПР Altium Designer. Перейдем к проектированию программного комплекса, решающего анализируемую задачу.

Согласно техническому заданию, проектирование будет проводиться с использованием объектно-ориентированного подхода и языка программирования Java.

Разрабатываемую систему (программный комплекс) можно разделить на три составляющих (компонента) (рис. 2.2):

- ⌚ графический интерфейс пользователя (на рис. GraphicUserInterface) – ввод исходных данных, отображение рассчитанного теплового режима печатной платы;
- ⌚ разбор исходных данных (на рис. AltiumPcbDocParser);
- ⌚ расчет теплового режима печатной платы (на рис. PcbThermalConditionsAnalyzer).

Компоненты системы связаны между собой интерфейсами. Компонент GraphicUserInterface использует интерфейсы Browserable и Propertiable для сопряжения с компонентом AltiumPcbDocParser. Компонент PcbThermalConditionsAnalyzer использует интерфейсы IPcbModelForTca и IPcbElementModelForTca для сопряжения с компонентом AltiumPcbDocParser. В свою очередь, компонент AltiumPcbDocParser реализует интерфейсы Browserable, Propertiable, IPcbModelForTca и IPcbElementModelForTca.

Рассмотрим сценарий типичного использования разрабатываемого программного комплекса. Первым делом пользователь задает исходный файл - файл конструкторской документации печатной платы (*Altium PCB ASCII (\*.PcbDoc)*). Компонент GraphicUserInterface запрашивает у компонента AltiumPcbDocParser разбор исходного файла в соответствии с интерфейсами

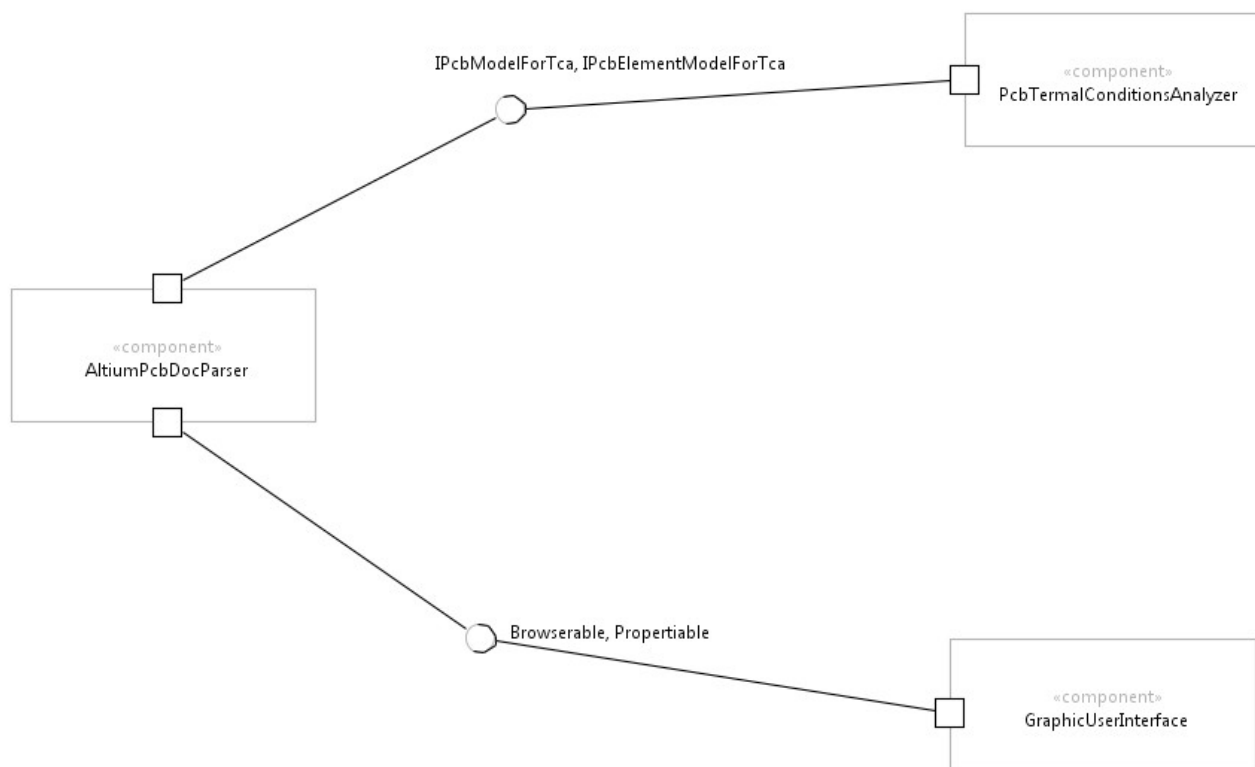


Рис. 2.2. UML диаграмма компонентов системы.

**Browserable** (для рисования компонентов) и **Propertiable** (для хранения параметров). Разобранные параметры печатной платы и её элементов отображаются пользователю. При необходимости, пользователь задает дополнительные параметры рассчитываемой печатной платы. Затем пользователь инициализирует расчет теплового режима. Компонент **PcbTermalConditionsAnalyzer** с использованием интерфейсов **IPcbModelForTca** и **IPcbElementModelForTca** принимает разобранные параметры печатной платы и её элементов соответственно. Этот же компонент производит расчет теплового режима печатной платы. Наконец, компонент **GraphicUserInterface** отображает результаты расчета теплового режима печатной платы, полученные от компонента **PcbTermalConditionsAnalyzer**.

Согласно проекту, объект сгенерированной модели печатной платы хранится в единственном экземпляре, в компоненте **AltiumPcbDocParser**.

Объект модели печатной платы отвечает за реализацию интерфейсов `Browsable`, `Propertiable`, `IPcbModelForTca` и `IPcbElementModelForTca`. Это позволяет использовать один и тот же объект в разных компонентах (`AltiumPcbDocParser` и `PcbTermalConditionsAnalyzer`). Так как объект хранится в единственном экземпляре, то анализатор `PcbTermalConditionsAnalyzer` лишь изменяет прежние параметры температур элементов модели на рассчитанные.

Таким образом, компоненты системы получаются мало связанными друг с другом. Это позволит в будущем наращивать систему новыми анализаторами, а также совершенствовать разбор исходных данных (`AltiumPcbDocParser`) без изменения других частей программного комплекса.

Далее более подробно будет рассматриваться проектирование каждого компонента разрабатываемой системы.

### 2.3.1 Проектирование компонента `AltiumPcbDocParser`

Роль компонента `AltiumPcbDocParser` - разобрать документ типа *PCB Editor*, экспортированный в виде *Altium PCB ASCII (\*.PcbDoc)*, на составляющие его параметры печатной платы. UML диаграмма классов `AltiumPcbDocParser` изображена на рис. 2.3.

Как было показано в п. 2.2, из документа типа *PCB Editor* необходимо извлечь информацию о печатной плате и об элементах, расположенных на печатной плате. Соответственно, вводятся объекты печатной платы (`PcbModel`) и элемента печатной платы (`PcbElementModel`). За управление разбором информации отвечает класс `AltiumPcbDocParser`.

Обозначено, что компонент `AltiumPcbDocParser` должен реализовывать интерфейсы `Browsable`, `Propertiable`, `IPcbModelForTca` и `IPcbElementModelForTca`. Интерфейсы `Browsable` и `Propertiable` используются компонентом `GraphicUserInterface`, а `IPcbModelForTca` и `IPcbElementModelForTca` — компонентом `PcbTermalConditionsAnalyzer`.



Рассмотрим каждый интерфейс подробно.

Интерфейс `Browserable`. Этот интерфейс служит для того, чтобы компонент `GraphicUserInterface` смог отобразить некий графический объект пользователю. Методами этого интерфейса являются:

- ⌚ `getPeak` - возвращает массив точек-узлов, необходимых для масштабирования графического объекта;
- ⌚ `getProperties` - возвращает список отображаемых пользователю параметров графического объекта в виде списка объектов, поддерживающих интерфейс `Propertiable`;
- ⌚ `getVisualizationShape` - возвращает объект-фигуру графического объекта;
- ⌚ `getBrowserableObjects` - возвращает список дочерних графических объектов (эти объекты так же поддерживают интерфейс `Browserable`);
- ⌚ `getDescription` - возвращает строку-описание данного графического объекта, которая отображается при наведении мышью на этот объект;
- ⌚ `getLocation` - возвращает координаты графического объекта.
- ⌚ `setLocation` – устанавливает координаты графического объекта;
- ⌚ `getDimension` – возвращает габариты графического объекта.

Интерфейс `Propertiable`. По этому интерфейсу компонент `GraphicUserInterface` отображает пользователю параметр графического объекта. Интерфейс полностью реализовывается классом `PcbObjectProperty`. Список методов интерфейса следующий:

- ⌚ `getName` – возвращает название параметра;
- ⌚ `getValue` – возвращает значение параметра;
- ⌚ `setValue` – устанавливает значение параметра;
- ⌚ `getViewComponent` – возвращает способ отображения значения параметра (например, изменяемая строка, либо «галочка»).

Интерфейсы `IPcbModelForTca` и `IPcbElementModelForTca`. Эти интерфейсы используются компонентом `PcbThermalConditionsAnalyzer` для

