

ЗАДАЧИ АВТОМАТИЗАЦИИ КОНСТРУКТОРСКОГО ЭТАПА ПРОЕКТИРОВАНИЯ РАДИОЭЛЕКТРОННОЙ АППАРАТУРЫ

Л.И.Макаров

Потребности современного развития науки и техники приводят к необходимости разработки и создания сложных устройств вычислительной техники и радиоэлектроники. С возрастанием сложности радиоэлектронной аппаратуры резко увеличивается трудоемкость процесса ее проектирования, и этот процесс становится невозможным без применения систем автоматизированного проектирования (САПр). САПр радиоэлектронной аппаратуры (РЭА) должна охватывать все этапы процесса проектирования (структурное, логическое, электрофизическое, конструкторское, техническое, надежностное проектирование и т.д.), поскольку только комплексная автоматизация всех этапов позволит эффективно получать качественные результаты проектирования.

При автоматизации проектирования РЭА на каждом этапе возникает целый комплекс сложных оптимизационных задач, учитывающих специфику данного этапа. Поэтому возникает необходимость рассмотрения круга задач, относящихся к каждому этапу проектирования, выделения из них основных и разработки эффективных алгоритмов их решения. На этапе конструкторского проектирования к таким задачам можно отнести задачи размещения элементов и трассировки их соединений. При формализации процесса проектирования эти задачи можно поставить следующим образом [1,2].

§1. Постановка задач

Радиоэлектронное устройство может быть описано схемой F , которая задается в виде пары (E, S) , где $E = \{e_i\}$ - множество элементов схемы, $i = \overline{1, n}$, а $S = \{s_j\}$ - система их соединений (связок), $j = \overline{1, m}$.

Предположим, что элементы e_i есть прямоугольники в евклидовой плоскости, заданные в собственных системах координат (X_i, Y_i) , $i = \overline{1, n}$, и, кроме того, в системе координат (X_0, Y_0) плоскости задан прямоугольник (плата) P и целочисленная решетка $\hat{H}$.

Определим ориентацию θ_i элемента e_i как угол поворота системы (X_i, Y_i) вместе с элементом e_i вокруг ее начала (репера) t_i относительно системы координат (X_0, Y_0) . Предположим далее, что ориентация $\theta_i = k \cdot 90^\circ$, $k = 1, 2, \dots$, и репер $t_i \in H_i = e_i \cap H$, $i = \overline{1, n}$.

Тогда для множества элементов E схемы F считаем заданным на плате P размещение $R = \{R_i\}$, $R_i = (x_i, y_i, \theta_i)$, если для каждого элемента e_i , $i = \overline{1, n}$, известны ориентация θ_i , координаты (x_i, y_i) его репера t_i в системе (X_0, Y_0) и $H_E = \bigcup_{i=1}^n H_i \subseteq H = R \cap \hat{H}$; $H_i \cap H_r = \emptyset$, $i \neq r$; $i, r = \overline{1, n}$. Через $\hat{R}$ обозначим множество всех размещений элементов из E на плате P .

Каждая связь $s_j \in S$ содержит множество тех точек (контактов) элементов, которые на плате должны быть соединены трассой между собой, т.е. $s_j = \{(k, i)_{q_j}\}$, $q_j = \overline{1, q_j}$, где k - номер контакта элемента e_i , входящего в связку s_j , $k \in \{1, 2, \dots, k_i\}$, $q_j \geq 1$, $s_j \cap s_r = \emptyset$, $j \neq r$; $j, r = \overline{1, m}$, $\sum_{j=1}^m q_j = \sum_{i=1}^n k_i = K$.

Конечную решетку H можно рассматривать как взвешенный граф $H = (V, U)$, множеством V вершин которого является множество узлов, а множеством U ребер - множество отрезков единичной длины, соединяющих соседние узлы решетки. Для заданного размещения R элементов схемы известны координаты каждого контакта всех ее элементов, поэтому для каждой связки s_j известно множество $V_j \subseteq H$, $j = \overline{1, m}$, узлов решетки, соответствующих ее контактам.

