

Федеральное агентство по образованию
Государственное образовательное учреждение высшего профессионального образования
«Владимирский государственный университет»
Кафедра Конструирования и технологии радиоэлектронных средств

В. Р. АСЛАНЯНЦ

УЧЕБНАЯ САПР ЭЛЕКТРОННЫХ СРЕДСТВ

ПРАКТИКУМ

Владимир 2007

УДК 621.396.6.001 (076.5)

A90

Рецензенты:

Доктор технических наук, профессор, заведующий кафедрой «Конструирование, технология и производство РЭС» Московского авиационного института (Государственного университета)

М.Н. Ушкар

Кандидат технических наук, преподаватель Владимирского государственного университета

Г.Ф. Долгов

Печатается по решению редакционно-издательского совета Владимирского государственного университета

Асланянц В.Р.

A90 Учебная САПР электронных средств:

Практикум / Владим. гос. ун-т; Владимир, 2007. 60 с.

ISBN

Основное назначение практикума показать, как применяются объекты и методы дискретной математики: множества и отношения (нечеткие множества в том числе), исчисление высказываний и алгебра логики, алгоритмы, теория графов и математическое программирование (методы решения оптимизационных задач) для формирования математических моделей электронных средств и алгоритмов синтеза топологии коммутационных плат. Описана структура учебной САПР CROCUS-3 и методы работы с ней.

Содержит перечень заданий к лабораторным работам и УИРС, список из 40 контрольных вопросов.

Предназначен для студентов специальностей 210201 и 210202 дневного и заочного отделения, выполняющих лабораторные работы по дисциплине «Математические основы информатики и САПР» и УИРС на 3-м курсе,

Ил. 18. Библиогр.: 12 назв.

УДК 621.396.6.001 (076.5)

ISBN 5-230-04841-7

© Владимирский государственный университет, 2007

ВВЕДЕНИЕ

Системы автоматизированного проектирования электронных средств (САПР ЭС) - основной инструмент проектировщика электронной аппаратуры. Математическое обеспечение САПР можно охарактеризовать двумя составляющими: математические модели (ММ) и алгоритмы решения проектных задач. ММ и алгоритмы синтеза топологии коммутационных плат основаны на следующих разделах дискретной математики: множества и отношения (нечеткие множества в том числе), исчисление высказываний и алгебра логики, алгоритмы, теория графов и математическое программирование (методы решения оптимизационных задач). Основная цель практикума - научить студентов специальностей 200800 и 220500 применять эти разделы математики в проектных задачах.

Для этого используется учебная САПР CROCUS-3, особенностью которой является возможность увидеть и проанализировать подробно работу проектирующих алгоритмов синтеза конструкции ячейки ЭС и соответствующих ММ схем и ММ монтажных пространств конструктивных узлов.

Эта цель достигается 2-мя средствами:

1. Студент имеет возможность посмотреть на экране видеомонитора процесс работы проектирующей процедуры в пошаговом режиме.
2. Студенты изучают и объясняют при защите лабораторных работ мелкомасштабные схемы алгоритма.

Использование практикума удобно, например, в цикле из 4-х лабораторных работ, в каждой из которых применяется одна из проектирующих программ, указанных в разделе 1. В результате студенты выполняют конструкторское проектирование небольшой ячейки ЭС. Фронтальный метод выполнения лабораторных работ создает у студента целостное представление о процессе проектирования.

1. ДИСКРЕТНАЯ МАТЕМАТИКА

Наиболее важными разделами дискретной математики, которые применяются для формирования математических моделей (ММ) объектов проектирования в области электроники и для получения оптимальных проектных решений, являются следующие:

- теория множеств и отношений;
- математическая логика (логика Буля в частности);
- алгоритмы;
- теория графов и гиперграфов;
- математическое моделирование;
- математическое программирование (теория оптимизации).

Из перечисленных разделов математическое моделирование относится не только к дискретной математике, поскольку помимо дискретных ММ широко используются непрерывные ММ.

Рассмотрим кратко содержание перечисленных разделов.

Теория множеств и отношений

Для задания множеств применяют 2 способа [7]:

- перечисление элементов множества;
- задание способа конструирования множества.

Упорядоченные множества называют кортежами (векторами).

В алгебре множеств применяются операции над множествами:

- объединение множеств;
- пересечение;
- разность (эта операция двухместная);
- дополнение (эта операция тоже двухместная);
- разбиение;
- декартово (прямое) произведение.

Отношения на множествах лежат в основе реляционного исчисления и применяются в наиболее распространенных в настоящее время реляционных базах данных.

Частные случаи отношений:

- отношения эквивалентности;
- отношения порядка и др.

Нечеткие множества – основа для нечеткой логики и нечеткой математики.

Исчисление высказываний (логика Буля)

Формализация силлогистики (логики Аристотеля), выполненная Дж. Булем, привела к появлению исчисления высказываний (булевой логики) [11]. В логике Буля сложные высказывания представляются в виде ППФ – правильно построенных формул логики.

Дж. Булем были введены также простейшие функции алгебры логики - конъюнкция, дизъюнкция, отрицание, импликация, эквивалентность и др. Простейшие и более сложные булевы функции описываются таблицами истинности. В исчислении высказываний (одной из дедуктивных систем) к формулам (сначала к аксиомам) применяют правила вывода (правило отделения, правило резолюции и др.) и получают цепочку выводов.

В алгебре логики действуют ряд законов, таких как переместительный, сочетательный, законы де Моргана, законы поглощения и др.

Булевы функции могут быть представлены в дизъюнктивной (ДНФ), конъюнктивной (КНФ) нормальной форме, в совершенной ДНФ (СНДФ) или совершенной КНФ (СКНФ). Эти формы используются при синтезе логических схем ЭС, в том числе при программировании ПЛИС.

Теория алгоритмов

Интуитивное понятие алгоритма подкрепляется эмпирическими свойствами алгоритмов:

- дискретность;
- детерминированность;
- массовость;
- результативность.

Существуют так называемые алгоритмически неразрешимые проблемы (задачи), например проблема самоприменимости (парадокс брадобрея) и др.

К способам представления алгоритмов относят:

- словесное описание алгоритма;
- схемное описание;
- псевдокоды;
- языки программирования.

Наиболее важными критериями оценки и сравнения алгоритмов являются следующие:

- быстродействие алгоритма;
- точность алгоритма;

- временная сложность.

Теория графов

Графы и гиперграфы широко используются в качестве математических моделей (ММ) схем и конструкций в задачах синтеза и анализа ЭС [2,6,7,8].

Понятие граф опирается на теорию множеств и отношений. Граф представляет собой бинарное отношение на множестве объектов (вершин графа). Граф состоит из вершин и ребер (дуг для ориентированного графа).

К разновидности графа относят:

- мультиграф;
- двудольный граф;
- полный граф;
- дерево;
- гиперграф.

Гиперграф представляет собой n -арное отношение на множестве вершин, т.е. гиперребра могут быть инцидентны трем, четырем и более вершинам.

К числовым способам представления графа относят:

- матричное представление;
- списковое представление.

Существует два особенных цикла в графе: эйлеров и гамильтонов.

Простой признак существования эйлеров цикла (это первый результат в теории графов) сформулирован в виде теоремы Эйлера.

Для гамильтонова цикла простого признака существования нет.

Нахождение гамильтонова цикла минимальной длины во взвешенном по ребрам графе называется задачей коммивояжера.

Изучение графов облегчается путем введения чисел графа [12]:

- цикломатического;
- хроматического;
- числа внутренней устойчивости;
- числа внешней устойчивости;
- числа планарности.

Для определения максимального ВУМ (внутренне устойчивого множества вершин) и раскраски вершин графа применяют метод Магу и эвристические алгоритмы.

При автоматизированном проектировании однослойных конструк-

ций ЭС применяют теорию планарных графов. Для планаризации схем ЭС применяют циклический метод Аусландера и Портера, алгоритмы Демукрона и др.

Математическое программирование

Процесс разработки конструкции ЭС включает в себя целый ряд задач поиска оптимальных конструктивно-технологических решений на множестве вариантов, число которых, как правило, велико [7]. В автоматизированном конструировании необходимо от содержательной постановки задачи перейти к математической модели [2]. Такой переход называется формализацией, в которой выделяют две стороны: построение модели конструкции ЭС и математическая формулировка оптимизационной задачи.

Для поиска оптимального решения необходимо сформулировать цель поиска, выраженную в виде критерия оптимизации F (целевой функции), экстремальное значение которого отыскивается. Предполагается, что любые два варианта из области поиска (области определения целевой функции) можно сравнить, т.е. поставить им в соответствие число – значение критерия F , следовательно, на множестве вариантов задано заранее неизвестное отношение порядка.

Критерий F зависит от управляемых параметров (вектор X), которые можно менять в процессе поиска оптимума, и неуправляемых (вектор Z). Тогда оптимизационная задача в общем виде формулируется следующим образом.

Требуется найти экстремальное значение критерия F :

$$\text{extr } F(X, Z)$$

при m ограничениях

$$q_k(X, Z) \leq a_k, \quad k = 1, 2, \dots, m ;$$
$$X \in D,$$

где D – область допустимых решений.

Методы решения оптимизационных задач (3.1), (3.2) рассматриваются в области математики, именуемой математическим программированием [7]. Эти методы существенно зависят от характера целевой функции и ограничений, что положено в основу приведенной ниже классификации разделов математического программирования.

Линейное программирование (ЛП) изучает оптимизационные задачи, в которых функции F и q линейны относительно параметров X и Z .

Ряд возникших в практической деятельности инженера оптимизаци-

онных задач приводят к линейной целевой функции и линейным ограничениям. В качестве примера рассмотрим производственную задачу о распределении ресурсов.

Пусть имеются некоторые ресурсы (сырье, рабочая сила и т.п.) в ограниченных количествах b_1, b_2, \dots, b_m . Известно, что для производства единицы j -й продукции необходим i -й ресурс в количестве a_{ij} . Требуется определить оптимальный план выпуска: какое количество x_j каждого вида продукции следует выпускать, чтобы получить максимальную прибыль, если известна чистая прибыль c_{ij} от реализации единицы j -й продукции. Математически эта задача сводится к поиску максимума линейной формы

$$\text{Max } F = \sum_{j=1}^n c_j x_j$$

при линейных ограничениях-неравенствах

$$\sum_{j=1}^n a_{ij} x_j \leq b_i, \quad i = 1, 2, \dots, m;$$
$$x_j \geq 0.$$

Если известны условия спроса, то на x_j накладываются дополнительные ограничения

$$x_j \leq p_i,$$

где p_i – максимальное в условиях спроса количество j -й продукции.

Для решения задачи ЛП применяют симплекс-метод [7].

Нелинейное программирование (НП) - функции F и g нелинейны относительно X и Z , область допустимых решений D может быть невыпуклой и содержать бесконечное множество решений, а целевая функция чаще всего многоэкстремальна.

Общего метода решения задач НП не существует. Выбор метода осуществляется на основе априорной информации о характере целевой функции и ограничений.

Например, для одномерных задач НП ($n = 1$) известны классические методы: метод, основанный на числах Фибоначчи, метод «золотого сечения» и др.

Ограниченное применение в многомерных задачах находят дифференциальное исчисление, метод множителей Лагранжа.

К поисковым методам, основанным на итерационном принципе (принципе последовательных приближений), относятся:

- метод локального поиска на сетке;

- градиентный метод;
- метод наискорейшего спуска;
- метод покоординатного спуска;
- метод штрафных функций;
- метод случайного поиска.

Указанные методы обеспечивают нахождение одного из локальных экстремумов. Для отыскания глобального экстремума применяют:

- метод «кенгуру и мыши»;
- метод Монте-Карло.