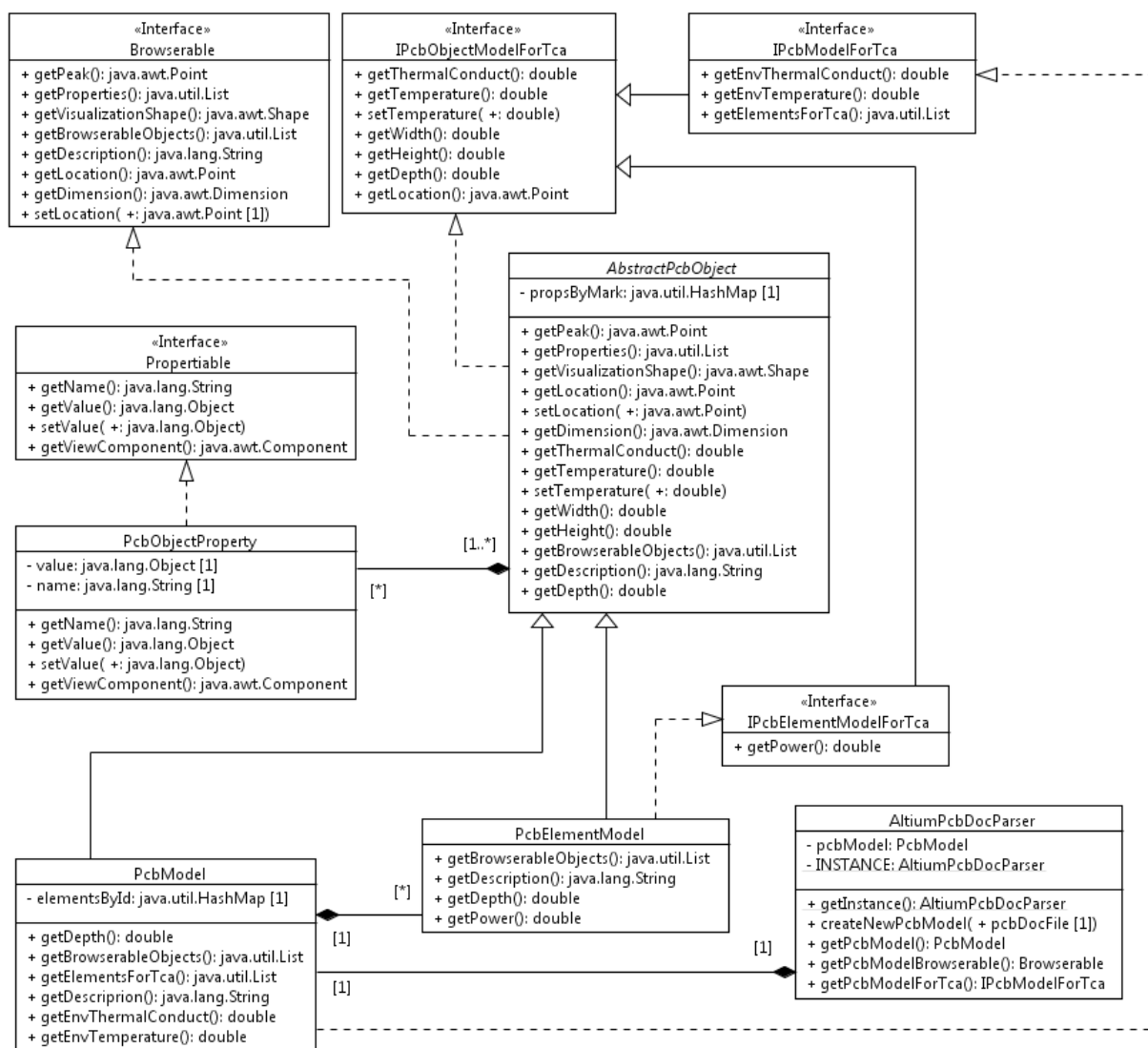


Рис. 2.3. UML диаграмма классов компонента AltiumPcbDocParser.

проведения расчета теплового режима печатной платы. IPcbModelForTca служит для представления печатной платы, а IPcbElementModelForTca — для представления элемента печатной платы. Так как многие параметры, необходимые для расчета, схожи у печатной платы и элемента печатной платы, то целесообразно объединить эти интерфейсы в общий, который будет отвечать за схожие параметры. Объединенным интерфейсом является IPcbObjectModelForTca. Методы общего интерфейса:

🕒 `getThermalConduct` – возвращает коэффициент теплопроводности объекта;

- ⌚ `getTemperature` — возвращает температуру объекта;
- ⌚ `getWidth` — возвращает ширину объекта;
- ⌚ `getHeight` — возвращает длину объекта;
- ⌚ `getDepth` — возвращает высоту объекта;
- ⌚ `getLocation` — возвращает координаты объекта.

Дополнительные методы интерфейса `IPcbModelForTca` (представление печатной платы):

- ⌚ `getEnvThermalConduct` — возвращает коэффициент теплопроводности окружающей среды;
- ⌚ `getEnvTemperature` — возвращает температуру окружающей среды;
- ⌚ `getElementsForTca` — возвращает список элементов печатной платы (поддерживающих интерфейс `IPcbElementModelForTca`).

Дополнительным методом интерфейса `IPcbElementModelForTca` (представление элемента печатной платы) является метод `getPower` — возвращает мощность элемента печатной платы.

Перейдём к описанию объектов печатной платы (`PcbModel`) и элемента печатной платы (`PcbElementModel`), упомянутых в начале данного параграфа. На этапе построения интерфейсов компонента `PcbThermalConditionsAnalyzer` было замечено, что многие параметры схожи у печатной платы и элемента печатной платы. Что касается интерфейса `Browsable` — согласно этому интерфейсу нет разницы, что отображать — печатную плату, либо элемент печатной платы — будет отображаться некий графический объект. Исходя из этого было принято решение реализовывать схожие параметры в абстрактном классе `AbstractPcbObject`.

Абстрактный класс `AbstractPcbObject`. Этот абстрактный класс хранит в себе список параметров отображаемого объекта (список `Propertiable`) и обслуживает эти параметры (добавление, изменение, удаление). Причем, как показано на диаграмме классов (рис. 2.3) связь списка параметров и объекта `AbstractPcbObject` физическая, то есть без существования объекта `AbstractPcbObject`, не существует объектов `Propertiable`.

AbstractPcbObject реализует все методы интерфейсов Browserable и IPcbObjectModelForTca, кроме методов (они являются абстрактными):

- ⌚ getBrowserableObjects – исходя из того, что элемент печатной платы не может иметь дочерних элементов;
- ⌚ getDescription — из-за особенностей структуры документа типа *PCB Editor*;
- ⌚ getDepth — из-за особенностей структуры документа типа *PCB Editor*.

Класс PcbElementModel наследуется от AbstractPcbObject (реализуя абстрактные методы) и дополнительно реализует интерфейс IPcbElementModelForTca.

Класс PcbModel так же, как и PcbElementModel, наследуется от AbstractPcbObject (реализуя абстрактные методы), но теперь реализовывается другой дополнительный интерфейс - IPcbModelForTca. Стоит отметить, что объекты PcbElementModel физически включаются в объект PcbModel (см. рис. 2.3).

Как было сказано выше, за управление разбором информации отвечает класс AltiumPcbDocParser. А это значит, что за создание объекта модели печатной платы (PcbModel), также отвечает этот класс. Методы класса AltiumPcbDocParser:

- ⌚ createNewPcbModel – создаёт объект модели печатной платы по переданному файлу документа типа *PCB Editor*;
- ⌚ getPcbModel — возвращает созданный объект модели печатной платы в виде PcbModel;
- ⌚ getPcbModelBrowserable — возвращает созданный объект модели печатной платы в виде Browserable (для компонента GraphicUserInterface);
- ⌚ getPcbModelForTca — возвращает созданный объект модели печатной платы в виде IPcbModelForTca (для компонента PcbTermalConditionsAnalyzer);

Созданный объект печатной платы типа PcbModel в единственном экземпляре физически включается в объект класса AltiumPcbDocParser, который

также создаётся в единственном экземпляре. Это означает, что при создании новой модели печатной платы, старая модель стирается из памяти компьютера.

### 2.3.2 Проектирование компонента `PcbThermalConditionsAnalyzer`

Компонент `PcbThermalConditionsAnalyzer` служит для проведения анализа теплового режима печатной платы. Главный класс компонента имеет то же имя, что и компонент — `PcbThermalConditionsAnalyzer`. Этот класс и использует, описанные в п. 2.3.1, интерфейсы `IPcbModelForTca` и `IPcbElementModelForTca`. UML диаграмма классов компонента `PcbThermalConditionsAnalyzer` изображена на рис. 2.4. Точнее будет сказать, что класс `PcbThermalConditionsAnalyzer` использует интерфейс `IPcbModelForTca`, а интерфейс `IPcbModelForTca` использует интерфейс `IPcbElementModelForTca`. Отметим ещё раз, что оба этих интерфейса наследуются от общего для них интерфейса `IPcbObjectModelForTca`.

Процесс анализа представляет следующую последовательность действий:

- ⌚ компонент разбирает файл документа типа *PCB Editor* и сохраняет модель печатной платы в объект типа `PcbModel`, поддерживающий и реализующий интерфейсы `Browsable`, `Propertiable`, `IPcbModelForTca` и `IPcbElementModelForTca`;
- ⌚ компонент `GraphicUserInterface`, используя интерфейсы `Browsable` и `Propertiable`, отображает пользователю объект модели печатной платы `PcbModel`;
- ⌚ компонент `GraphicUserInterface` передаёт объект модели печатной платы `PcbModel` компоненту-анализатору `PcbThermalConditionsAnalyzer` по интерфейсам `IPcbModelForTca` и `IPcbElementModelForTca`;
- ⌚ класс `PcbThermalConditionsAnalyzer` компонента `PcbThermalConditionsAnalyzer` производит расчет теплового режима модели печатной платы, изменяя параметры объекта `PcbModel`;

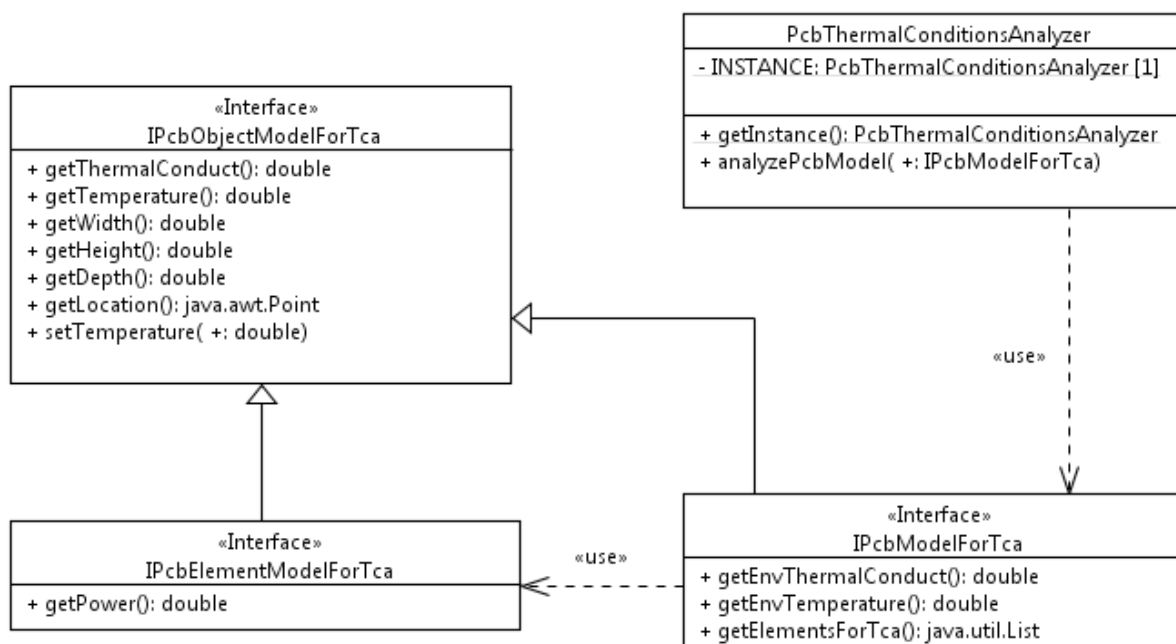


Рис. 2.4. UML диаграмма классов компонента PcbThermalConditionsAnalyzer.

⌚ наконец, компонент GraphicUserInterface, используя интерфейсы Browserable и Propertiable, снова отображает пользователю объект модели печатной платы PcbModel, но теперь уже с рассчитанным тепловым режимом;

Процесс расчета производится по методу численного решения модели с сосредоточенными параметрами, описанному в п. 1.3.

### 2.3.3 Проектирование компонента GraphicUserInterface

Компонент GraphicUserInterface представляет собой графический пользовательский интерфейс, созданный на основе технологии Swing, входящей в состав инструментария для Java2. Swing предоставляет набор графических компонентов для создания пользовательских интерфейсов любой сложности, а также вспомогательные средства и классы для работы с этими компонентами. Компоненты Swing не нуждаются в поддержке операционной системы, и поэтому интерфейс будет корректно работать независимо от того, в какой

системе запущена программа.