Трассой T_j связки s_j при заданном размещении R назовем любой связный подграф-дерево графа H , содержащий множество узлов V_j , т.е. $T_j = (\hat{V}_j, U_j)$, $\hat{V}_j \subseteq V$, $U_j \subseteq U$, $V_j \subseteq \hat{V}_j$, $j = \overline{1, m}$. Предположим, что трассы разных связок не могут иметь общих ребер, т.е. $U_j \cap U_r = \emptyset$, $j \neq r$, $j, r = \overline{1, m}$. Тогда при заданном размещении R множество всех возможных трасс связки s_j обозначим через $\hat{T}_j$, а набор трасс, в котором каждой связке сопоставлена единственная трасса, - через $T = \{T_j\}$, $j = \overline{1, m}$. Для заданной трассы T_j определим длину L_j трассы как величину, равную сумме длин ее ребер, а для заданного набора трасс T - его длину $L(R, T) = \sum_{j=1}^m L_j$.

Задача размещения состоит в нахождении на плате P такого размещения $R^* = \{R_i^*\}$ элементов множества $E = \{e_i\}$, $i = \overline{1, n}$, схемы F , при котором $L(R^*, T^*) = \min_{R \in \hat{R}} \min_{T \in \hat{T}} L(R, T)$, где $\hat{T}$ – множество всех возможных наборов трасс.

Задача трассировки состоит в нахождении для заданного размещения R элементов из E на плате P такого набора трасс T^0 , при котором $L(R, T^0) = \min_{T \in \hat{T}} L(R, T)$.

В приведенной постановке в задаче размещения содержится в качестве подзадачи задача трассировки. Для упрощения задачи размещения обычно используют не значение L_j длины трассы, а ее оценку, величина которой зависит от взаимного расположения множества V_j узлов решетки – контактов связки v_j и от выбранной модели трассы (полный граф, кратчайшее связывающее дерево, звезда, габаритный прямоугольник и т.д.).

Задачи размещения и трассировки приведены здесь в упрощенной постановке, не учитывающей многие реальные требования, например, в задаче размещения отсутствует требование "равномерности" размещения элементов на плате, в задаче трассировки – требование минимизации числа пересечений трасс и т.д. Однако даже в такой постановке эти задачи относятся к классу экстремальных комбинаторных задач и могут быть сведены к общему случаю задачи дискретного программирования, если условия непересечения элементов заменить на неравенства, задающие расстояния между ними. При этом мы получим задачу, в которой области определения и значений целевой функции дискретны, а целевая функция и ограничения нелинейны.

Точные методы решения таких задач на практике не находят применения в силу следующих основных причин.

1. Точные методы решения связаны с перебором, т.е. имеют экспоненциальную сложность и не могут быть использованы для решения реальных задач.

2. Модели, получаемые при формализации процесса проектирования, в силу сложности как процесса, так и объекта проектирования, являются упрощенными и не достаточно точными, поэтому стремление к точному решению задач, основанных на этих моделях, не приводит к повышению качества объекта проектирования.

Размерность n реальных задач имеет порядок $10^3 - 10^4$, что приводит к необходимости использования для их решения алгоритмов, сложность которых не превышает $n^2 - n^3$. Поэтому на практике основ-

ное внимание уделяется разработке приближенных быстродействующих алгоритмов решения задач проектирования, которые могут и не находить точное решение, но

1) обладают небольшой вычислительной сложностью, что обеспечивает получение допустимого решения в приемлемое время;

2) позволяют при их разработке учитывать специфику решаемых задач с целью повышения качества решения.

§2. Алгоритмы решения задач размещения и трассировки

Для решения поставленных задач проектирования разработаны быстродействующие приближенные алгоритмы размещения одинаковых [1, 3, 4] и разногабаритных [2, 5, 6] элементов схемы и трассировки их соединений [1, 3, 7].

Алгоритмы размещения одинаковых элементов схемы в позициях регулярного поля позиций платы предназначены для проектирования устройств цифровой вычислительной аппаратуры. В этом случае задача размещения сводится к задаче оптимизации размещения точечных моделей элементов в узлах плоской целочисленной прямоугольной решетки, являющейся моделью поля позиций платы. При этом шаг решетки задается, исходя из технологических требований к расстоянию между элементами и обеспечению возможности проведения трасс, соединяющих контакты элементов схемы.

В алгоритме размещения [1, 3] используется метод последовательного размещения упорядоченных по связности элементов схемы с минимизацией приращения функции качества (оценки суммарной длины трасс) для получения начального размещения и метод локальной оптимизации с помощью попарных перестановок элементов для повышения качества размещения. С целью увеличения быстродействия алгоритма функция качества размещения вычисляется в соответствии с принятой моделью трассы - звездой, в которой все контакты данной связки соединены с центром системы этих контактов. Сложность вычисления приращения функции качества при размещении в данную позицию очередного i -го элемента пропорциональна k_i числу его контактов, а сложность алгоритма - nK , где n - число элементов, K - общее число контактов схемы.