Динамическое программирование – процесс решения естественным образом (или искусственно) распадается на этапы, и значение функции F определяется как сумма (произведение) соответствующих значений F на отдельных этапах, т.е. целевая функция F аддитивна (мультипликативна).

Примером может служить задача отыскания кратчайшего пути на взвешенном по ребрам графе, для решения которой может быть применен алгоритм Беллмана-Калаба. Частный случай этой задачи – трассировка проводников на печатной плате, для которой применяется волновой алгоритм [6].

Стохастическое программирование – параметры Z – случайные величины, и вместо функции F отыскивается экстремум ее математического ожидания.

Дискретное программирование – на X , Z наложено условие дискретности (например, целочисленности), т.е. область определения D представляет собой конечное множество. В зависимости от вида функции F и q задачи дискретного программирования могут быть линейными (целочисленное линейное программирование - ЦЛП) и нелинейными (дискретное нелинейное программирование - ДНП).

Для решения задач ЦЛП применяют:

- метод отсечений, основанный на многократном применении симплекс-метода;
- метод ветвей и границ;
- итерационные эвристические алгоритмы, которые можно назвать дискретными аналогами градиентных методов и др.

Для решения задач ДНП применяют:

- метод ветвей и границ;

- метод случайного поиска;
- эвристические алгоритмы (последовательные итерационные, дихотомические, генетические и др.). В этом случае можно говорить об эвристическом программировании.

Эвристическое программирование. Если решение задачи точными методами трудоемко, то довольствуются приближенным достаточно хорошим для практики решением, которое отыскивается с использованием специальных приемов – эвристик, позволяющих сокращать число рассматриваемых вариантов. Эвристики разрабатываются на основе изучения и использования специфики конкретных задач.

Потребности практики автоматизированного конструирования ЭС обусловили развитие и широкое использование методов оптимизации, в которых ценой отказа от получения точного решения экстремальной задачи существенно уменьшается высокая трудоемкость, присущая описанным выше методам решения задач дискретного программирования. Для этого направления характерны два принципиально разных подхода: методы случайного поиска и эвристические методы.

Эвристический подход основан на использовании совокупности эвристик – приемов, обеспечивающих в процессе поиска приближение к оптимальному решению, хотя и не гарантирующих его достижение.

Примером эвристических алгоритмов могут служить последовательные алгоритмы размещения конструктивных элементов и разбиения схем ЭС [1, 3, 5, 6], а также комбинаторные аналоги градиентных методов, рассмотренные в подразделе 4.1. Действительно, как известно градиентные методы основаны на свойстве непрерывности целевой функции, и применение идей градиентных методов к дискретным задачам может дать хорошие результаты только для определенного класса задач, что определяется экспериментально.

Один из путей построения эвристических алгоритмов – анализ процесса решения задачи некоторым точным методом и введение эвристик в этот процесс, обеспечивающих существенное сокращение объема вычислительной работы. Так, например, если в методе ветвей и границ после каждого ветвления сохранять лишь одно (или небольшое число) наиболее перспективных подмножеств решений, то процесс быстро сходится. Однако гарантий получения оптимального решения уже нет: оно может оказаться в отброшенном подмножестве решений.

Применение метода Монте-Карло основано на гипотезе, что во множестве всех допустимых решений существует сравнительно большое количество вариантов решений, близких к оптимальному, т.е. вероятность случайного выбора приемлемого решения достаточно высокая. К сожалению, экспериментальные исследования ряда задач технического проектирования ЭС эту гипотезу не подтверждают. В таких задачах, как правило, отношение числа «хороших вариантов» к их общему числу ничтожно мало, что ограничивает использование метода Монте-Карло. Преимущество метода Монте-Карло – возможность оценить степень близости полученного решения к оптимальному. Ценность метода в том, что на его основе возможно сравнение различных алгоритмов решения дискретных задач.

Вариационные задачи. Управляемыми параметрами X могут быть не только переменные, но и функции. Критерий F в этом случае называется функционалом, пространство поиска – функциональным, а задача – вариационной.

Многокритериальные задачи. Практические задачи конструирования ЭС могут требовать оптимизации не по одному критерию F , а по нескольким, в общем случае противоречивым. Методы решения многокритериальных задач интенсивно развиваются в настоящее время.

Для решения многокритериальных задач применяют:

- свертку критериев качества в один критерий с весовыми коэффициентами для каждого исходного критерия;
- ранжирование критериев качества и др.

Конструктору ЭС приходится иногда принимать решение в условиях неполной информации. Наличие неопределенности усложняет оптимизационную задачу, но и в этом случае математические приемы помогают человеку принять более обоснованное решение.

В том случае, если проектную задачу не удастся формализовать, т.е. построить ММ объекта проектирования и разработать или подобрать подходящий метод (алгоритм) решения задачи, то такую задачу называют трудно формализуемой. Для решения таких задач применяют технологии экспертных систем, искусственные нейронные сети и другие методы искусственного интеллекта.

2. АРХИТЕКТУРА УЧЕБНОЙ САПР CROCUS-3

Программное обеспечение учебной САПР CROCUS-3 (минимальный вариант) составляют четыре проектирующих программы DECOM-3, PLACE-3, TLO-3 и TRACE-3 в виде загрузочных модулей и одна служебная программа CODER-3.

Программа DECOM-3 предназначена для разбиения электрических схем ЭС.

Программа PLACE-3 выполняет размещение элементов на коммутационной плате и распределение цепей по выводам конструктивного узла.

Программа TLO-3 выполняет три подготовительные задачи трассировки электрических соединений: построение кратчайших соединений, расслоение монтажа и упорядочение соединений .

Программа TRACE-3 выполняет прокладку трасс в монтажном пространстве конструктивного узла.

Программа CODER-3 выполняет перекодировку (перенумерование) конструктивных элементов и электрических цепей одной (указанной преподавателем в качестве задания) подсхемы, полученной в результате разбиения, так чтобы и элементы и цепи получили номера 1, 2, 3,

Информационное обеспечение учебной САПР состоит из двух разделов: папка LIBRARY и папка PROJECTS.

Папка LIBRARY содержит описание посадочных мест под конструктивные элементы (ИМС): файлы с именами типа EL4-14. В файле EL4-14 описано посадочное место под корпус четвертого типа с 14-ю выводами.

Папка PROJECTS содержит описание выполненных проектов (базы данных проектов), в том числе и исходные описания контрольных задач, например, контрольных проектов TEST27DEC и TEST3x4, описанных в практикуме.

Среда функционирования САПР – персональный компьютер и операционная система Windows 98 и старше.

Инструментальная среда разработки САПР - C++ Builder.

Ограничения САПР CROCUS-3

Вследствие учебного назначения САПР CROCUS-3 имеет следующие ограничения.

1. Все конструктивные элементы (в том числе и соединители) размещаются в посадочные места одного типа.

2. Число элементов равно числу посадочных мест. Если фактическое число элементов схемы меньше числа посадочных мест, то вводятся “пустые” элементы, т. е. такие, к которым не подсоединены электрические цепи.

3. Нумерация элементов следующая: сначала нумеруются ИМС, затем “пустые” элементы (если они есть), затем соединители.

4. Число элементов (вместе с соединителями) не более 25.

5. Ориентация платы и ориентация фиксированная - как показано на рис. 7.

7. Межслойные переходы располагаются только по выводам элементов и занимают один дискрет ДРП (дискретного рабочего поля) [2, 6].

8. Запрещенные для трассировки области имеют прямоугольную форму и повторяются во всех коммутационных слоях.

9. Подготовку исходного описания проекта следует производить в среде текстового редактора DOS (или редактора встроенного в FAR-менеджер и т. п.).

После запуска соответствующая проектирующая программа считывает нужные ей файлы с диска и начинает выполнять проектирование.

Для описания проектных задач применяется стандартизованная или общепринятая терминология [10] со следующими уточнениями.

Термины конструктивный узел и конструктивный элемент означают конструктивные модули смежных уровней иерархии конструкции ЭС. Например, если конструктивный узел – это ячейка ЭС (типовой элемент замены (ТЭЗ), конструктивный модуль первого уровня), то конструктивные элементы – интегральные микросхемы (ИМС), дискретные электрорадиоэлементы (ЭРЭ), то есть конструктивные модули нулевого уровня иерархии.

Другой пример: конструктивным узлом является блок ЭС (модуль второго уровня), а элементами – ячейки ЭС.

При изложении материала термин «электрическая цепь» или просто «цепь» употребляется в общепринятом смысле – это связанная эквипотенци-

альная часть электрической схемы, термин «электрическое соединение» или просто «соединение» означает связанную электрическим проводником пару контактов (например, пару контактных площадок ИС), термин «трасса соединения» или просто «трасса» означает полностью и точно геометрически определённый проводник в монтажном пространстве коммутационной платы.

Во избежание путаницы предлагается термин «связь» употреблять лишь для обозначения паразитных связей в конструкциях ЭС, но не для электрических проводников монтажа.

И, наконец, предлагается не употреблять в проектных задачах, описанных в практикуме, термин «компоновка» в связи с тем, что он в разных литературных источниках трактуется по-разному: под ним подразумевают разбиение схем, или покрытие схем набором функционально-типизированных узлов (ФТУ), или размещение конструктивных элементов в монтажном пространстве конструкторского узла.

3. ПОКРЫТИЕ ФУНКЦИОНАЛЬНОЙ СХЕМЫ ЭС НАБОРОМ ФТУ И РАЗБИЕНИЕ СХЕМЫ ЭС

Лабораторная работа № 1

3.1. Описание проектной задачи покрытия электрической функциональной схемы ЭС (СхЭФ) набором функционально-типизированных узлов (ФТУ)

Покрытие СхЭФ набором ФТУ рассмотрим для цифровой аппаратуры [1]. В этом случае функциональной схемой является логическая схема ЭС, ФТУ- это набор ИМС (например серия ИМС).

Логическая схема состоит из функциональных (логических) элементов. В простейшем случае это логические вентили и триггеры.

ИМС из заданного набора состоят из таких же связанных между собой элементов, т. е. известны электрические схемы каждой ИМС, известно к каким выводам корпуса ИМС подходят соответствующие цепи.

Фактически покрытие схемы – это разбиение логической схемы на подсхемы, которые заранее заданы.

Таким образом, разработчик должен предварительно выбрать подхо-

дящую элементную базу (набор ИМС), затем решать задачу покрытия логической схемы набором ИМС.

В результате покрытия логическая схема преобразуется в схему электрическую принципиальную, но так, чтобы законы преобразования сигналов, заданные логической схемой, не нарушались.

Входные данные

1. Схема электрическая функциональная ЭС (логическая схема).
2. Набор ФТУ (ИМС обычно).

Выходные данные

Схема электрическая принципиальная ЭС (СхЭПр).

Обратите внимание, что в этой задаче ИМС играют роль конструктивных узлов, а не элементов.

На рис. 2 представлена СхЭПр, полученная в результате покрытия логической схемы интегральными микросхемами малой степени интеграции серии 137.

Критерии качества

1. Минимальная стоимость покрытия. Если все ИМС стоят одинаково, то тогда минимизируется число израсходованных ИМС.
2. Минимальное число типов узлов покрытия.
3. Минимальное число межузловых соединений. Этот критерий используется совместно с первым критерием, так как первый критерий дает не одно, а несколько одинаковых оптимальных решений.

Математические модели объектов проектирования

Математическая модель логической схемы - взвешенный по вершинам гиперграф. "Весы" вершин означают различные типы логических элементов схемы.

Математические модели ИМС – аналогичные гиперграфы меньшей размерности.