UML диаграмма классов компонента `GraphicUserInterface` представлена на рис 2.5. Класс `BrowserFrame`, являющийся потомком `JFrame`, реализует главное окно программы, в котором расположены меню, панель инструментов и рабочая область. Для быстрого доступа к этим компонентам он имеет ряд методов:

- ⌚ `getBrowserMenuBar` – возвращает ссылку на объект меню;
- ⌚ `getBrowserToolBar` – возвращает ссылку на объект панели инструментов;
- ⌚ `getBrowserWorkPanel` – возвращает ссылку на объект рабочей области программы.

Как меню, так и панель инструментов может содержать различное количество разнообразных компонентов предоставляющих пользователю необходимый набор инструментария.

Пункты меню “Open file” и “Start analysis” реализованы с помощью классов `OpenFileAction` и `StartAnalysisAction` соответственно. Метод `actionPerformed` этих классов отвечает за действия, выполняемые при нажатии на кнопку или пункт меню. Реализация этого метода в классе `OpenFileAction` представляет собой создание и отображение пользователю диалогового окна `OpenFileDialog` (в котором можно выбрать необходимый файл), инициирование разбора документа *Altium PCB ASCII (\*.PcbDoc)* для получения объекта модели платы и его передачи объекту рабочей области программы. В случае класса `StartAnalysisAction` этот метод запускает анализатор.

Класс `BrowserWorkPanel`, реализующий рабочую область, расширяет класс `JPanel`. Он содержит объекты редактора свойств, области отображения модели печатной платы и саму модель, а также предоставляет методы для доступа к ним:

- ⌚ `setBrowserableObject` – передает ссылку на объект печатной платы;
- ⌚ `setPropertyEditorPanel` – передает ссылку на объект редактора свойств;
- ⌚ `setBrowserPaintPanel` – передает ссылку на объект области рисования;

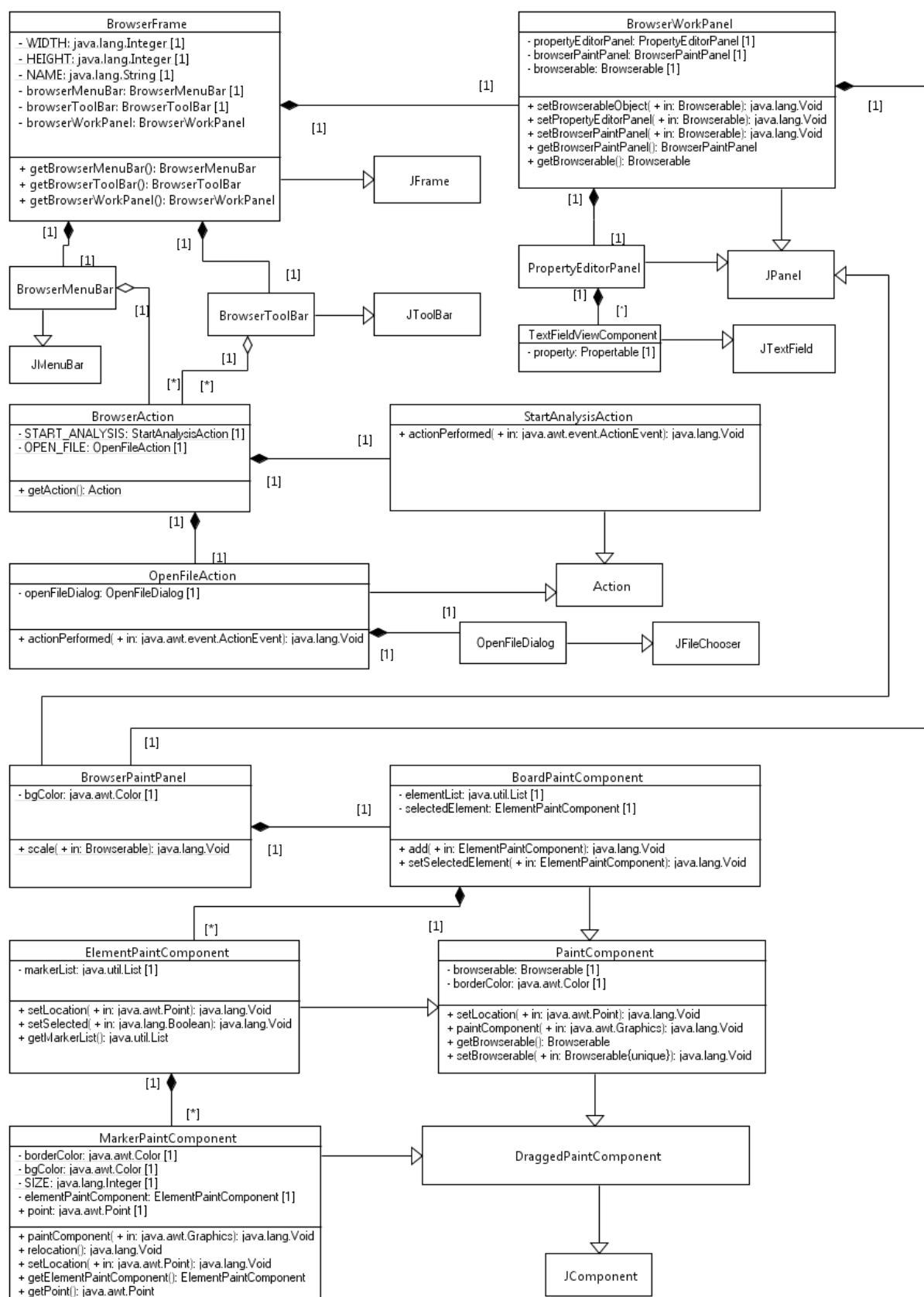


Рис. 2.5. UML диаграмма классов компонента GraphicUserInterface.



- ⌚ `getBrowserPaintPanel` – возвращает ссылку на объект области рисования;
- ⌚ `getBrowserable` – возвращает ссылку на модель печатной платы.

Для реализации редактора свойств потребовалось создать класс `PropertyEditorPanel` расширяющий компонент `Jpanel`, входящий в состав `Swing`. Конструктор класса `PropertyEditorPanel` получает объект, реализующий интерфейс `Browserable`, который имеет методы для доступа к свойствам, таким как ширина и высота. Для каждого свойства создается пара текстовых полей, содержащих имя свойства и его значение соответственно. Текстовое поле, содержащее имя свойства неизменяемое и представляет собой экземпляр стандартного класса `JTextField`, в то время как для значения свойства создается экземпляр класса `TextFieldViewComponent`, в котором реализована обработка событий, возникающих при изменении содержимого пользователем.

Наконец, для графического отображения модели печатной платы создан класс `BrowserPaintPanel`. Он играет роль контейнера для компонента графической модели печатной платы. Также класс имеет единственный статический метод для изменения масштаба.

Класс `BoardPaintComponent` реализует графическую модель печатной платы на основе объекта, реализующего интерфейс `Browserable`. Экземпляр этого класса содержит список элементов расположенных на плате, который формируется при его создании, а также ссылку на элемент, считающийся выделенным пользователем с помощью мыши. Стоит отметить, что он расширяет класс `PaintComponent`, речь о котором пойдет ниже. Методы данного класса следующие:

- ⌚ `add(ElementPaintComponent c)` – специализированный метод для добавления компонентов типа `ElementPaintComponent` с их предобработкой;
- ⌚ `setSelectedElement` – устанавливает ссылку на выбранный элемент.

Элементы печатной платы представляют собой экземпляры класса `ElementPaintComponent`, каждый из которых имеет список объектов `MarkerPaintComponent`, созданием которого и занимается специализированный

метод `add(ElementPaintComponent c)` класса `BoardPaintComponent`. Данный класс также является потомком `PaintComponent`, в котором реализованы механизмы, являющиеся общими для графических моделей печатной платы и элементов. Методы класса `ElementPaintComponent`:

- ⌚ `setLocation` – выполняет необходимые вычисления при изменении позиции элемента, после чего вызывает метод супер класса;
- ⌚ `setSelected` – устанавливает признак того что элемент является выделенным;
- ⌚ `getMarkerList` – возвращает ссылку на список маркеров.

Как уже было сказано, предком классов `ElementPaintComponent` и `BoardPaintComponent` является класс `PaintComponent`. Он хранит ссылку на объект, реализующий интерфейс `Browserable`, который нужен для получения доступа к информации, необходимой для рисования компонента. В его функции входит рисование компонента, а также выполнение всей рутинной работы, возникающей при изменении положения объекта в контейнере, в котором он содержится, изменении размеров и так далее. Этот класс является предком класса `DraggedPaintComponent`. Методы класса `PaintComponent`:

- ⌚ `setLocation` – устанавливает новую позицию компонента, вычисляет новое положение маркеров и так далее;
- ⌚ `paintComponent` – рисует компонент;
- ⌚ `getBrowserable` – устанавливает ссылку на объект, реализующий интерфейс `Browserable`;
- ⌚ `setBrowserable` - возвращает ссылку на объект, реализующий интерфейс `Browserable`.

Реализация маркеров, с помощью которых можно менять размеры элементов содержится в классе `MarkerPaintComponent`. Экземпляры этого класса создаются для каждого элемента, исходя из количества точек необходимых для рисования элемента и их значений. Каждый объект этого класса имеет ссылки на объект элемента, которому принадлежит, и одну из вершин этого элемента. Этот класс также является потомком класса

DraggedPaintComponent. Методы класса MarkerPaintComponent:

- ⌚ paintComponent – отвечает за рисование маркера, задает его графический вид;
- ⌚ relocation – вычисляет действительное положение маркера на основе значения вершины;
- ⌚ setLocation – устанавливает новое положение маркера и пересчитывает значение точки элемента которому принадлежит;
- ⌚ getElementPaintComponent – возвращает ссылку на объект элемента;
- ⌚ getPoint – возвращает объект представляющий собой значение вершины к которой прикреплен маркер.

Наконец, механизм реализующий возможность перетаскивания элементов, маркеров и печатной платы реализован в классе DraggedPaintComponent, который расширяет стандартный класс JComponent, входящий в состав пакета Swing.

## **2.4 Анализ существующих сред разработки программного обеспечения**

Для рациональной работы по созданию программного обеспечения для процесса автоматического расчета теплового режима печатной платы требуется выбрать подходящую среду разработки программного обеспечения. Рассмотрено несколько популярных сред, поддерживающих разработку с использованием языка программирования Java (по убыванию популярности): Eclipse от Eclipse Foundation, IntelliJ IDEA от JetBrains, NetBeans IDE от NetBeans Org., JBuilder от Embarcadero, JDeveloper от Oracle.

Среда разработки Eclipse.

Eclipse – это расширяемая, open-source (с открытым кодом, свободно распространяемая) интегрированная среда разработки. Этот проект был запущен в ноябре 2001 года, когда IBM выделили исходный код из Websphere Studio Workbench, ценой в 40 миллионов долларов, и сформировали Eclipse

Consortium для управления разработкой этого инструмента.