В алгоритме размещения [4] с целью улучшения качества размещения использован метод построения произведения разбиений двудольного графа $G = (V_E \cup V_C, U)$, моделирующего схему устройства, при этом множество V_E соответствует множеству элементов, V_C - мно-

жеству связей схемы и $(v_{e_1}, v_{s_j}) \in U$, $v_{e_1} \in V_E$, $v_{s_j} \in V_G$, если хотя бы один из контактов элемента e_1 входит в связку s_j схемы.

Для графа G строятся два разбиения $\hat{V}_1$ и $\hat{V}_2$ множества его вершин такие, что ребра графа соединяют только соседние подмножества в каждом разбиении. Строится произведение разбиений $\hat{V}_1$ и $\hat{V}_2$, в котором каждое подмножество есть непустое пересечение двух подмножеств, взятых из разных разбиений.

Структура полученного произведения позволяет разбить поле позиций платы на прямоугольные области и элементам поставить в соответствие позиции из этих областей так, что элементам каждой связки будут соответствовать позиции, лежащие не более чем в четырех соседних областях. Выбор разбиений производится таким образом, чтобы минимизировать верхнюю оценку суммарной длины кратчайших прямоугольных деревьев Штейнера, соответствующих связкам схемы. Размещения, получаемые данным алгоритмом, близки по качеству к размещениям, получаемым вручную опытными специалистами.

При проектировании устройств, содержащих разногабаритные элементы, точечные модели элементов неприемлемы, поэтому в алгоритмах размещения разногабаритных элементов [2,5] моделями элементов и платы служат прямоугольники. При этом размеры моделей учитывают технологические ограничения и требование равномерности размещения, вытекающее из требования получения стопроцентной трассировки. Приближенные последовательные алгоритмы размещения разногабаритных элементов обладают рядом недостатков, в том числе низким быстродействием и отсутствием гарантии допустимого размещения всех элементов схемы. Поэтому разработаны быстродействующий итеративно-последовательный алгоритм размещения и алгоритм коррекции размещения разногабаритных элементов.

Итеративно-последовательный алгоритм размещения разногабаритных элементов, одна из реализации которого описана в [2], состоит из следующих этапов:

1. Упорядочение элементов схемы по связности.
2. Исходное разбиение платы на области. В установленном порядке элементы последовательно распределяются по множеству центров областей исходного разбиения. При этом находится такое положение очередного элемента, при котором минимально приращение оценки качества и удовлетворяется ограничение для плотности заполнения областей.

3. Итеративная процедура разбиения областей и покрытия платы системой зон. В каждой очередной итерации каждая из областей предыдущей итерации разбивается на ряд меньших областей, из которых формируется система частично-перекрывающихся зон, покрывающая плату. В соответствии с выбранной конечной последовательностью бесповторного обхода зон системы элементы очередной зоны распределяются (алгоритмом этапа 2) по областям этой зоны. При этом для элементов, размеры которых близки к размерам содержащих их областей, находится окончательное размещение.

4. С помощью последовательного алгоритма для каждого элемента схемы находится его размещение в некоторой окрестности области, содержащей этот элемент.

Разбиение платы на области, в которых происходит их размещение, приводит к увеличению быстродействия данного алгоритма по сравнению с последовательным. Увеличение количества областей разбиения платы ведет к уточнению положения на плате каждого элемента в очередной итерации, а перекрытие соседних зон обеспечивает возможность перехода элементов из одной области в другую - все это обеспечивает улучшение качества размещения. Вычислительная сложность итеративно-последовательного алгоритма не превосходит $(c_1 + c_2 f)g$, $c_1 + c_2 = 1$, $g = \text{cn}N$ - сложность последовательного алгоритма, n - число элементов схемы, $N = |N|$ - число узлов решетки платы, $f = f(d, r, N)$ - выпуклая функция, имеющая значения, меньшие единицы в области определения, задаваемой реальными значениями N , d - исходного числа областей разбиения платы и r - числа итераций алгоритма. Управляющие параметры $(d, r, \dots)$ алгоритма позволяют для каждого конкретного устройства задавать режим работы алгоритма, обеспечивающий его наибольшее быстродействие [6].