Формализованная формулировка

Покрытие схем – это задача идентификации графов (распознавание структур). Действительно, при решении ее необходимо отыскать в гиперграфе схемы подграфы, соответствующие гиперграфам ИМС. Таким образом, покрытие схем сводится к задаче покрытия взвешенного по вершинам гиперграфа гиперграфами из заданного набора.

Сведения к задаче математического программирования

В общем случае покрытие логическая схема – комбинаторная задача

экстремального типа: нелинейная дискретная задача математического программирования.

В частном случае, когда ИМС содержит несвязанные между собой логические элементы, то задача становится линейной и упрощается: это задача ЦЛП – целочисленного линейного программирования [7].

Алгоритмы покрытия

Ввиду трудности формализации и сложности решения задача покрытия схем ЭС решается обычно вручную. Известные алгоритмы являются эвристическими [1]. Для частного случая, когда удастся свести задачу к ЦЛП, можно применить метод отсечений [7] или эвристические алгоритмы.

3.2. Описание проектной задачи разбиения схем ЭС

Подсхемы, полученные в результате разбиения СхЭПр, реализуются в дальнейшем в виде отдельных конструктивных узлов (например, ячеек ЭС). Поэтому говорят также о разбиении схемы на узлы.

Входные данные

1. Схема электрическая принципиальная ЭС (пример СхЭПр представлен на рис. 2).
2. Параметры конструкций будущих конструктивных узлов. Обычно это вместимость узла и число внешних выводов узла (число контактов соединителей конструктивного узла).

Выходные данные

Полученная совокупность подсхем (электрических принципиальных).

Критерии качества

1. Минимум числа межузловых соединений.
2. Минимум числа узлов (подсхем).
3. Минимум числа типов узлов (задача типизации).
4. Функциональная законченность узлов (подсхем).

Ограничения

1. Вместимость конструктивного узла,
2. Число внешних выводов узла.

Математические модели объектов проектирования

Математическая модель СхЭПр ЭС – взвешенный по вершинам ги-

перграф. В отличие от задачи покрытия здесь веса вершин имеют другой смысл: это например площадь, занимаемая элементами в монтажном поле. Другой пример – это мощность теплоотдачи элемента.

Формализованная формулировка задачи

В терминах введенных ММ разбиения схем сводится к разбиению гиперграфа, взвешенного по вершинам, на подграфы, для которых указаны ограничения на суммарный вес их вершин и число внешних гиперребер.

Сведение к задаче математического программирования

Разбиение схем ЭС можно свести к задаче дискретного нелинейного программирования (ДНП).

Алгоритмы решения

1. Метод ветвей и границ.
2. Последовательные алгоритмы.
3. Итерационные алгоритмы.
4. Дихотомические алгоритмы.

3.3. Описание программы DECOM-3

В лабораторной работе исследуется последовательный алгоритм разбиения схем. Схема программы DECOM-3 представлена на рис. 1.

Входные данные

E - количество элементов схемы;

H - число электрических цепей схемы;

V - число внешних цепей схемы;

B - число выводов элемента;

G - максимальная вместимость узла;

W - максимальное число внешних выводов узла;

$CB(E, B)$ - список цепей схемы. Например, $CB_{17,9} = 37$ означает, что к 9-му выводу 17-го элемента присоединена 37-я цепь;

$GE(E)$ - вес элементов (принимает значения 1, 2, ...). Например, если $GE_{17} = 1$, а $GE_{41} = 4$, то это означает, что для 41-го элемента потребуется площадь в 4 раза большая, чем для 17-го;

$CV(V)$ - список внешних цепей схемы.

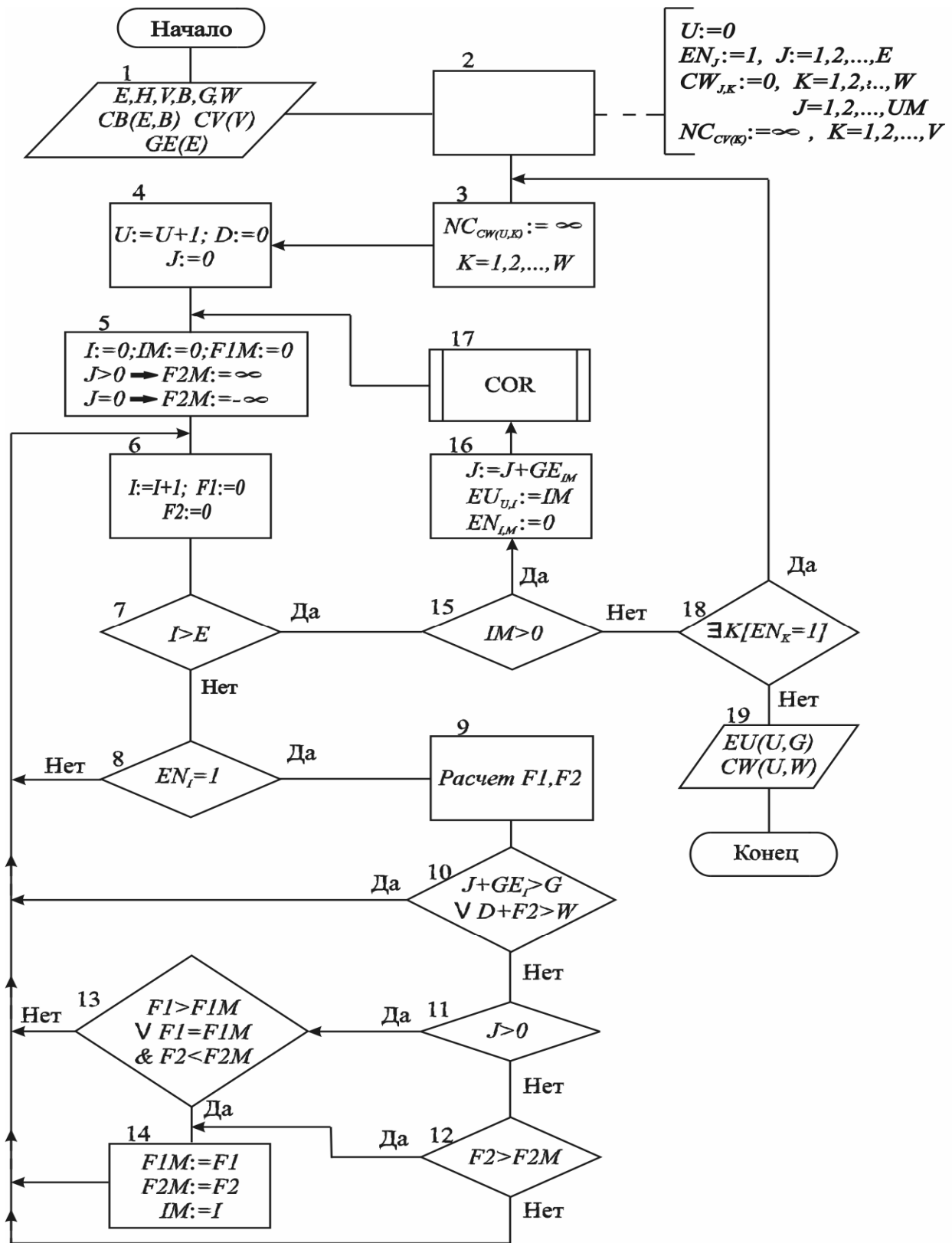


Рис.1. Схема программы DECOM-3

Выходные данные

U - число сформированных узлов;

$EU(U, G)$ - массив элементного состава узлов;

$CW(U, W)$ - массив межузловых соединений. Например, $CW_{3,12} = 87$ означает, что на 12-й вывод соединителя 3-го узла выходит 87-я цепь.

Промежуточные данные

I - номер очередного анализируемого элемента;

J - степень заполнения монтажного пространства формируемого узла элементами к данному шагу. Если все корпуса элементов одинаковы, то J - текущее число элементов в формируемом узле;

IM - номер оптимального элемента на данном шаге;

$F1, F1M$ - число цепей, общих для I -го (IM -го) элемента и формируемого узла;

$F2, F2M$ - приращение числа внешних цепей узла при включении в него I -го (IM -го) элемента;

D - текущее число внешних цепей узла;

$EN(E)$ - вектор нераспределенных элементов. Если $EN_i = 1$, то I -й элемент не распределен еще в узлы; $EN_i = 0$ - в противоположном случае;

$NC(H)$ - число не распределенных элементов каждой цепи. Пусть, например, "размер" 81-й цепи равен 5, т.е. эта цепь соединяет 5 элементов. Тогда, если на данном шаге алгоритма $NC_{81} = 3$, то это означает, что три из 5 элементов 81-й цепи находятся уже в узлах;

A - номер цепи на K -м выводе IM -го элемента ($A = CB_{IM, K}$);

P, Q, X, K - вспомогательные переменные.

Описание схемы программы DECOM-3

В блоке 2 происходит присвоение начальных значений параметрам: $U := 0$, поскольку еще ни один узел не сформирован; $EN_i := 1$ ($i = 1, 2, \dots, E$), поскольку все элементы еще не распределены в узлы; список $CW_{JK} := 0$ ($J = 1, 2, \dots, E$; $K = 1, 2, \dots, W$) пуст. Затем рассчитывается начальный состав списка NC_J , $J = 1, 2, \dots, H$. Поскольку внешние цепи схемы никогда не смогут полностью попасть внутрь узлов, происходит присвоение $NC_{CV(k)} := \infty$, ($k=1, 2, \dots, V$).

Формирование очередного узла начинается с блока 3. Цепи, которые стали внешними для U -го узла, сформированного перед входением в блок 3, уже не смогут полностью попасть внутрь формируемых в дальнейшем узлов, поэтому $NC_{CW(U,K)} := \infty$, ($K=1,2,\dots,W$). Первоначально узел

пуст ($J:=0$), и его внешние цепи отсутствуют ($D:=0$) (блок 4). Затем перебираются все элементы из числа нераспределенных в узлы (блоки 6, 7, 8). Для очередного I -го элемента рассчитываются критерии качества $F1$ и $F2$ (блок 9), и если этот элемент удовлетворяет ограничениям на вместимость узла и число внешних цепей узла (блок 10) и при этом оказывается лучше, чем все ранее рассмотренные элементы (блок 13), то в блоке 14 запоминается его номер IM и соответствующие значения критериев качества $F1M$ и $F2M$.

Блок 11 передает управление блоку 12 или блоку 13 в зависимости от того, пуст или частично заполнен узел. Если узел не пуст ($J > 0$), то элемент отбирается по максимуму критерия $F1$. Если оказалось, что лучший из ранее рассмотренных IM -й элемент имеет такое же значение критерия $F1M$, то отбор идет по минимуму критерия $F2$ (блок 13).

Для случая, когда узел пуст, т.е. в узел включается первый элемент, выбор элемента осуществляется по максимуму критерия $F2$ (блок 12).

Контрольная задача Test27dec

Пусть требуется выполнить разбиение электрической схемы, приведенной на рис. 2 (ее гиперграф показан на рис. 3) при вместимости узлов $G=9$ и ограничении на число внешних выводов узла $W=18$.