Первоначальная цель заключалась в том, чтобы "разработать стабильную, полнофункциональную, промышленную платформу, качества коммерческого продукта, для разработки интегрируемых инструментов". На настоящий момент Eclipse Consortium фокусируется на следующих проектах:

1. The Eclipse Project (<http://www.eclipse.org/eclipse/index.html>) – ответственен за разработку непосредственно Eclipse IDE (платформу для сборки прочих инструментов Eclipse), Java Development Tools (JDT) и Plug-In Development Environment (PDE), используемую для предоставления возможности расширения самой платформы.
2. The Eclipse Tools Project (<http://www.eclipse.org/tools/index.html>) – занимается созданием оптимальных инструментальных средств для платформы Eclipse. В текущие подпроекты входят: Cobol IDE, C/C++ IDE, а также инструмент моделирования EMF.
3. The Eclipse Technology Project (<http://www.eclipse.org/technology/index.html>) – занимается технологическими исследованиями, инкубацией и образованием по части использования платформы Eclipse.

Совместно с JDT, платформа Eclipse предоставляет множество различных возможностей, которые можно наблюдать в коммерческих IDE: подсветка синтаксиса в редакторе, компиляция кода, отладчик уровня исходного кода с поддержкой "нитей" (threads), навигатор по классам, файловый менеджер и менеджер проектов, интерфейсы для стандартных контролирующих систем исходного кода, таких как, например, CVS и ClearCase.

Помимо этого Eclipse содержит ряд уникальных возможностей, например, рефакторинг кода (<http://www.refactoring.com/>), автоматическое обновление и сборка кода (посредством Update Manager), список задач, поддержка возможности тестирования модулей с помощью JUnit (<http://www.junit.org/>), а также интеграция с инструментом сборки приложений Jakarta Ant (<http://jakarta.apache.org/ant/index.html>).

Несмотря на большое количество стандартного набора возможностей, Eclipse отличается от традиционных IDE по нескольким фундаментальным особенностям. Может быть, самая интересная возможность Eclipse – это абсолютная нейтральность относительно платформы и языка программирования. Вдобавок к эклектичному набору языков программирования, которые поддерживаются Eclipse Consortium (Java, C/C++, Cobol), существует множество сторонних проектов, с помощью которых вы можете обеспечить поддержку интересующего вас языка программирования в Eclipse. На сегодняшний день существуют реализации следующих популярных языков программирования: Python, Eiffel, PHP, Ruby, и C#.

Платформа Eclipse предоставляется, благодаря Eclipse Consortium, в виде заранее скомпилированных исполняемых файлов для Windows, Linux, Solaris, HP-UX, AIX, QNX, и Mac OS X. Очень много внимания концентрируется вокруг архитектурной системы plug-in'ов (расширений) этой платформы, а также "богатых" API (Application Programming Interface), поставляемых с Plug-in Development Environment для расширения Eclipse. Добавить поддержку нового типа редактора, просмотрщика (панели) или языка программирования до безобразия просто, благодаря хорошо спроектированным API и строительным блокам, которые предоставляет Eclipse.

Имея приблизительно сотню проектов, по разработке plug-in'ов, имея ресурсы, предоставляемые такими промышленными гигантами, как IBM, HP и Rational (которая недавно была куплена IBM), и дизайн Erich Gamma, который помогает быстро разобраться в процессе разработки – Eclipse выглядит весьма ярким, интересным и перспективным.

Среда разработки IntelliJ IDEA.

IntelliJ IDEA — коммерческая интегрированная среда разработки программного обеспечения на многих языках программирования, в частности Java, JavaScript, Python и др., разработанная компанией JetBrains.

Первая версия IntelliJ IDEA появилась в январе 2001 года и быстро приобрела популярность, как первая Java IDE с широким набором

интегрированных инструментов для рефакторинга, которые позволяли программистам быстро реорганизовывать исходные тексты программ. Дизайн среды ориентирован на продуктивность работы программистов, позволяя им сконцентрироваться на разработке функциональности, в то время как IntelliJ IDEA берет на себя выполнение рутинных операций.

Начиная с шестой версии продукта IntelliJ IDEA предоставляет интегрированный инструментарий для разработки графического пользовательского интерфейса.

Среди прочих возможностей, IntelliJ IDEA хорошо совместима со многими популярными свободными инструментами разработчиков, такими как CVS, Subversion, Apache Ant, Maven и JUnit. В феврале 2007 года разработчики IntelliJ анонсировали раннюю версию плагина для поддержки программирования на языке Ruby.

Начиная с версии 9.0, IntelliJ IDEA доступна в двух версиях: Community Edition и Ultimate Edition. Community Edition является полностью свободной версией, доступной под лицензией Apache 2.0. В ней реализована полная поддержка Java SE, Groovy, Scala, а также интеграция с наиболее популярными системами управления версиями. В версии Ultimate Edition реализована поддержка Java EE, UML-диаграмм, подсчет покрытия кода, а также поддержка других систем управления версиями, языков и фреймворков.

Среда разработки NetBeans IDE.

NetBeans IDE — свободная интегрированная среда разработки приложений (IDE) на языках программирования Java, JavaFX, Python, PHP, JavaScript, C, C++, Ада и ряда других.

Проект NetBeans IDE поддерживается и спонсируется компанией Oracle, однако разработка NetBeans ведется независимым сообществом разработчиков-энтузиастов (NetBeans Community) и компанией NetBeans Org.

По качеству и возможностям последние версии NetBeans IDE не уступают лучшим коммерческим (платным) интегрированным средам разработки для языка Java, таким, как IntelliJ IDEA, поддерживая рефакторинг,

профилирование, выделение синтаксических конструкций цветом, автодополнение набираемых конструкций на лету, множество предопределённых шаблонов кода и др.

Для разработки программ в среде NetBeans и для успешной инсталляции и работы самой среды NetBeans должен быть предварительно установлен Sun JDK или J2EE SDK подходящей версии. Среда разработки NetBeans по умолчанию поддерживала разработку для платформ J2SE и J2EE. Начиная с версии 6.0 NetBeans поддерживает разработку для мобильных платформ J2ME, C++ и PHP без установки дополнительных компонентов.

В версии NetBeans IDE 6.1 декларируется поддержка UML, SOA, языка программирования Ruby (включая поддержку Ruby on Rails), а также средства для создания приложений на J2ME для мобильных телефонов. В версии 6.5 добавлена поддержка языка PHP. Также для тестирования выложен модуль поддержки Python.

NetBeans IDE поддерживает плагины, позволяя разработчикам расширять возможности среды. Одним из самых популярных плагинов является мощный дизайнер отчётов iReport (основанный на библиотеке JasperReports).

На идеях, технологиях и в значительной части на исходном коде NetBeans IDE базируются предлагаемые фирмой Sun коммерческие интегрированные среды разработки для Java — Sun Java Studio Creator, Sun Java Studio Enterprise и Sun Studio (для ведения разработки на C, C++ или Фортран). Сравнительно недавно Sun стала предлагать эти среды разработки бесплатно для зарегистрировавшихся в Sun Developer Network (SDN) разработчиков, сама же регистрация на сайте бесплатна и не требует никаких предварительных условий, кроме согласия с лицензией CDDL.

NetBeans IDE доступна в виде готовых дистрибутивов (прекомпилированных бинарных файлов) для платформ Microsoft Windows, Linux, FreeBSD, Mac OS X, OpenSolaris и Solaris (как для SPARC, так и для x86 — Intel и AMD). Для всех остальных платформ доступна возможность скомпилировать NetBeans самостоятельно из исходных текстов.

В релизе NetBeans IDE 6.7 была добавлена интеграция с Project Kenai, поддержка языка Groovy и веб-фреймворка Grails. В версии 6.8 — поддержка PHP-фреймворка Symfony, а в 6.9 — Zend Framework.

В версии 7 производитель отказался от поддержки языка Ruby и Ruby on Rails, объясняя этот отказ большим объемом работ по поддержке Java 7 и сравнительно невысокой потребностью в этом языке и технологии среди пользователей NetBeans (компонент Ruby and Rails был взят на сопровождение сообществом и продолжает существовать). Таким образом, «из коробки» последние версии NetBeans поддерживают только Java (включая Java FX, Java ME, Java EE), C/C++, Groovy, PHP, HTML, JavaScript, CSS. Также для версии 7 не имеется ни собственных, ни бесплатных сторонних компонентов поддержки UML-моделирования.

Среда разработки JBuilder.

Embarcadero JBuilder — проприетарная среда разработки на языке Java компании Embarcadero (в дальнейшем Oracle), входит в состав Embarcadero RAD Studio. Среда разработана в соответствии с концепцией визуального программирования.

Последняя версия — «Embarcadero JBuilder 2008 R2». По утверждению разработчиков, JBuilder является средой разработки промышленного класса, основанной на open-source каркасе Eclipse.

JBuilder 2008 выпускается в двух версиях: JBuilder 2008 Professional и JBuilder 2008 Enterprise. Последняя дополнительно включает в себя средства профилирования использования памяти и процессора, настройки производительности, базовые средства UML-моделирования, средства преобразования код—диаграммы и наоборот, и проч.

Среда разработки JDeveloper.

JDeveloper — бесплатная интегрированная среда разработки программного обеспечения, разработанная корпорацией Oracle. Предоставляет возможность для разработки на языках программирования Java, XML, SQL и PL/SQL, HTML, JavaScript, BPEL и PHP. JDeveloper покрывает весь жизненный



цикл разработки программного обеспечения от проектирования, кодирования, отладки, оптимизации и профилирования до его развёртывания.

Производитель отмечает в качестве основной задачи среды — максимальное использование возможностей визуального и декларативного подхода к разработке программного обеспечения в дополнение к удобной среде кодирования. Oracle JDeveloper интегрирована с Oracle Application Development Framework — Java EE-каркасом для создания коммерческих приложений на Java.

До версии 11g JDeveloper поставлялся в трёх редакциях (от младшей к старшей): Java Edition, J2EE Edition и Studio Edition. Каждая более старшая редакция включает возможности младшей, все редакции поставлялись бесплатно. JDeveloper 11g имеет только две редакции: Java Edition и Studio Edition (возможности J2EE Edition внесены в Studio Edition).

JDeveloper написана полностью на Java, поэтому работает на всех операционных системах, имеющих JDK.

### Выбор IDE

Все рассмотренные IDE реализуют базовый набор возможностей для объектно-ориентированного проектирования и программирования на языке Java. Так же все они бесплатные (по крайней мере есть возможность установить бесплатную версию). Тем не менее, выгоднее описанных, смотрится среда Eclipse. Во-первых данная среда очень хорошо поддерживается разработчиком — обновления и исправления ошибок среды поставляются достаточно часто. Во-вторых имеется мощная система расширений (plug-in'ов), что позволяет настроить среду практически под любую современную задачу (вплоть до построения кода из UML-диаграмм). В-третьих Eclipse содержит в себе привлекательную систему рефакторинга (реорганизация кода по всему проекту), которая достаточно эффективно экономит время программистов.

### 3. ЭКСПЕРИМЕНТАЛЬНО-ТЕСТОВАЯ ЧАСТЬ

Для оценки надёжности и эффективности программного обеспечения проведены ряд экспериментов (табл. 1), по каждому пункту проводилось не менее десяти испытаний четырьмя экспертами.