Алгоритм коррекции размещения [5] предназначен для устранения недопустимого размещения элементов (наложения элементов друг на друга или на запрещенные зоны платы) без существенного изменения качества размещения. Удовлетворение этих требований и увеличение быстродействия осуществляется с помощью разбиения платы на ячейки и применения в выбранном порядке обхода ячеек процедуры уплотнения для каждой ячейки отдельно. Процедура уплотнения состоит в установке очередного элемента в наилучшую (в смысле плотности размещения) позицию граничного контура ранее размещенных элементов. Вычислительная сложность алгоритма пропорциональна n^2/d , n - число элементов схемы, d - число ячеек разбиения платы.

Построение трассы, соединяющей все контакты данной связи, осуществляется с помощью алгоритма кратчайшей трассировки [1,3]. При заданном размещении R элементов схемы на плате для каждой связи a_j известно множество $V_j \subset V$ вершин графа $H = (V, U)$, которые должны принадлежать трассе T_j , $j = \overline{1, m}$. Для множества V_j строится полный взвешенный граф $G_j(V_j, W_j)$, каждому ребру которого приписан вес, равный расстоянию в H между инцидентными ему вершинами. Для графа G_j с помощью алгоритма Прима находится кратчайшее связывающее дерево, вершины которого упорядочиваются. Для всех ребер, инцидентных очередной вершине, находятся соответствующие им кратчайшие пути в графе H . Среди найденных путей выбирается сочетание путей, содержащее единственный путь для каждого ребра и имеющее наибольшее совмещение по длине с уже построенной частью кратчайшей трассы T_j .

Для кратчайших трасс установлена верхняя оценка величины совмещения длин путей. Время работы алгоритма при построении одной трассы пропорционально квадрату числа ее контактов.

Решение задачи трассировки может быть не единственным, т.е. для каждой связи могут существовать несколько трасс, удовлетворяющих условиям задачи. Тогда возникает вопрос о выборе для каждой связи такой трассы, совокупность которых по всем связкам имеет экстремальное значение некоторого критерия, например количества взаимных пересечений трасс. Задача о пересечении трасс сведена [7] к задаче альтернативного выбора.

Дана симметрическая матрица $A = \|a_{ik}\|$ порядка N , a_{ik} — действительные числа. Множество $\{a_i\}$ строк (соответственно столбцов) разбито на m подмножеств $\hat{a}_j$, $j = \overline{1, m}$, по n_j строк в каждом,

$\sum_{j=1}^m n_j = N$. Каждая строка $a_i \in \hat{a}_j$ соответствует некоторой трассе T_i из множества $\hat{T}_j$ трасс связи a_j , a_{ik} — число пересечений трасс T_i и T_k . Каждой строке (столбцу) a_i сопоставим переменную $x_i \in \{0, 1\}$ и введем функцию качества $F(X, A) = \sum_{i=1}^N \sum_{k=1}^N x_i x_k a_{ik}$. Задача альтернативного выбора состоит в нахождении оптимального набора X^* такого, что $F(X^*, A) = \min_X F(X, A)$ при условии $\sum_{a_i \in \hat{a}_j} x_i = 1$,

$j = \overline{1, m}$, т.е. из каждого подмножества $\hat{a}_j$ нужно выбрать по одной строке так, чтобы образовавшаяся в результате этого матрица порядка m имела минимальную сумму элементов.

Для построения алгоритма приближенного решения этой задачи использован метод локальной оптимизации, состоящий из нахождения исходного допустимого набора строк и поиска в его окрестности набора, функция качества которого не превышает исходную. Вычислительная сложность алгоритма пропорциональна N^2 , т.е. квадрату числа всех возможных трасс.

Разработанные алгоритмы размещения и трассировки явились основой для создания САПр печатных плат.

§3. Система автоматизированного проектирования печатных плат

САПр печатных плат, разработанная в Институте математики СО АН СССР [8], предназначена для автоматизации этапа конструкторского проектирования радиоэлектронной аппаратуры на двухслойных печатных платах с односторонним расположением разногабаритных и одинаковых элементов.

Основными функциональными компонентами системы являются информационная компонента, банк данных, комплекс прикладных модулей.