Входные данные (файл BDP.DEC) имеют вид:

27 68 16 14 9 18	}	E, H, V, B, G, W
12 0 0 0 0 0 0 15 14 13 0 11 0 0	}	$CB(27, 14)$
66 0 17 13 16 0 0 11 18 66 0 17 0 0		
.....		
50 0 63 66 10 0 0 11 50 50 0 68 0 0	}	$GE(27)$
1 1 1 1 1 1 1 1 1 1 1 1 1 1		
1 1 1 1 1 1 1 1 1 1 1 1 1 1	}	$CV(16)$
1 2 3 4 5 6 7 8 9 10 11 14 16 61 62 64		

Выходные данные (файл BDP.DER) имеют вид:

3	}	U
4 10 14 13 12 15 17 11 21	}	$EU(3, 9)$
22 23 24 27 16 2 3 1 25		
5 8 7 6 9 19 18 20 26		
47 7 22 6 4 12 2 8 9 35 1 34 3 45 5 56 46 0	}	$CW(3, 18)$
12 64 47 53 4 22 7 10 11 49 45 13 16 1 19 14 9 0		
34 35 19 14 62 53 13 49 56 61 46 0 0 0 0 0 0 0		

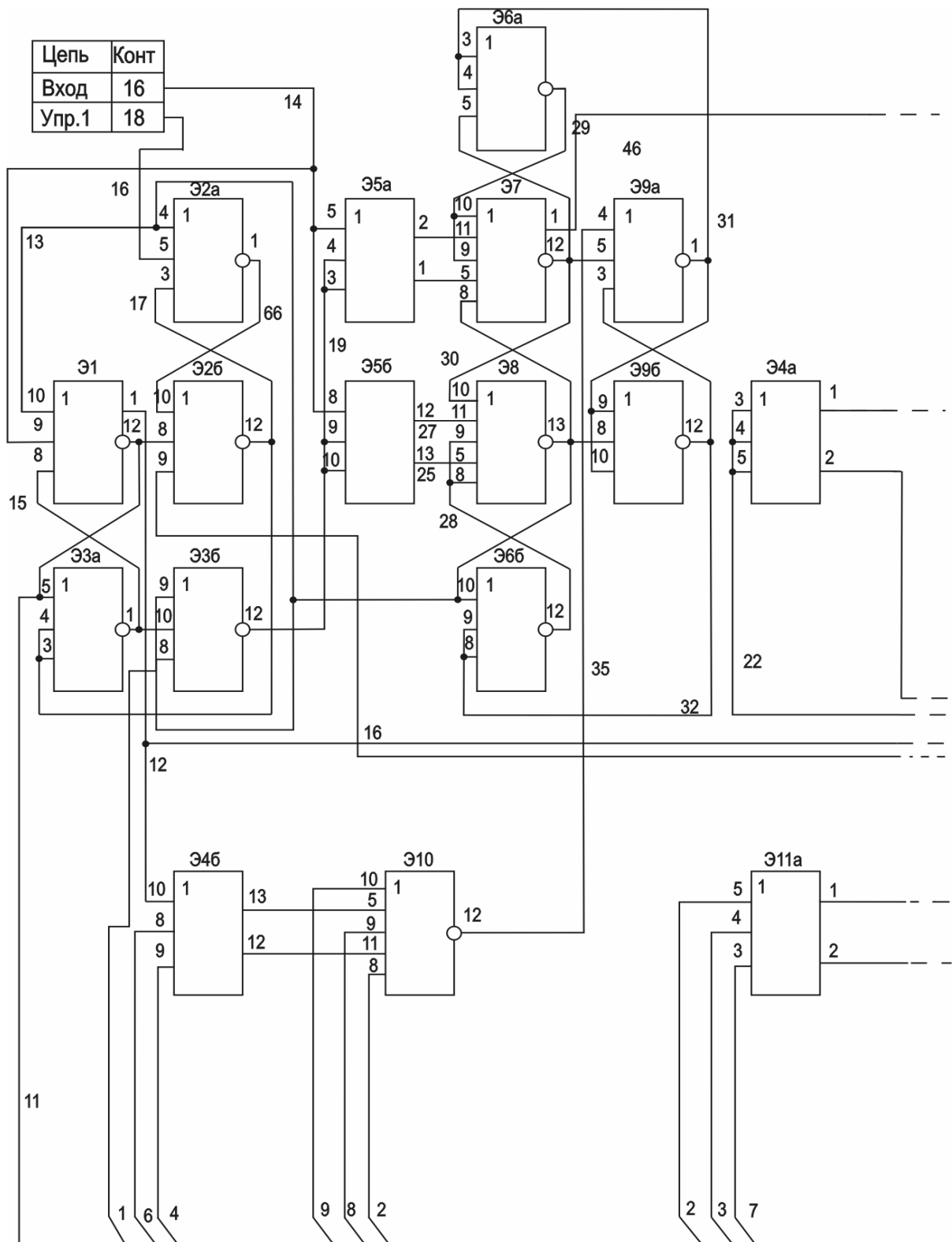


Рис. 2. Схема электрическая

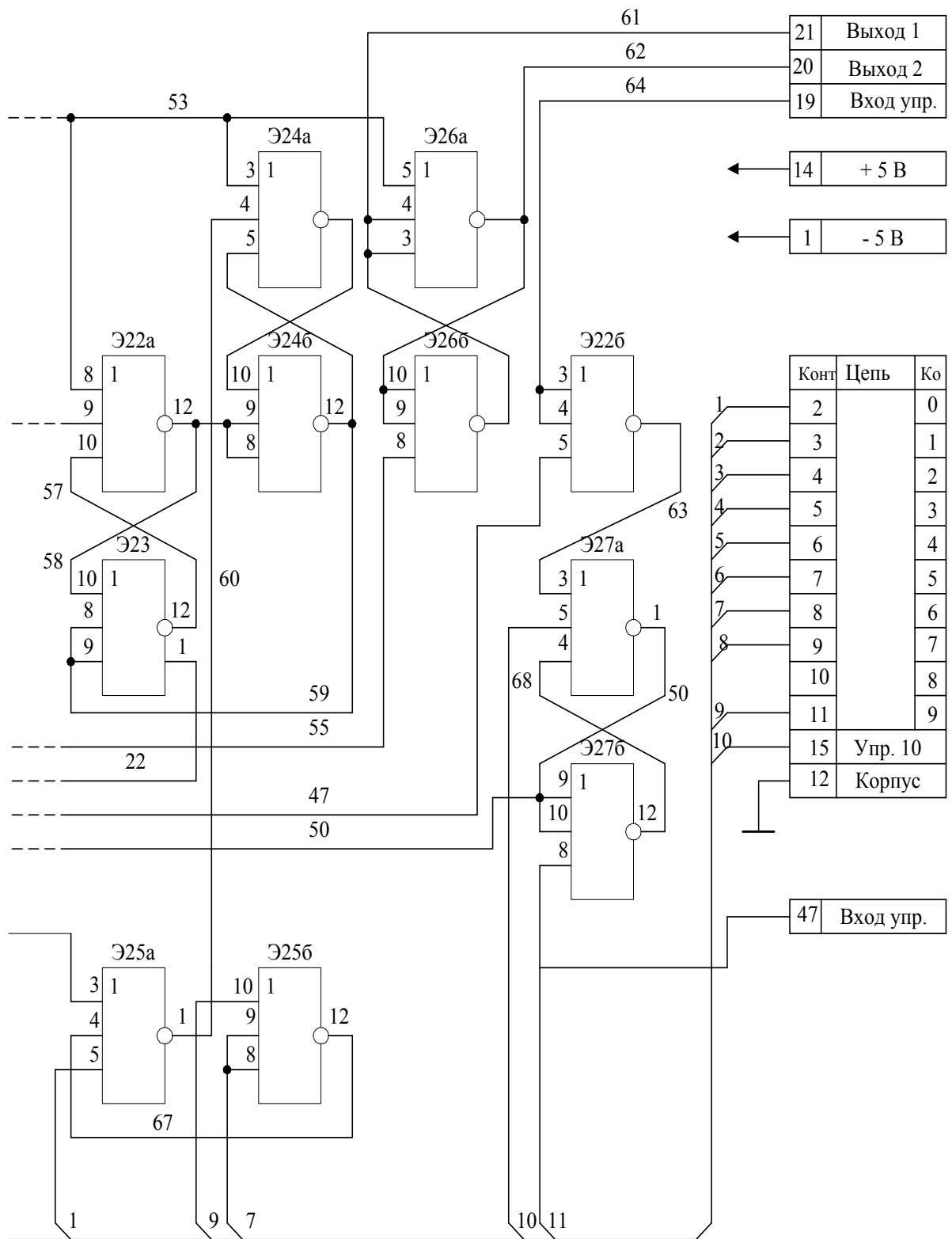


Рис. 2. Окончание.

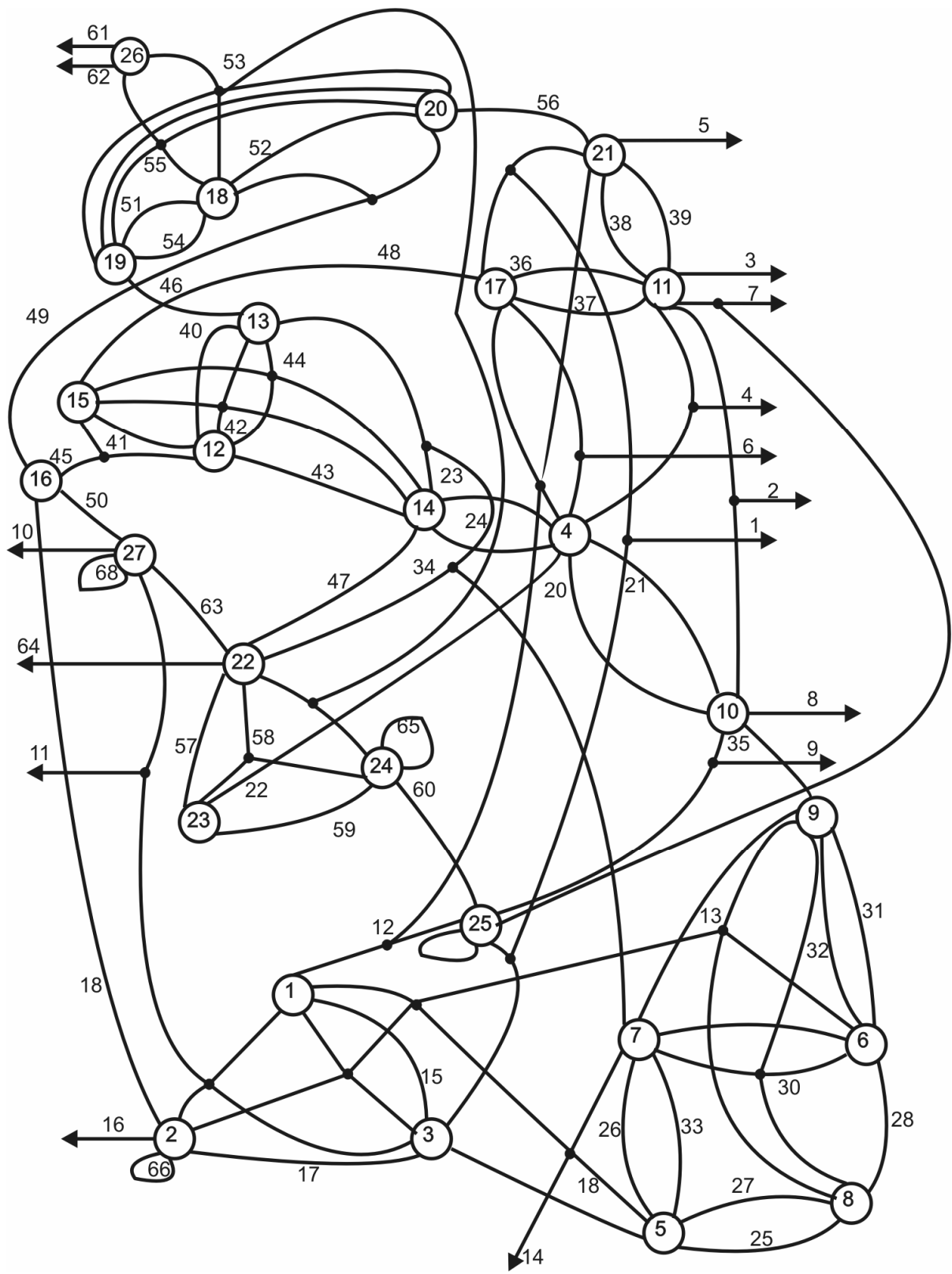


Рис.3. Гиперграф схемы

Полученное проектное решение представлено на рис.4 в виде 3-х под-схем (3-х подграфов исходного гиперграфа).