Таблица 1

Цели тестирования

Эксперимент	Цель
Подача на вход программы ложных документов	Проверить устойчивость программы к нестандартному случаю.
Использование обрабатываемого документа во время работы программы.	Исследование способности программы к обеспечению информационной безопасности.
Подача на вход программы документов с неполной информацией о ПП и её элементах.	Исследовать способность программы обработать нестандартный документ.
Сравнение теплового режима ПП, рассчитанного с применением программы и рассчитанного человеком	Исследование качества расчета теплового режима ПП проводимого с использованием программы.
Измерение времени, требуемого человеку и программе для расчета теплового режима печатной платы	Исследование эффективности программы.

Эксперименты проводились с использованием портативного персонального компьютера Toshiba Satellite L775-12F. Его характеристики:

- ⌚ операционная система Windows 7 Home Premium;
- ⌚ процессор Intel Core i5 2300 МГц;
- ⌚ оперативная память 4096 Мб DDR3 1333 МГц;
- ⌚ графический чипсет NVIDIA GeForce GT 525M 2048 Мб GDDR3.

Руководство пользователя разработанного ПО предоставлено в приложении.

После проведения описанных экспериментов (табл. 1) получились следующие результаты:

1. Подача на вход программы ложных документов.

Как указывалось в п. 2.1, для последующего программного разбора документа печатной платы САПР Altium Designer 10, он должен быть типа *PCB Editor*, экспортированный в виде *Altium PCB ASCII (\*.PcbDoc)*. Этот документ имеет свой особый формат (см. п. 2.2), так что ложными считаются документы, имеющие отличный от данного формат. Пустые файлы также считаются ложными.

В случае задания ложных документов реакция программы следующая: будет запущен разбор документа, но никакой информации из этих документов извлечено не будет. Пользователю будет отображаться пустой компонент 2D изображения ПП и набор свойств ПП со значениями по умолчанию (рис. 3.1).

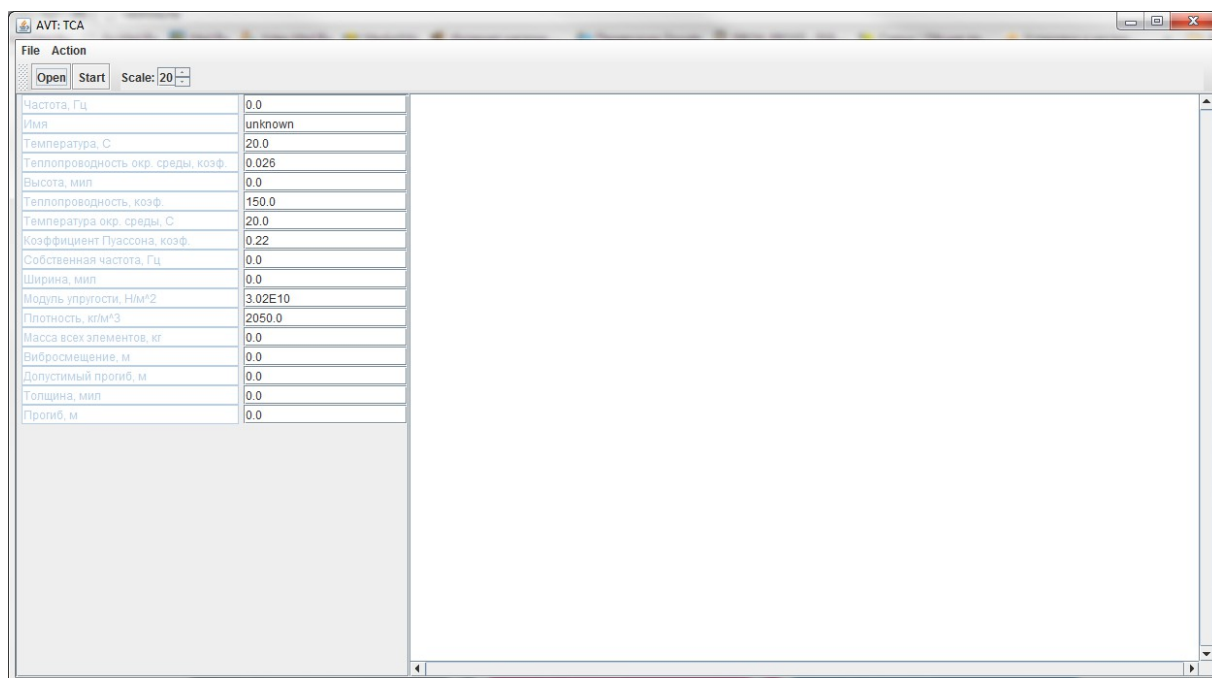


Рис. 3.1. Интерфейс программного комплекса в случае задания ложных документов

Изменение отображённых в таком случае свойств, так же как и инициирование анализа теплового режима ПП ни к чему не приведёт.

Стоит отметить, что в программном комплексе присутствует тривиальная защита от задания ложного документа — фильтр расширения файла в диалоговом окне выбора файла *Altium PCB ASCII (\*.PcbDoc)*.

В ходе проведения данного эксперимента не обнаружено ни одного сбоя ПО. В итоге можно сказать, что программа устойчива к этому нестандартному типу воздействия.

## 2. Использование обрабатываемого документа во время работы программы.

Программный комплекс сконструирован так, что исходный файл во-первых, открывается только на чтение, а во-вторых читается лишь один раз, после чего использование его прекращается. При чтении разобранная информация записывается в соответствующую модель ПП или модель элемента ПП.

При проведении данного эксперимента, исходный файл конструкторской документации ПП *Altium PCB ASCII (\*.PcbDoc)* был открыт и использовался в САПР Altium Designer 10. Никаких конфликтов данных обнаружено не было, анализ документа и теплового режима ПП выполнялись успешно. Работа над проектом ПП в САПР Altium Designer 10 также продолжалась успешно.

По итогу эксперимента сделан вывод о том, что программный комплекс обеспечивает информационную безопасность исходного файла *Altium PCB ASCII (\*.PcbDoc)*.

## 3. Подача на вход программы документов с неполной информацией о ПП и её элементах.

Зачастую, разрабатывая печатную плату в САПР Altium Designer используются библиотеки элементов, в которых информация об элементах неполна. Критическими показателями для анализатора теплового режима печатной платы являются информация о габаритах и о пространственном расположении.

На вход программы было подано 2 типа документов: с неполной информацией о ПП, с неполной информацией о некоторых элементах ПП. В случае неполной информации о ПП, а именно информации о габаритах, картина будет схожа с рис. 3.1. В случае неполной информации о некоторых элементах ПП:

- ⌚ при отсутствии информации о габаритах, принимаются значения по умолчанию: ширина — 100 мил, высота — 100 мил, толщина — 0 мил (рис. 3.2, у выделенного элемента отсутствует информация о габаритах);
- ⌚ при отсутствии информации о пространственном расположении элементов, такие элементы располагаются в левом верхнем углу ПП (рис. 3.3, у выделенного элемента отсутствует информация о пространственном расположении).

В обоих случаях ввод правильных значений возлагается на пользователя.

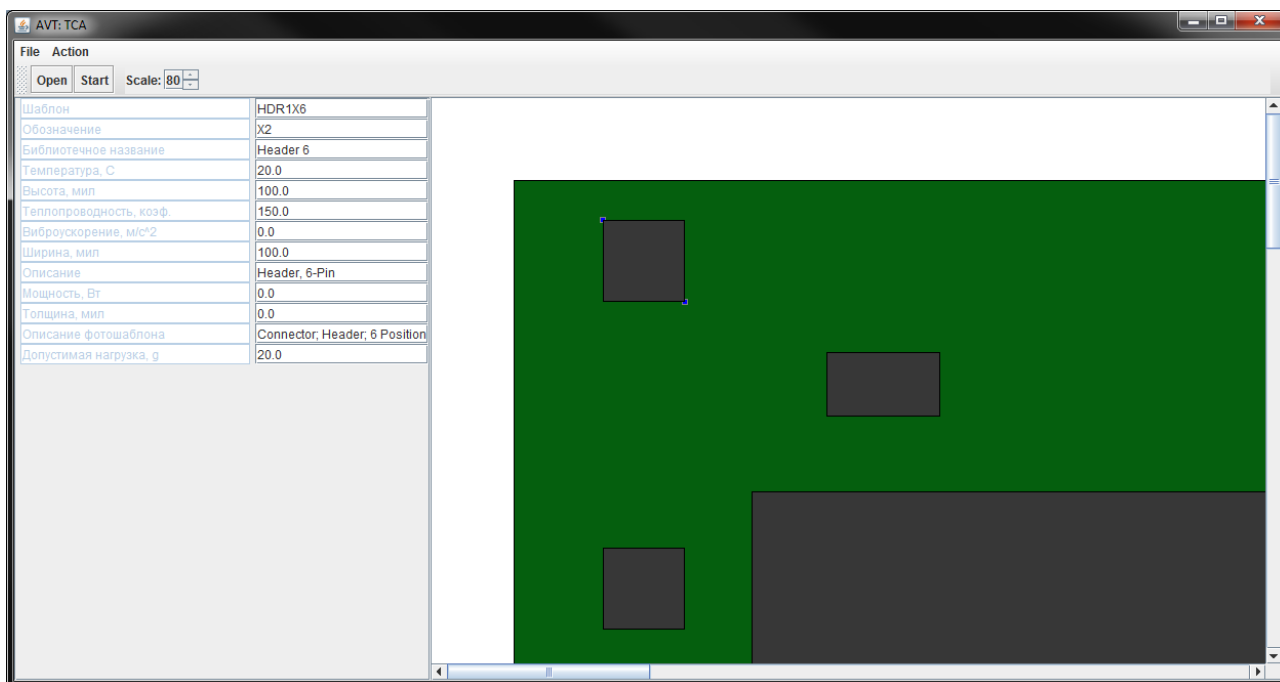


Рис. 3.2. Отображение элемента, у которого отсутствует информация о габаритах.