Основу информационной компоненты составляют средства описания входной и выходной информации системы и ее модулей, управления модулями системы и документации. Исходное описание проектируемого устройства состоит из таблицы спецификаций (наименования элементов и их позиционные обозначения), описания печатной платы (размеры печатной платы, шаги размещения и трассировки, размеры и положение на плате запрещенных зон, описание полей позиций одинаковых элементов), принципиальной схемы (описание связей и звезд, звезда — множество связей, в которые входят контакты данного элемента), указаний конструктора (фиксированные на плате элементы и трассы, пары элементов или трасс с заданными ограничениями на расстояния между ними, особенности контактов элементов и трасс). Выходная информация системы состоит из описания размещения элементов, описания трасс, управляющих параметров модулей и характеристик полученных результатов (процент трассировки, суммарная длина соединений, число межслойных переходов и т.д.).

Банк данных предназначен для информационного обеспечения системных и прикладных модулей, осуществляющих автоматизированный процесс проектирования устройства. Банк данных состоит из базы данных, разделенной на постоянную (ПОБД) и переменную (ПЕБД) базы

и поддерживающей ее системы управления. Постоянная база данных содержит библиотеку используемых элементов, в которой хранятся описания их геометрии и особенностей; характеристики прикладных наборов ПЕБД и описание технологических параметров. Переменная база данных содержит входные описания проектируемых устройств и варианты выходной информации прикладных модулей. Функции поддерживающей системы управления банка данных состоят в хранении, наполнении и реконструкции ПОВД и ПЕБД и обеспечении обмена между основной памятью и прикладными наборами переменной базы данных. Исходное описание устройства обрабатывается системными модулями грамматического контроля, семантического контроля и сепарации, который формирует наборы, входные для комплекса прикладных модулей, и записывает их в одноименные файлы ПЕБД.

Комплекс прикладных модулей непосредственно осуществляет процесс автоматизированного проектирования печатных плат и состоит из модулей

- размещения разногабаритных элементов [2], управляющие параметры которого позволяют изменять геометрию моделей элементов и способ разбиения платы на перекрывающиеся зоны;
- коррекции размещения элементов [5], управляющие параметры которого воздействуют на направление уплотнения элементов и на количество ячеек разбиения платы;
- размещения одинаковых элементов [4], применяемого при проектировании цифровой аппаратуры;
- оперативного редактирования размещений элементов, полученных другими модулями;
- маршрутной и волновой трассировки соединений, управляющие параметры которых воздействуют на их быстродействие, число точек поворота трасс, количество межслойных переходов и т.д.

Воздействие на результаты работы прикладных модулей посредством изменения управляющих параметров обеспечивает вариативность выходной информации. Наличие нескольких вариантов выходной информации позволяет выбрать лучший из них в качестве входного для последующего этапа проектирования. Тем самым управляющие параметры модулей совместно с указаниями конструктора, возможностью изменения технологии в ПОВД и режима функционирования системы позволяют адаптировать процесс проектирования к конкретным особенностям проектируемого устройства.

Структура и функционирование разработанной САПр печатных плат ориентированы на проектирование устройств РЭА с разногабаритными элементами и определяются спецификой возникающих при этом задач. Формализация такого процесса проектирования и его объекта (устройства) приводит к сложным математическим моделям (взвешенные гиперграфы, многосвязные пространства и т.д.) и оптимизационным задачам в общей постановке. Необходимость завершения процесса проектирования в приемлемое время требует разработки быстродействующих приближенных алгоритмов их решения. Каждый из таких алгоритмов не может гарантировать получение хорошего качества результатов, поэтому возникает необходимость включения в САПр различных алгоритмов, решающих задачи с разными критериями оптимизации и при разных условиях и имеющих управляющие параметры, воздействующие на качество результата. Кроме того, САПр должна предоставлять оператору-проектировщику возможность управления процессом проектирования посредством изменения схемы функционирования системы: изменения последовательности работы модулей, изменения метода проектирования (например, совместное решение задачи размещения и трассировки или их последовательное независимое решение, последовательное построение трасс или выбор лучшего набора трасс из всех возможных) и т.д. и возможность оперативного редактирования результатов работы алгоритмов с помощью специализированной операционной системы САПр и ее эффективных периферийных устройств.

Изложенные требования к системам частично реализованы в САПр печатных плат [8], которая обеспечивает проектирование нескольких устройств одновременно, контроль входной и выходной информации, учет технологических, конструктивных и схемотехнических требований к устройству, воздействие на процесс проектирования посредством управляющих параметров модулей, оперативную работу с разными вариантами выходной информации модулей системы.