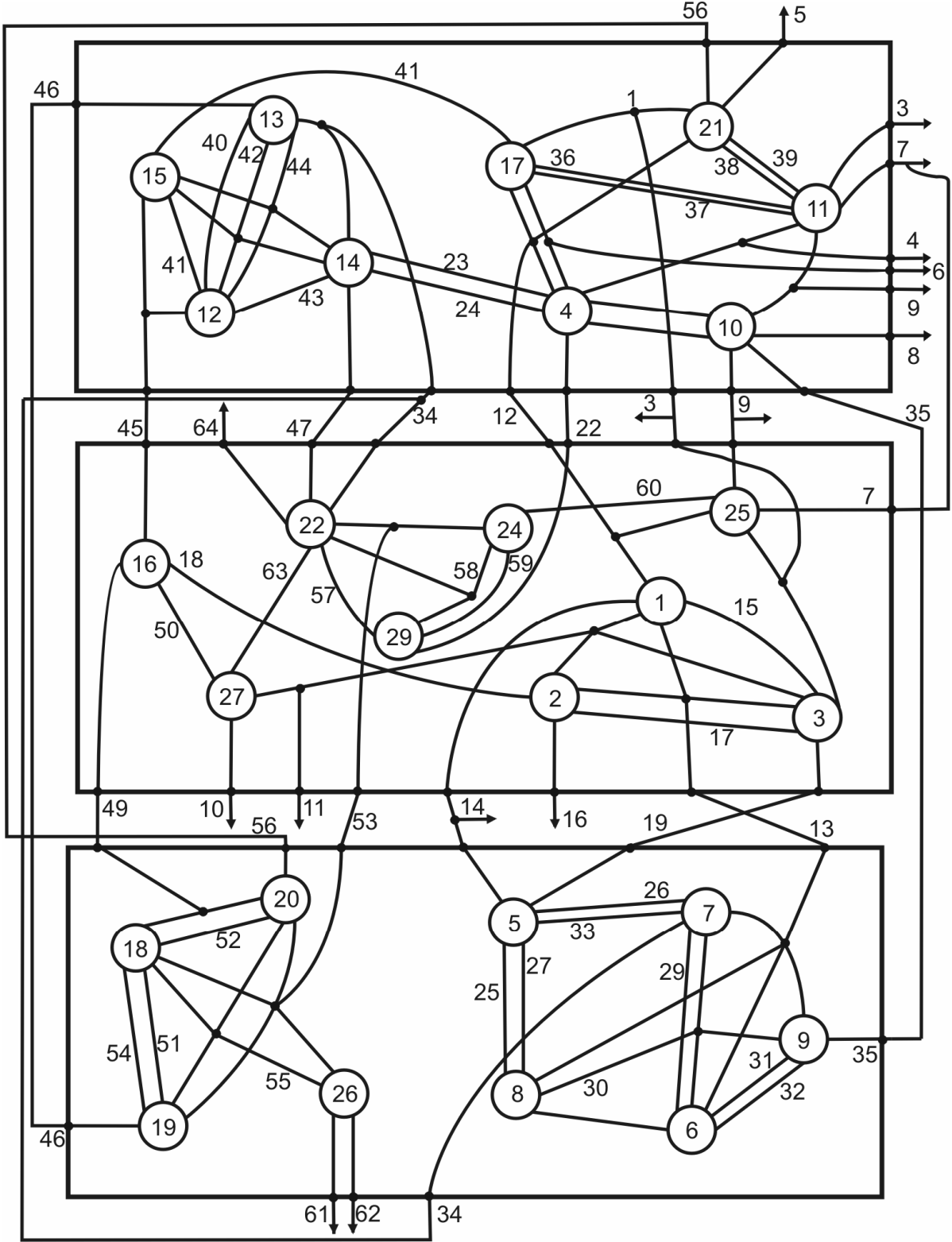


Рис.4. Результат разбиения схемы

3.4. Описание программы CODER-3

Назначение программы

Программа CODER-3 предназначена для облегчения подготовки исходных данных для лабораторных работ 2, 3, 4. Наиболее трудоемкая часть подготовки исходных – формирование списка цепей схемы (файл BDP.CIR). Программа CODER-3 используется в том случае, когда студент получает задание на лабораторные работы 2, 3, 4 по результатам, полученным в результате разбиения схемы при выполнении 1-ой лабораторной работы. В этом случае преподаватель указывает студенту, какую именно подсхему он должен взять в качестве исходной для дальнейшего проектирования ячейки ЭС.

Например, в разделе 3 для контрольной задачи из трех полученных в результате разбиения подсхем была выбрана подсхема № 2.

Техническая проблема при формировании исходных данных заключается в требовании перенумерования (перекодировки) конструктивных элементов так, чтобы они образовали ряд 1, 2, 3..., N, а также электрических цепей 1, 2, ... , C. Это требование обусловлено 2-мя причинами.

Первая (и главная) причина методическая: студентам легче в этом случае готовить данные и анализировать результаты работы программ при выполнении лабораторных работ 2, 3, 4.

Вторая причина – при такой нумерации элементов и цепей упрощаются программы PLACE-3, TLO-3, TRACE-3.

Входные данные программы CODER-3

1. Файл BDP.DEC
2. Файл BDP.DER
3. NCIR – номер заданной программы подсхемы.

Содержание файлов BDP.DEC и BDP.DER описано в лабораторной работе 1 (подраздел 3.2). Эти файлы программа находит на магнитном диске по их именам.

Номер NCIR заданной преподавателем подсхемы вводится в программу CODER-3 с клавиатуры.

Выходные данные программы CODER-3

1. Файл BDP.CIR
2. Файлы TAB1 и TAB2

Содержание файла BDP.CIR описано выше. Файлы TAB1 и TAB2 содержат информацию о старых и новых номерах элементов и цепей соответственно.

3.5. Задания на лабораторную работу и УИРС

1. При теоретической подготовке использовать данные методические указания, контрольные вопросы и цитированную литературу. Вычертить схему программы последовательного алгоритма разбиения схем и разобратся в работе алгоритма.

2. Получить у преподавателя задание в одном из трёх вариантов:

а) студенту предлагается готовая схема электрическая принципиальная ЭС;

б) студенту предлагается схема электрическая функциональная (логическая). В этом случае студент должен вручную решить задачу покрытия этой схемы набором ИМС подходящей серии (набором ФТУ) и получить СхЭПр;

в) студенту предлагается поведенческое описание цифрового ЭС (комбинационная логика), например в виде таблицы истинности. В этом случае студент должен синтезировать логическую схему (получить совершенную дизъюнктивную нормальную форму (СДНФ) и минимизировать булевы функции), решить задачу покрытия СхЭФ набором ИМС (ФТУ).

3. Подготовить входные данные и решить задачу с помощью программы DECOM-3. Оценить качество решения. Посчитать число межзловых цепей. Проанализировать, можно ли улучшить полученное проектное решение, например, путём перестановок элементов.

4. Задания УИРС:

а) исследовать влияние ограничений G и W на результаты решения. Для этого выполнить серию решений для одной и той же схемы, который раз меняя параметры G и W ;

б) исследовать временную сложность алгоритма $O(E)$. Для этого провести серию решений для схем различной размерности с фиксацией времени решения;

в) решить задачу с применением разных алгоритмов и проанализировать результаты.

5. Отчёт должен содержать:

- цель работы;
- краткие теоретические сведения, которые можно использовать при защите лабораторной работы;
- схему программы DECOM-3;

- задание на лабораторную работу;
- входные данные, выходные данные и рисунок, аналогичный рис. 4. Под рисунком указать полученное значение критерия качества. Для контрольной задачи это выглядит так:

$$F = 17+17+11 = 45;$$

- Оценку полученного решения и попытку улучшить его вручную. Результаты отразить в анализе результатов.
- анализ полученных результатов.

Контрольные вопросы

1. Этапы и уровни проектирования ЭС. Проектная процедура, проектное решение, проектная операция. Терминология. Схема процесса проектирования.
2. Способы задания множеств. Алгебра множеств. Кортежи.
3. Отношения на множествах. Функциональные отношения и отображения. Отношения эквивалентности и отношение порядка. Реляционные базы данных.
4. Нечеткие множества. Нечетные алгоритмы.
5. Эмпирические свойства алгоритмов. Способы представления алгоритмов. Критерии оценки и сравнения алгоритмов. Алгоритмы полиномиальной и экспоненциальной сложности.
6. Проблема уточнения понятия алгоритм. Машина Тьюринга. Тезис Тьюринга.
7. Классификация и критерии сравнения математических моделей (ММ). Нечеткий алгоритм построения ММ.
8. Покрытие электрических схем ФТУ: содержательная формулировка задачи, входные и выходные данные, математические модели объектов проектирования, формализованная формулировка и алгоритмы решения.
9. Разбиение схем ЭС: содержательная формулировка задачи, входные и выходные данные, математические модели объектов проектирования, формализованная формулировка, комбинаторный анализ и алгоритмы решения.
10. Классы задач математического программирования.
11. Нелинейное дискретное программирование. Методы решения.
12. Поясните работу каждого блока в схеме программы DECOM-3.

4. РАЗМЕЩЕНИЕ ЭЛЕМЕНТОВ НА КОММУТАЦИОННОЙ ПЛАТЕ И РАСПРЕДЕЛЕНИЕ ЦЕПЕЙ ПО ВЫВОДАМ УЗЛА

Лабораторная работа № 2

4.1. Описание проектной задачи размещения элементов на коммутационной плате

Монтажное пространство конструктивного узла, в котором необходимо оптимально разместить элементы, может быть двумерным и трёхмерным. Двумерное пространство называют также монтажным полем. Задачи размещения элементов разделяются на два класса: размещение одногабаритных элементов и размещение разногабаритных элементов. Задачи первого класса проще, поскольку заранее известны посадочные места под элементы. В этом случае расположение посадочных мест в монтажном поле рассчитывается исходя из заданных шагов установки конструктивных элементов (ИМС). Шаги установки в свою очередь определяются плотностью электро монтажа для данной схемы и заданных размеров платы. В этом случае задачу размещения элементов можно сформулировать как оптимальное отображение множества элементов на множество посадочных мест. В лабораторной работе в дальнейшем рассматривается задача размещения одногабаритных элементов.

Входные данные задачи

1. Схема принципиальная электрическая, которая обычно задается списком цепей.
2. Описание монтажного пространства конструктивного узла (размеры платы, координаты посадочных мест, координаты выводов узла (контактов соединителей), координаты и размеры запрещенных зон).
3. Описание посадочного места под конструктивный элемент.

Выходные данные

Вектор (кортеж) распределения элементов по посадочным местам. Зная этот вектор и координаты посадочных мест, можно определить координаты элементов.

Критерии качества задачи и ограничения

- При размещении элементов стремятся достичь следующие цели:
- топологические: необходимо создать наилучшие условия для по-

следующей трассировки проводников;

- электрические: минимум искажения логических сигналов в проводниках, минимальная задержка сигнала и др.

- тепловые: равномерное распределение тепловыделяющих элементов по плате и т.п.

В большей или меньшей степени эти цели достигаются за счет применения следующих простых (быстро считаемых) критериев качества.