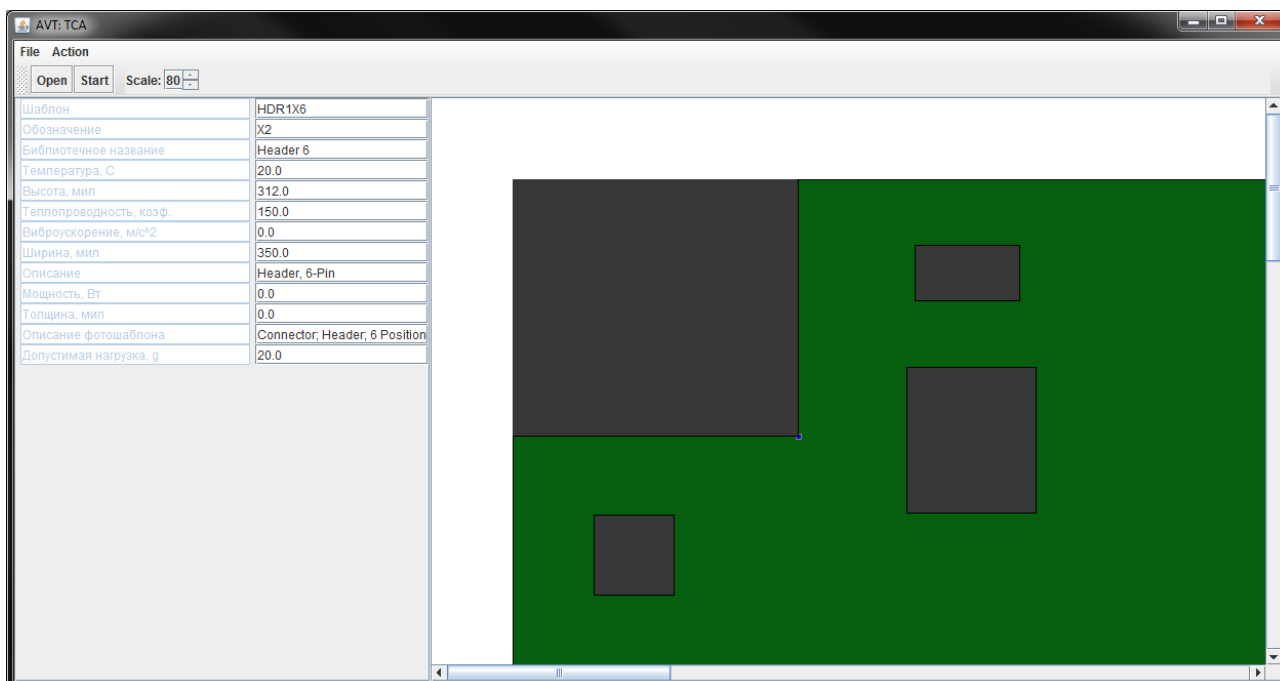


Рис. 3.3. Отображение элемента, у которого отсутствует информация о пространственном расположении.

В итоге программный комплекс обработал все документы с неполной информацией, обеспечив информационную целостность. Во всех рассмотренных случаях, кроме неполной информации о габаритах ПП, после ввода недостающих значений, был успешно проведён анализ теплового режима ПП.

#### 4. Сравнение теплового режима ПП, рассчитанного с применением программы и рассчитанного человеком.

Анализ теплового режима ПП программным комплексом проводится по методу, описанному в п. 1.3. Для эксперимента был взят проект САПР Altium Designer печатной платы, фрагмент принципиальной схемы которой изображен на рис. 3.4. Это некий цифровой блок. Изображение фрагмента сборочного чертежа приведен на рис. 3.5. На плате располагаются 19 элементов, а именно:

- ⌚ 7 резисторов номиналом 1 кОм, мощность 2 Вт;
- ⌚ 5 конденсаторов номиналом 100 пФ, мощность 2 Вт;



- ⌚ 2 электрических соединителя;
- ⌚ логический элемент 6 НЕ ЭКФ1553ЛН1, мощностью 2 Вт;
- ⌚ логический элемент 4 2И-НЕ ЭКФ1553ЛА3, мощностью 2 Вт;
- ⌚ логический элемент 6 повторителей ЭКФ1553ЛП16, мощностью 2 Вт;
- ⌚ память EEPROM AT49F001, мощностью 2 Вт;
- ⌚ контроллер SWITCH EPM7192SQC160, мощностью 10 Вт.

Экспортированный файл *Altium PCB ASCII (\*.PcbDoc)* задаётся как исходный в программный комплекс. Результат разбора приведён на рис. 3.6. Не у всех элементов ПП достаточно информации о параметрах, а именно не заданы значения мощностей; тепловых проводимостей сред, элементов и ПП; температуры окружающей среды. Все эти данные были уточнены и введены в программу. Значение температуры окружающей среды (воздух) принято 20°C.

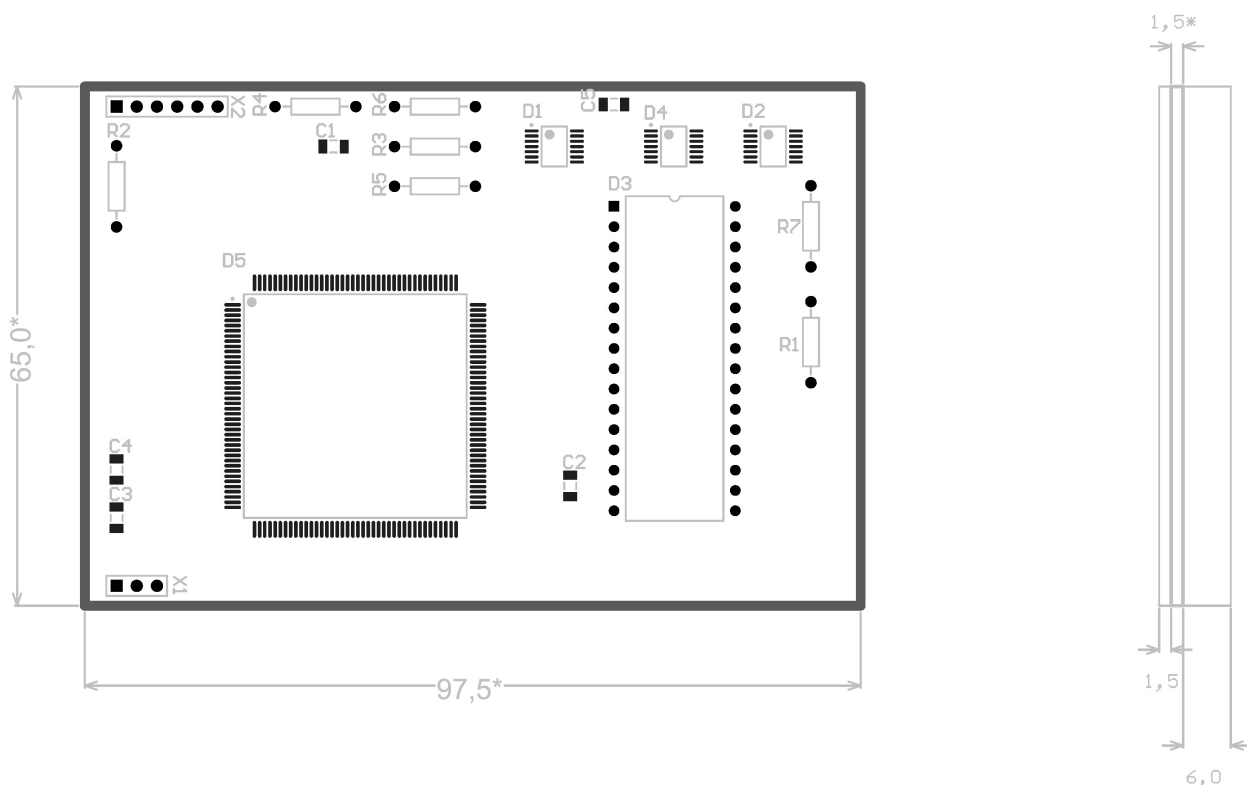


Рис. 3.5 Фрагмент сборочного чертёжа печатной платы проекта цифрового блока САПР Altium Designer.



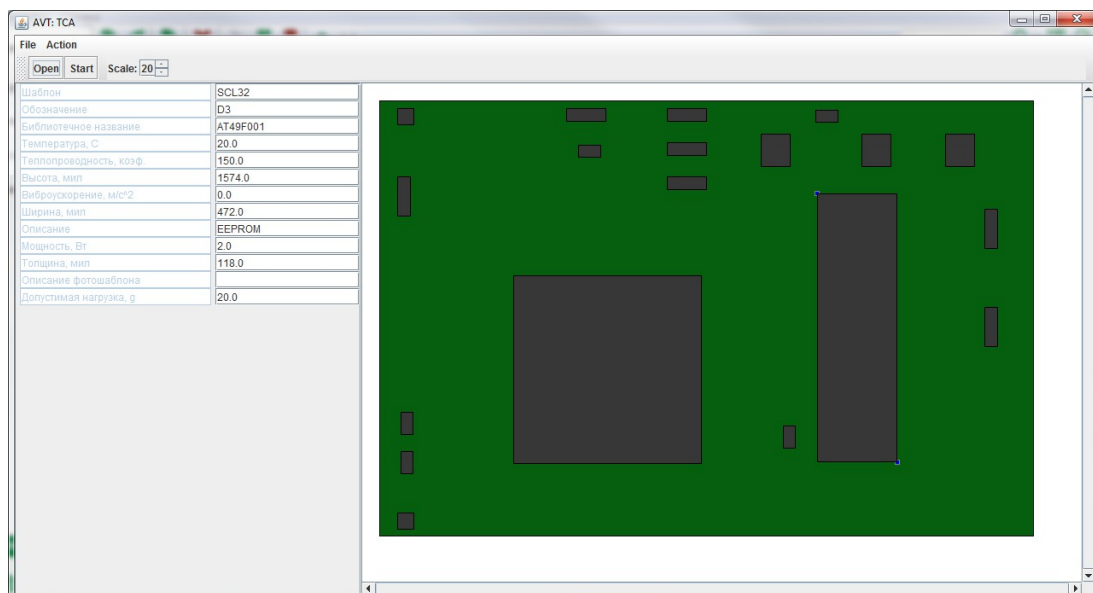


Рис. 3.6. Результат разбора экспортированного файла *Altium PCB ASCII* (\*.PcbDoc).

Был произведён анализ теплового режима ПП программой и вручную (аналитическим методом). Результат работы программы приведён на рис. 3.7. В итоге получились значения, отличающиеся друг от друга не более, чем на 5°C. Это доказывает приемлемое качество программного продукта, что позволяет его использовать для автоматизированного расчета теплового режима ПП.

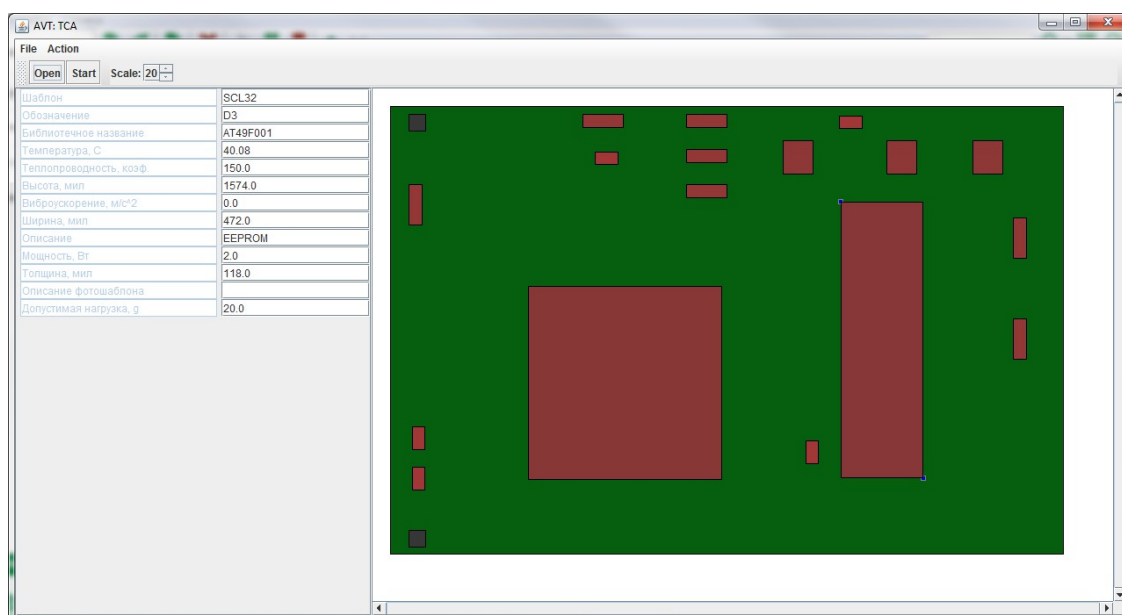


Рис. 3.7. Результат разбора анализа теплового режима программой.