Заключение

Основной целью работ по автоматизации проектирования в радиоэлектронике и вычислительной технике следует считать создание САПр как инструмента оператора-проектировщика, позволяющего автоматизировать в едином комплексе все этапы проектирования устройств РЭА. При этом необходимо учитывать объектную область и временной интервал применения создаваемой САПр в связи с быстрым

изменением как технологии изготовления РЭА, так и технической базы реализации САПр.

Можно выделить следующие этапы создания САПр РЭА:

- исследование процессов и объектов проектирования для разработки адекватных им математических моделей и постановки задач путем формализации технологических, конструкторских, схемотехнических и т.д. условий и целей проектирования устройства РЭА;

- разработку эффективных методов и алгоритмов нахождения приближенных решений поставленных задач, т.е. создание для каждого класса задач набора быстродействующих алгоритмов, адаптируемых к условиям конкретной задачи с помощью управляющих параметров или посредством изменения схемы функционирования; выбор лучших из алгоритмов может быть произведен либо на специальном полигоне сравнения алгоритмов, либо в самой САПр по результатам их эксплуатации в системе;

- реализацию САПр на соответствующей технической базе как специализированной операционной системы, при этом некоторые алгоритмы могут быть реализованы аппаратно.

Основные требования и принципы реализации САПр (см. [9]) суть:

- инвариантность системы к изменениям технологии производства, способов и средств проектирования, т.е. иерархичность структуры, перестраиваемость и наращиваемость системы;

- интерактивность системы, т.е. оперативное управление процессом проектирования с целью обеспечения изменения режимов (ручное, автоматизированное, автоматическое) проектирования и схемы функционирования системы;

- адаптивность системы, т.е. возможность настройки структуры системы, ее схемы функционирования и прикладных модулей на особенности конкретного проектируемого устройства, а также способность оценки и отбора лучших алгоритмов по результатам их работы.

Трудоемкость создания такой системы оценивается примерно в 600 человеко-лет.

Л и т е р а т у р а

1. МАКАРОВ Л.И. Размещение и трассировка в плоской прямоугольной решетке. - В кн.: Автоматизация проектирования в микроэлектронике. Теория. Методы. Алгоритмы (Вычислительные системы, вып. 64), Новосибирск, 1975, с. 65-72.

2. КАШИН В.И., МАКАРОВ Л.И. Итеративно-последовательный алгоритм размещения моделей разногабаритных элементов. -В кн.: Алгоритмические основы обработки структурной информации (Вычислительные системы, вып. 85). Новосибирск, 1981, с. 64-77.

3. БСЧКО Г.Д., КАШИН В.И., МАКАРОВ Л.И. Алгоритмы последовательного размещения и кратчайшей трассировки. -В кн.: Автоматизация проектирования в микроэлектронике. Теория. Методы. Алгоритмы. (Вычислительные системы, вып. 64). Новосибирск, 1975, с. 73-81.

4. БЕССОНОВ Ю.Е., СКОРОВОГАТОВ В.А. Использование структурных особенностей схем при оптимизации размещений. -В кн.: Алгоритмические основы обработки структурной информации (Вычислительные системы; вып. 85). Новосибирск, 1981, с. 21-34.

5. БЕЛЯЕВ Е.И., МАКАРОВ Л.И. Алгоритм коррекции размещения разногабаритных элементов. -Там же, с. 78-80.

6. БЕЛЯЕВ Е.И., МАКАРОВ Л.И. Быстродействие алгоритмов размещения и коррекции размещения разногабаритных элементов. -В кн.: Вычислительная техника и дискретная математика. Тезисы докл. науч.-техн. конф. Новосибирск, 1983.

7. МАКАРОВ Л.И. Дихотомическая постановка задачи минимизации числа пересечений ребер графа. -В кн.: Вопросы обработки информации при проектировании систем (Вычислительные системы, вып. 69), Новосибирск, 1977, с. 40-46.

8. КУЛИШ Г.Н., МАКАРОВ Л.И. Система автоматизированного проектирования печатных плат. -Тезисы докл. III Всесоюз. конф. АПС-83, Иваново, 1983.

9. АНИШЕВ П.А., ПИСКУНОВ С.В., СЕРГЕЕВ С.Н. О проекте системы автоматизированного проектирования в микроэлектронике. УСИМ, 1974, №2, с. 106-109.

Поступила в ред.-изд.отд.

19 декабря 1983 года