1. Минимум суммарной длины соединений.
2. Минимум длины самого длинного проводника.
3. Максимум числа цепей, которые соединяют элементы, находящиеся в соседних посадочных местах.
4. Минимум числа пересечений проводников.
5. Максимум числа соединений с возможно более простой конструкцией.

Ограничением задачи часто является ограничение на длину самого длинного проводника (или конкретных проводников), что определяет время распространения сигнала в этих проводниках, подверженность помехам.

Математические модели (ММ) объектов проектирования

(электрической схемы и монтажного пространства)

В качестве ММ схемы обычно применяется граф, взвешенный по ребрам. Методика расчета весов ребер изложена в [2].

ММ монтажного пространства – регулярная прямоугольная решетка, каждая ячейка которой определяет соответствующее посадочное место.

Формализованная формулировка задачи

В терминах введенных математических моделей техническую задачу размещения элементов в монтажном пространстве можно сформулировать как математическую задачу оптимального вложения взвешенного по ребрам графа в прямоугольную решетку.

Сведения к задаче математического программирования

С использованием введенных математических моделей задачу размещения можно свести к хорошо изученной задаче квадратичного назначения [7]. Это комбинаторная задача экстремального типа относится к классу дискретных нелинейных оптимизационных задач (дискретное нелинейное программирование).

Алгоритмы размещения элементов [1, 5, 6].

1. Метод ветвей и границ.
2. Последовательные алгоритмы.
3. Итерационные (парных перестановок и групповых перестановок).
4. Дихотомические.
5. Силовые (алгоритм попарных релаксаций и алгоритм Гото).

Метод ветвей и границ дает точное решение, остальные алгоритмы относятся к классу эвристических.

4.2. Описание проектной задачи распределения электрических цепей по выводам конструктивного узла (РЦВУ)

При размещении элементов на плате уже имеется некоторый вариант РЦВУ. Оптимизация РЦВУ выполняется при фиксированном размещении элементов.

Входные данные задачи

1. Схема принципиальная электрическая проектируемого узла.
2. Описание монтажного пространства конструктивного узла: размеры платы, координаты посадочных мест, координаты выводов узла (контактов соединителей), координаты и размеры запрещенных зон.
3. Описание посадочного места под конструктивный элемент.
4. Результаты размещения элементов (вектор распределения элементов по посадочным местам).

В данной лабораторной работе считается, что каждый соединитель описывается таким же посадочным местом, что и элементы (ИМС), исходя из этого координаты выводов узла вычисляются в программе.

Выходные данные

Вектор (кортеж) распределения электрических цепей по выводам узла.

В данной лабораторной работе соединители включены в общий список цепей схемы и выходные данные – это изменения в последних R строках списка цепей после обработки описания проекта подпрограммой LEAD-3.

Критерии качества и ограничения

1. Минимум суммарной длины соединений.
2. Минимум числа пересечений проводников.

Могут быть заданы ограничения на взаимное назначение некоторых цепей на выводы узла, например для усилителя следует разнести его входные и выходные цепи.

Математические модели

Задачу РЦВУ можно формально описать с помощью полного двудольного графа, взвешенного по ребрам. Вершины одной доли графа – внешние электрические цепи (т.е. те, которые необходимо вывести из конструктивного узла), вершины второй доли – внешние выводы узла (т.е. контакты соединителей узла).

Вес E_{ik} ребра двудольного графа рассчитываются как кратчайшая длина проводника при назначении i -ой электрической цепи на k -й вывод соединителя.

Результат решения задачи РЦВУ – максимальное паросочетание двудольного графа с минимальным весом ребер.

Формализованная формулировка задачи

Техническая задача РЦВУ сводится к математической задаче определения на взвешенном по ребрам полном двудольном графе максимального паросочетания с минимальным весом ребер.

Сведения к задаче математического программирования

С использованием введенной математической модели РЦВУ можно свести к линейной задаче оптимального назначения [7]. Это частный случай целочисленного линейного программирования (ЦЛП).

Алгоритмы РЦВУ

1. Венгерский алгоритм.
2. Метод ветвей и границ (МВГ).
3. Последовательный алгоритм.
4. Дихотомический алгоритм.

Первые два алгоритма дают точное решение, остальные алгоритмы – эвристические.

4.3. Описание программы PLACE-3

В программе PLACE-3 для размещения элементов реализован алгоритм парных перестановок [1, 5, 6]. Схема программы представлена на рис. 5.

Входные данные

Файл BDP.PAR (параметры массивов):

N – число элементов (с учетом "пустых" элементов и соединителей);

R – число вертикальных рядов элементов (число соединителей);

B – число выводов конструктивного элемента;

C – число цепей схемы;

RM – технологическое ограничение на число проводников, которые разрешено присоединить к одному контакту. Для печатного монтажа $RM = \infty$;

S – число слоев коммутационной платы;

Z – число запрещенных для трассировки зон;

DX, DY – размеры платы.

Файл BDP.CIR (описание схемы в виде списка цепей):

$CE(N, B)$ – список цепей схемы. Например, $CE_{8,12} = 37$ означает, что к 12-му выводу 8-го элемента присоединена 37-я цепь.

Файл BDP.CON (описание монтажного пространства коммутационной платы):

$XP, YP(N)$ – координаты посадочных мест относительно левого нижнего угла платы;

$XZ, YZ, ZX, ZY(Z)$ – координаты и размеры запрещенных зон.

Файл BDP.EL (описание посадочного места под элементы):

T – тип элемента; определяет, планарные или штыревые выводы имеет элемент;

BX, BY – размеры контактных площадок под выводы элемента;

$XB, YB(B)$ – координаты контактных площадок посадочного места относительно контактной площадки под первый вывод элемента.

Файл BDP.PE (описание размещения элементов):

$PE(N)$ – номера посадочных мест, в которые установлены элементы. Например, $PE_{12} = 7$ означает, что 12-й элемент установлен в 7-ое посадочное место; $PE_8 = 0$ означает, что 8-й элемент ещё не установлен.

Этот файл нужен во входных данных в связи с тем, что реализованный в программе алгоритм относится к классу итерационных, для которых требуется начальное решение (начальное размещение в данной задаче).

Выходные данные

$PE(N)$ – массив размещения элементов по посадочным местам.

$CE(N, B)$ (последние R строк) – новое распределение внешних цепей схемы по выводам конструктивного узла (по выводам соединителей), полученное в результате работы подпрограммы LEAD-3.

Промежуточные данные

I, J – номера очередной пары элементов – кандидатов на перестановку;

F – переменная для счета числа больших итераций алгоритма;

L – приращение целевой функции (суммарной длины соединений) при перестановке I -го и J -го элементов;

LM – максимальное отрицательное приращение L ;

JM – номер элемента, наилучшего из J -х элементов;

$MC(N, N)$ – взвешенная матрица смежности графа схемы;

$ML(N, N)$ – матрица расстояний между посадочными местами элементов;

$RC(C)$ – "размеры" электрических цепей. Например, $RC_{47} = 5$ означает, что 47-я цепь объединяет пять выводов.

Внимание! При подготовке исходного описания проекта следует учесть следующие ограничения:

1. Все элементы (в том числе соединители) размещаются по одинаковым посадочным местам регулярно вертикальными и горизонтальными рядами.

2. Число соединителей равно R – числу вертикальных рядов элементов.

3. Соединители всегда находятся внизу платы. Число элементов (вместе с соединителями) равно числу посадочных мест N . При необходимости вводят "пустые" элементы.

4. При назначении цепей на выводы соединителей следует использовать только "верхние" (номера 7, 8, ..., 14) контактные площадки посадочных мест под соединители.

5. Размер дискрета ДРП (дискретного рабочего поля [2]) равен шагу выводов элемента (ИМС).

6. Все координаты и размеры, заданные в мм, следует пересчитать в количество дискретов ДРП.

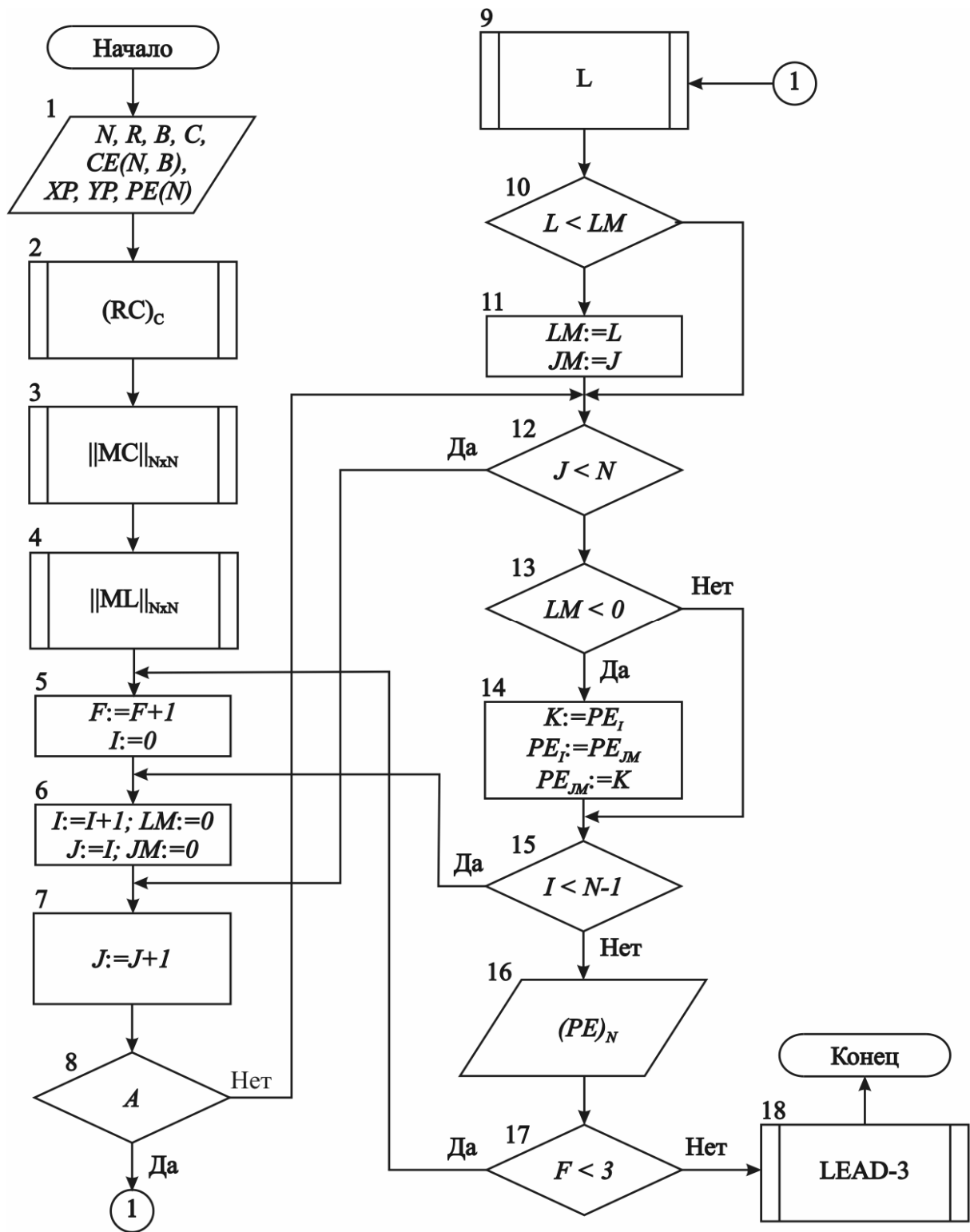


Рис.5. Схема программы PLACE-3

7. Предполагается, что цепи питания выполнены в виде навесных шин либо расположены в отдельном слое печатной платы. Поэтому в списке цепей схемы для соответствующих выводов ИМС запишите нули.