5. Измерение времени, требуемого человеку и программе для расчета теплового режима ПП.

Наряду с качеством, эффективность программного комплекса является основным показателем успешности разработки. Для эксперимента экспортировались несколько проектов ПП из САПР Altium Designer: с 5ю элементами, с 10ю элементами, с 20 элементами и с 50ю элементами.

В любом случае, эксперту потребовалось во много раз больше времени для проведения расчета теплового режима вручную. Причём, проводя расчет непрерывно, растёт процент вероятности ошибок. Затраченное время на расчет проектов печатных плат приведено в табл. 2.

Таблица 2

Затраченное время

Количество элементов на ПП, шт.	Затраченное на расчет время с использованием программы, минут	Затраченное на расчет время без использования программы, минут
5	1,5	25
10	4	60
20	9	90
50	20	150

По табл. 2 можно построить график зависимости потраченного времени (в минутах) на проведение расчета вручную и с помощью программы от количества элементов на ПП. Этот график, со спрогнозированными значениями для более сложных проектов приведён на рис. 3.8.

По графику видно, что с помощью разработанной программы возможен расчет теплового режима печатной платы со значительным выигрышем по времени. Сделан вывод, что программа достаточно эффективна для расчета теплового режима ПП.

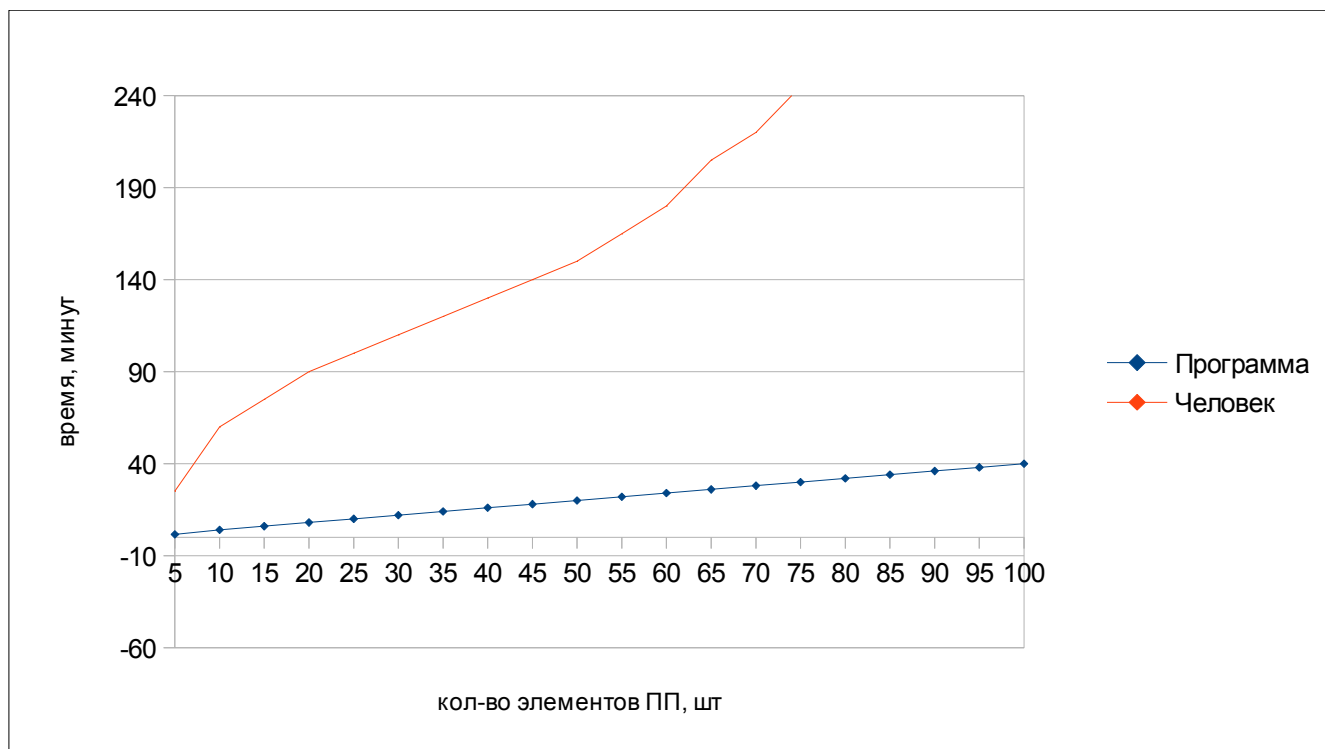


Рис. 3.8 График зависимости потраченного времени (в минутах) на проведение расчета вручную и с помощью программы от количества элементов на ПП.

## **ЗАКЛЮЧЕНИЕ**

В работе были рассмотрены методы расчета теплового режима печатной платы, проведен обзор существующих систем автоматизации анализа теплового режима печатной платы, исследован метод расчета теплового режима печатной платы с использованием модели со сосредоточенными параметрами. Написан программный комплекс автоматизации процесса расчета теплового режима печатной платы. Программа спроектирована с использованием объектно-ориентированного подхода и написана на языке Java в среде разработки Eclipse. ПО удовлетворяет всем требованиям, которые были предъявлены. Однако функционал должен быть повышен в следующих редакциях данной разработки, так как существуют и другие модели для расчета теплового режима, которые более точны. А также в данной разработке не учтены потоки теплоносителей. Программный комплекс поможет инженерам-конструкторам в оценке теплового режима печатной платы.

## ЛИТЕРАТУРА

1. Дульнев Г.Н. Тепловые режимы электронной аппаратуры/Учебное пособие для студентов высших технических заведений. – Л.: «Энергия», 1971 г. – 248 с.
2. Жаднов В.В., Сарафанов А.В. Управление качеством при проектировании теплонагруженных радиоэлектронных средств. - М.: СОЛОН-Пресс, 2004. - 464 с.: ил. (Серия «Библиотека инженера»)
3. Дульнев Г.Н. и др. Методы расчета теплового режима приборов/Г.Н. Дульнев, В.Г. Парфенов, А.В. Сигалов. - М.: Радио и связь, 1990. - 312 с.:ил.
4. Суходольский В.Ю. - Altium Designer. Проектирование функциональных узлов РЭС на печатных платах. — СПб.: БХВ-Петербург, 2010. — 480 с.
5. Дульнев Г.Н. Тепло- и массообмен в радиоэлектронной аппаратуре: Учебник для вузов по спец. «Конструир. и произв. Радио-аппаратуры». - М.: Высш. шк., 1984. - 247 с., ил.
6. Гради Буч, Объектно-ориентированный анализ и проектирование с примерами приложений (3-е издание): Вильямс, 2008. - 721 с.
7. Кей Хорстманн, Гари Корнелл, Java 2. Библиотека профессионала. Том 1. Основы (8-е издание): Вильямс, 2010. — 816 с.
8. Кей Хорстманн, Гари Корнелл, Java 2. Библиотека профессионала. Том 2. Тонкости программирования (8-е издание): Вильямс, 2010. — 992 с.
9. Исследование тепловых характеристик РЭС методами математического моделирования: Монография /В.В.Гольдин и др.; Под ред. А.В.Сарафанова. - М.:Радио и связь, 2003
- 10.Пирогова Е.В., Проектирование и технология печатных плат: Учебник. — М.: ФОРУМ: ИНФРА-М, 2005. — 560 с.
- 11.Суходольский В.Ю. Altium Designer. Проектирование функциональных

- узлов РЭС на печатных платах. — СПб.: БХВ-Петербург, 2010. — 480 с.
12. Ожегов С.И. Словарь русского языка/ Под ред. И.Ю. Шведовой.- М.:Русский язык, 1983.

## ИСТОЧНИКИ В INTERNET

<http://translate.google.com/> - Переводчик Google;

<http://ru.wikipedia.org/> - Википедия – свободная энциклопедия;

<http://gostexpert.ru/> - ГОСТ Эксперт - база ГОСТов РФ;

[http://www.platan.ru/shem/pdf/beta\\_soft.pdf](http://www.platan.ru/shem/pdf/beta_soft.pdf) — Программное обеспечение анализа тепловых режимов печатных плат BetaSoft-Board, Схемотехника №3 март 2001, Владимир Стешенко;

[http://www.javaportal.ru/java/ide/intro\\_eclipse.html](http://www.javaportal.ru/java/ide/intro_eclipse.html) — Введение в интегрированную среду разработки Eclipse;

<http://wiki.altium.com/display/ADOH/Importing+and+Exporting+Design+Files> — Altium Designer 10. Importing and Exporting Design Files – Altium Designer 10. Импорт и экспорт файлов конструкторской документации.

[http://www.eurointech.ru/products/Altium/Protel99SE\\_PCB\\_ASCII\\_File\\_Format.pdf](http://www.eurointech.ru/products/Altium/Protel99SE_PCB_ASCII_File_Format.pdf) - PCB ASCII File Format Reference – Описание формата файла PCB ASCII

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ  
ВЫСШЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ**

**“САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ  
УНИВЕРСИТЕТ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ,  
МЕХАНИКИ И ОПТИКИ”**

**Факультет компьютерных технологий и управления**

**ПРИЛОЖЕНИЕ. РУКОВОДСТВО ПОЛЬЗОВАТЕЛЯ**

**Разработка программного комплекса автоматизированного расчета  
теплового режима печатной платы**

**Разработчик: Ягодаров А.Е.  
студент гр. 6159**

**Санкт-Петербург  
2013**



## О программе

Программный комплекс автоматизированного расчета теплового режима печатной платы.

ПО предназначено инженерам-конструкторам для оценки теплового режима печатной платы, сконструированной в САПР Altium Designer 10. Программа генерирует температурную карту элементов печатной платы, анализируя исходные документы. Исходным документом является файл *Altium PCB ASCII (\*.PcbDoc)*, экспортированный из САПР Altium Designer 10. Путь к этому документу задаётся пользователем.

К дополнительным функциям программы относятся отображение свойств элементов и самой печатной платы в удобном виде, 2D просмотр печатной платы с возможностью масштабирования и изменения габаритных и пространственных характеристик ПП и её элементов. Интерфейс программного комплекса представлен на рис.4.1.