8. Описание посадочного места (файл BDP.EL) можно не создавать вручную, а скопировать его из библиотеки элементов (папка LIBRARY).

9. Для ускорения подготовки файла BDP.CIR можно воспользоваться программой CODER-3 (см. подраздел 3.4).

Описание схемы программы PLACE-3

В блоках 2 и 3 рассчитывается матрица $MC(N, N)$, в блоке 4 - матрица $ML(N, N)$. В блоках 6, 7, 12, 15 к очередному I -му элементу подбирается J -й элемент и вычисляется приращение целевой функции (блок 9).

Номер элемента JM , дающий наибольшее отрицательное приращение при его перестановке с I -м элементом, запоминается (блок 11).

В блоке 8 проверяется условие А:

$$I \leq N-R \ \& \ J \leq N-R \vee I > N-R \ \& \ J > N-R, \quad (1)$$

которое исключает возможность перестановки элементов с соединителями.

Если удалось подобрать пару I, JM элементов, дающих отрицательное приращение (блок 13), то эти элементы переставляются (блок 14). Блоки 5, 17 организуют три большие итерации для новых попыток улучшить полученное на предыдущих итерациях размещение.

Контрольная задача Test3x4

В качестве задания взята 2-я подсхема, полученная в результате разбиения (рис. 4) в лабораторной работе № 1. После перекодировки номеров элементов и цепей схема представлена на рис. 6.

Например в результате переименования элемент № 2 из схемы, приведенной на рис. 2, получил номер 8, а цепь 64 получила номер 26 и т. д.

Задано, что элементы устанавливаются на плате тремя вертикальными рядами с шагом 37,5x20 мм. Размеры платы 125x85 мм. Шаг координатной сетки 2,5 мм (рис. 7).

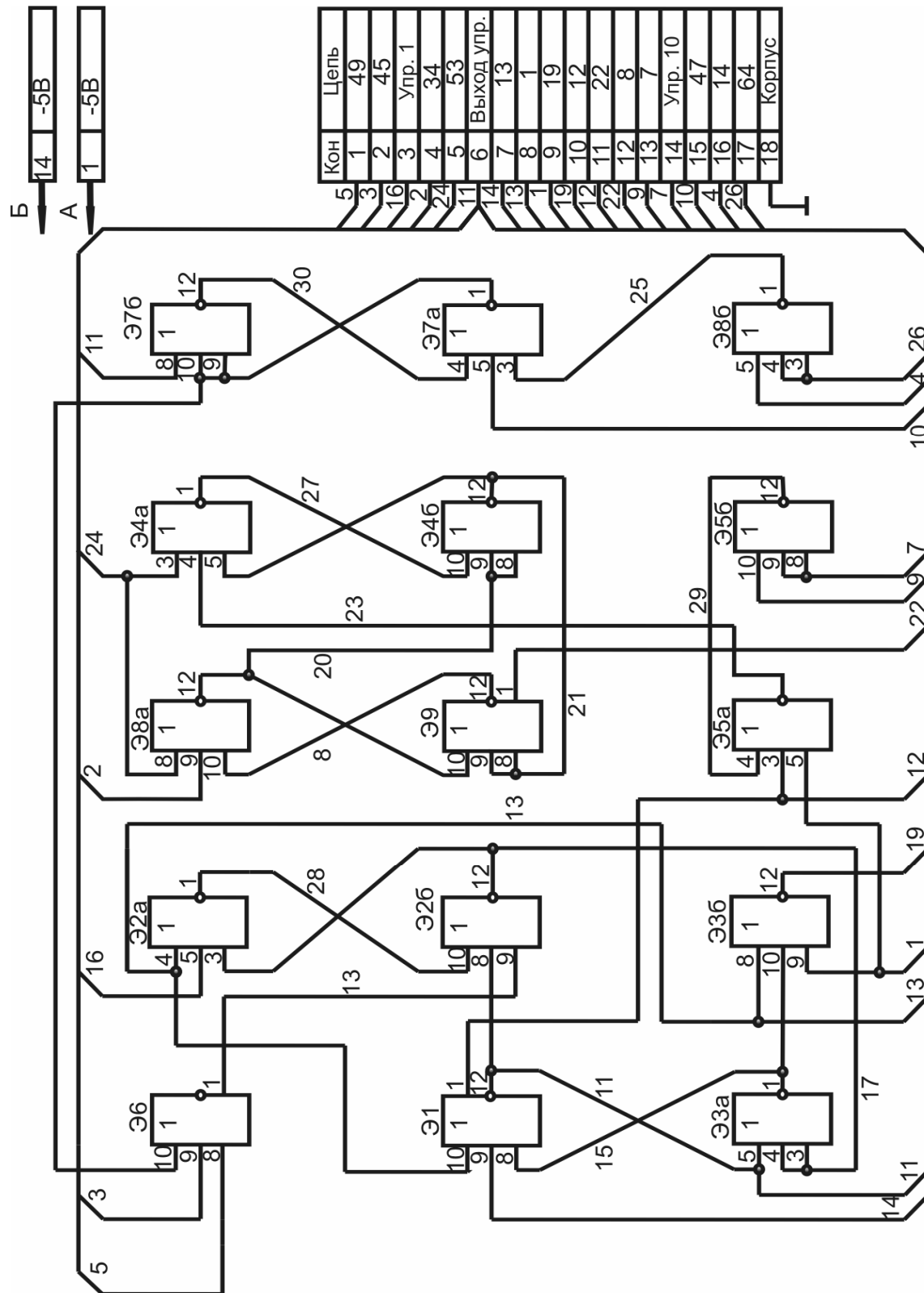


Рис. 6. Схема узла

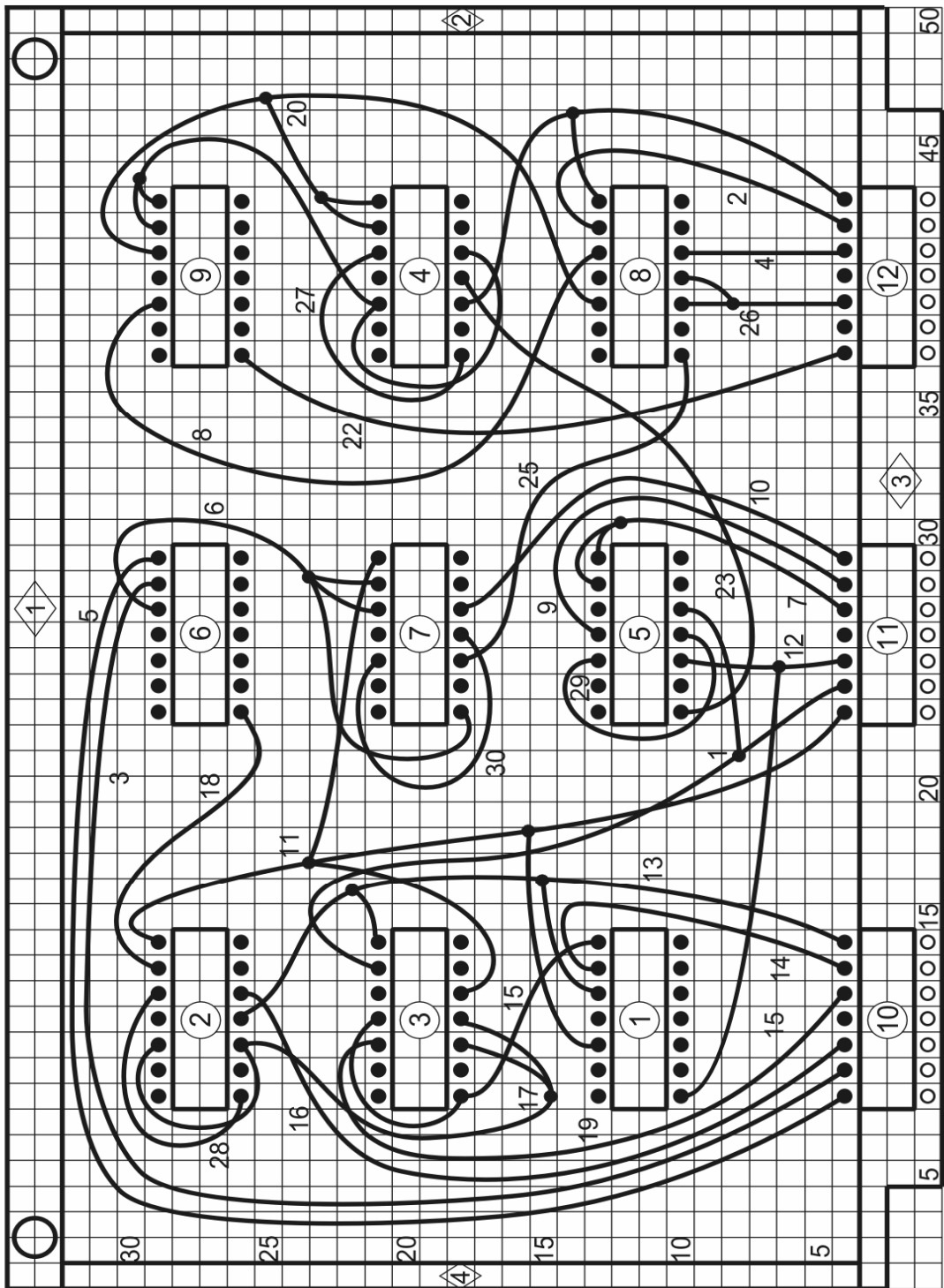


Рис. 7. Размещение элементов

Входные данные

12 3 14 30 88 2 4 50 34	} <i>N, R, B, C, RM, S, Z, DX, DY</i>		
12 0 0 0 0 0 0 15 14 13 0 11 0 0	} (Файл BDP.PAR)		
28 0 17 13 16 0 0 11 18 28 0 17 0 0			
15 0 17 17 11 0 0 13 1 15 0 19 0 0			
27 0 24 23 21 0 0 20 20 27 0 21 0 0			
23 0 12 29 1 0 0 7 7 9 0 29 0 0			
18 0 0 0 0 0 0 5 3 6 0 0 0 0		} <i>CE (12, 14)</i>	
6 0 25 30 10 0 0 11 6 6 0 30 0 0			} (Файл BDP.CIR)
25 0 26 26 4 0 0 24 2 8 0 20 0 0			
22 0 0 0 0 0 0 21 21 20 0 8 0 0			
0 0 0 0 0 0 0 13 14 19 0 16 3 5			
0 0 0 0 0 0 0 10 9 7 0 12 1 11			
0 0 0 0 0 0 0 24 2 4 0 26 0 22			
8 23 38 8 23 38 8 23 38 8 23 38			
1 1 1 10 10 10 18 18 18 26 26 26			
1 50 1 1			
33 4 1 4	} <i>XZ, YZ, ZX, ZY (4)</i>		
50 1 50 1		} (Файл BDP.CON)	
2 29 3 29			
22 1 1	} <i>T, BX, BY</i>		
1 2 3 4 5 6 7 7 6 5 4 3 2 1	} <i>XB, YB (14)</i>		
1 1 1 1 1 1 1 4 4 4 4 4 4 4	} (Файл BDP.EL)		
4 5 6 7 8 9 10 11 12 1 2 3	} <i>PE (12)</i>		
	(Файл BDP.PR)		

Выходные данные

4 10 7 9 5 11 8 6 12 1 2 3	} <i>PE (12)</i>	
	(Файл BDP.PR)	
.....	}	
0 0 0 0 0 0 0 0 14 13 16 11 19 22		} <i>CE (12, 14)</i>
0 0 0 0 0 0 0 5 7 1 9 10 3 0		
0 0 0 0 0 0 0 24 2 4 0 26 0 22		

Полученное проектное решение в виде рисунка (гиперграф схемы в решетке) представлено на рис. 8.