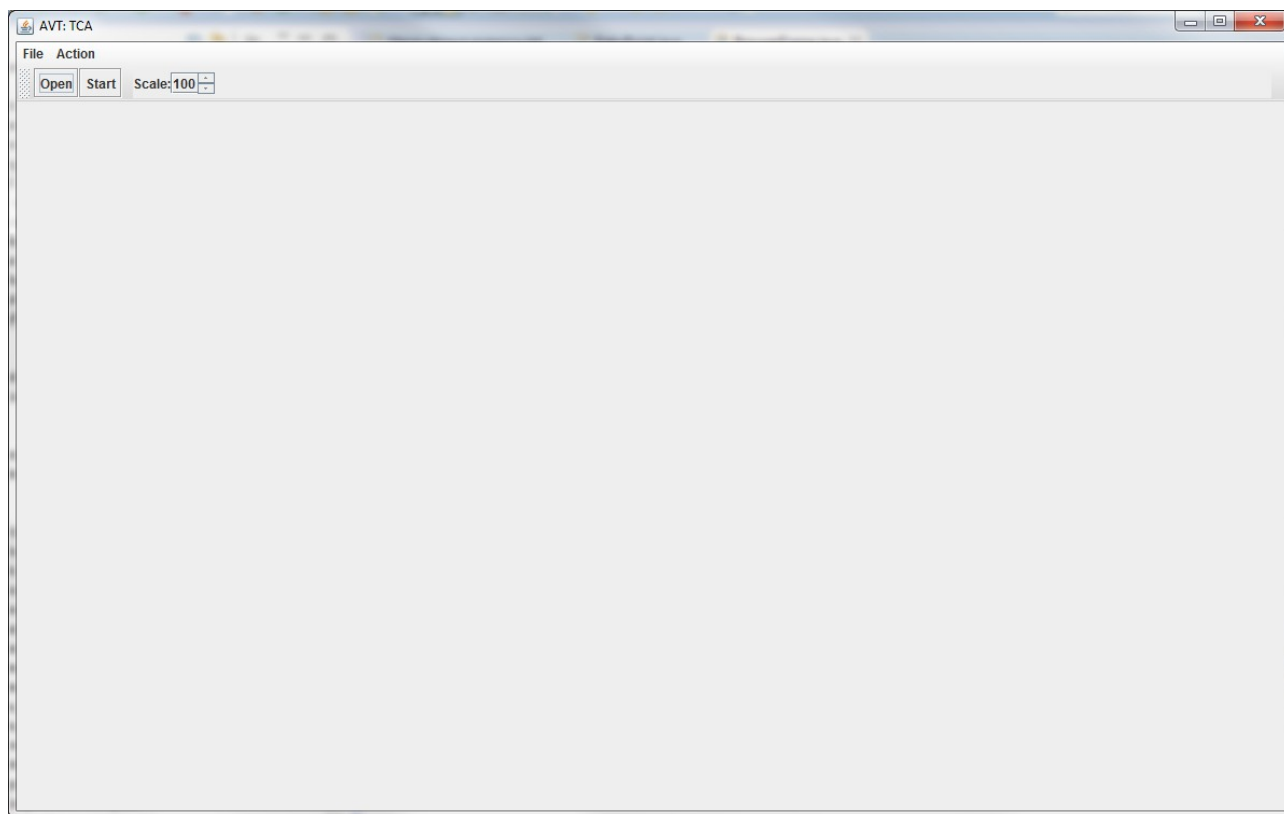


Рис.4.1 Интерфейс программного комплекса

Взаимодействие с пользователем ведётся с помощью диалогового окна выбора файла; списка изменяемых параметров ПП и её элементов; режима 2D просмотра ПП; кнопок масштабирования; контекстной информации.

ПО рассчитано на работу в любой платформе с необходимым программным обеспечением (JRE 7). Требования к компьютеру: Pentium Celeron, 256Мб ОЗУ, около 10 Мб на жёстком диске, клавиатура или манипулятор типа «мышь», монитор.

#### Задание исходного документа

Чтобы задать файл *Altium PCB ASCII (\*.PcbDoc)*, необходимо либо совершить действие «**File** → **Open**», либо нажать кнопку «**Open**». Любое из этих действий вызовет диалоговое окно выбора исходного файла. После того, как файл будет выбран, программа начнёт его разбор. Разобранный файл отобразится пользователю в виде 2D изображения печатной платы и её свойств (рис. 4.2). Нажимая на стрелки около надписи «Scale» изменит масштаб отображения ПП.

#### Изменение свойств печатной платы и её элементов

Чтобы изменить свойства ПП либо её элемента, изначально нужно щёлкнуть указателем на изменяемый объект 2D изображения ПП. Любое свойство объекта можно изменить редактируя список свойств (слева на рис. 4.2). Однако габаритные и пространственные свойства элемента ПП, можно также изменять непосредственно на 2D изображении ПП (перетаскивая объект указателем — изменять положение; перетаскивая левый верхний, либо правый нижний маркер — изменять длину и ширину).

#### Анализ теплового режима печатной платы

Если исходный файл задан и нажата кнопка «**Start**» (либо произведено действие «**Action** → **Start**»), начнётся анализ теплового режима печатной

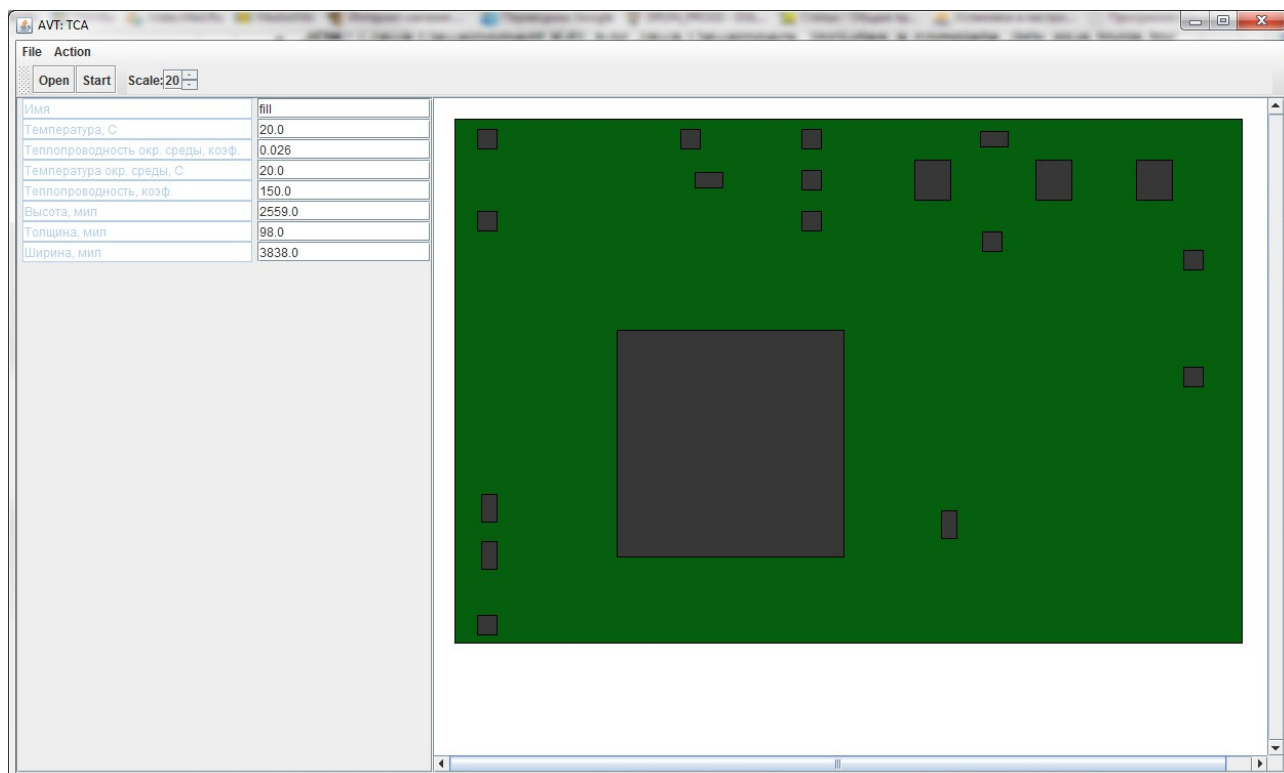


Рис.4.2 Интерфейс программного комплекса после задания файла *Altium PCB ASCII (\*.PcbDoc)*

платы. После того, как анализ завершится, пользователю снова отобразится 2D изображение печатной платы, её элементов со свойствами. Однако теперь цвета элементов изменятся относительно рассчитанной температуры (рис. 4.3). На этом основная работа программы закончится.

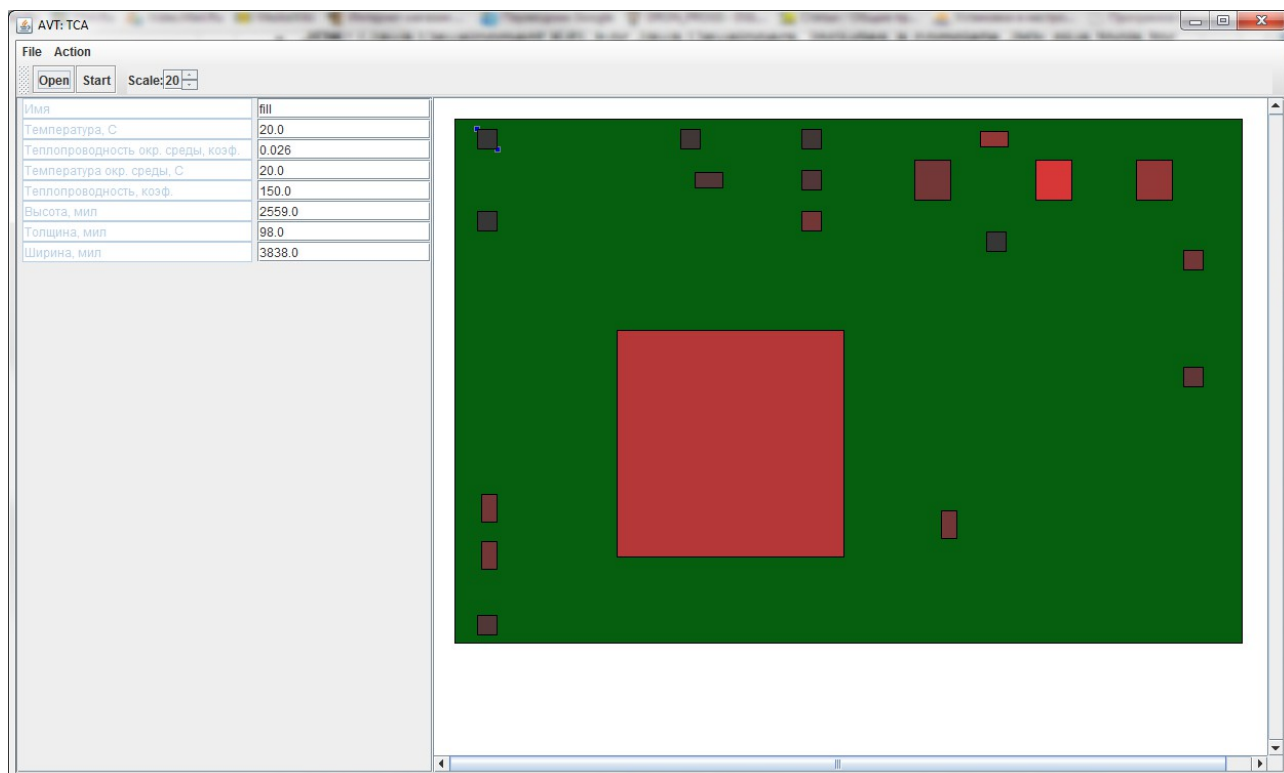


Рис. 4.3. Интерфейс программного комплекса после проведения анализа теплового режима ПП.