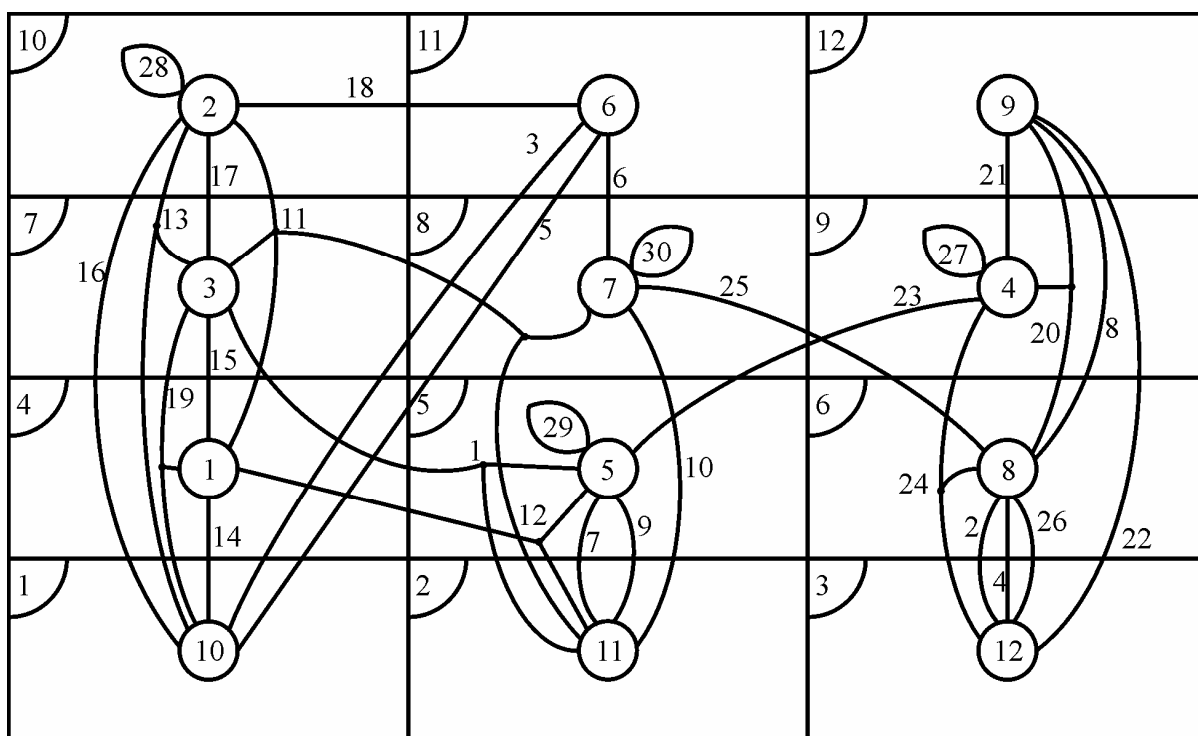


Рис. 8. Гиперграф в решетке.

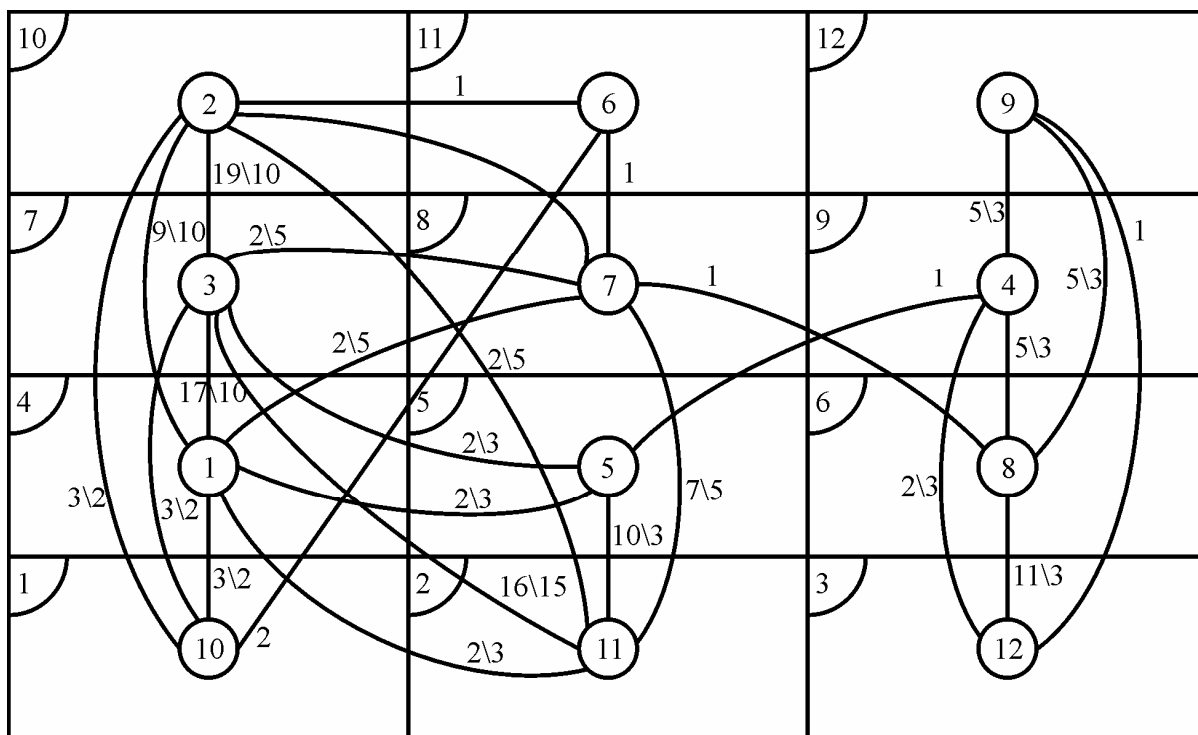


Рис. 9. Взвешенный по рёбрам граф в решётке.

Из описания работы программы PLACE-3 следует, что в качестве модели электрической схемы проектируемого узла используется взвешенный по ребрам граф, который представлен на рис. 9. Методика расчета весов ребер графа описана в [2].

В исходном описании задачи распределения цепей по выводам конструктивного узла (контактам соединителей) применена модель – взвешенный по ребрам полный двудольный граф (здесь не приводится). Полученное проектное решение этой задачи представлено на рис. 10 максимальным паросочетанием с минимальным весом ребер.

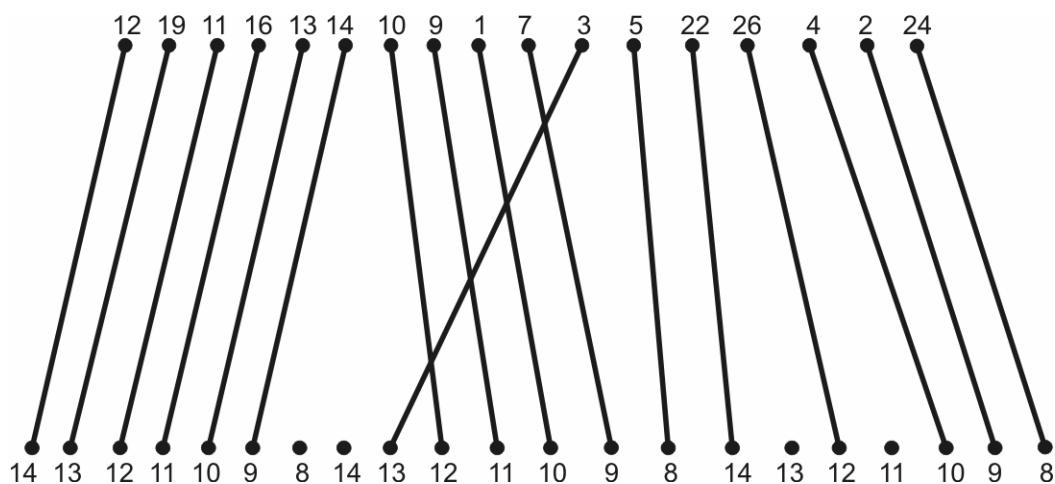


Рис. 10. Максимальное паросочетание с минимальным весом ребер

4.4. Задания на лабораторную работу и УИРС

1. При теоретической подготовке использовать данные, контрольные вопросы и цитированную литературу. Уяснить содержание и сущность задачи размещения элементов на коммутационной плате и распределения цепей по выводам узла. Изучить математические модели и алгоритмы решения этих задач. Вычертить схему программы PLACE-3 и разобраться в работе каждого блока.

2. Получить у преподавателя задание: электрическую принципиальную схему и параметры конструктивного узла. Схема может быть взята как результат решения задачи покрытия из предыдущей лабораторной работы. Это может быть также одна из подсхем, полученных в результате разбиения схемы в 1-й лабораторной работе.

3. Выполнить размещение элементов и распределение цепей по выводам узла вручную. Представить размещение в виде гиперграфа в решетке. Рассчитать суммарную длину соединений. Нарисовать фрагмент полного двудольного графа для задачи РЦВУ.

4. Подготовить исходные данные и решить обе задачи с помощью программы PLACE-3. Изобразить полученное решение в виде гиперграфа в решетке. Рассчитать суммарную длину соединений, записать ее значение под рисунком гиперграфа и сравнить с ручным решением. По результатам автоматизированного решения РЦВУ изобразить максимальное паросочетание с минимальным весом ребер.

5. Задания УИРС:

1) исследовать влияние начального размещения на работу алгоритма парных перестановок;

2) исследовать временную сложность $O(N)$ алгоритма размещения;

3) сравнить результаты работы разных алгоритмов размещения.

6. Отчет должен содержать:

- цель работы;
- краткие теоретические сведения, которые можно использовать при защите лабораторной работы;
- схему программы PLACE-3;
- задание на лабораторную работу;
- электрическую схему конструктивного узла;
- исходные данные и распечатку машинного решения;
- результаты ручного и машинного проектных решений в виде двух рисунков (гиперграфы в решетках). Под этими рисунками записать значения критериев качества;
 - на первом рисунке указать новые и старые (в скобках) номера элементов и цепей. Например 38(71) означает, что цепь при разбиении имела номер 71, теперь в схеме конструктивного узла ее номер 38.
 - полный двудольный граф и максимальное паросочетание для задачи РЦВУ.
 - анализ полученных результатов. Один из пунктов анализа - объяснение результатов сравнения ручного и машинного проектных решений.

Контрольные вопросы

1. Определение графа и гиперграфа. Способы задания графов. Разно- видности графов. Теорема Эйлера. Задачи о коммивояжере.
2. Числа графов. Метод Магу.
3. Размещение одногабаритных элементов ЭС на коммутационной плате: содержательная формулировка задачи, входные и выходные данные, математические модели объектов проектирования и алгоритмы решения.
4. Размещение разногабаритных элементов ЭС на коммутационной плате: содержательная формулировка задачи, входные и выходные данные, математические модели объектов проектирования и алгоритмы решения.
5. Распределение электрических цепей по выводам конструктивного узла: содержательная формулировка задачи, входные и выходные данные, математические модели объектов проектирования, формализованная фор- мулировка, комбинаторный анализ и алгоритмы решения.
6. Топологические модели электрических схем и их погрешности.
7. Объясните, как рассчитываются веса ребер графа схемы.
8. Методы измерения расстояний в монтажном пространстве конст- руктивного узла.
9. Линейное программирование: формулировка задачи и методы ре- шения.
10. Целочисленное линейное программирование: формулировка за- дачи и методы решения.
11. Объясните работу каждого блока схемы программы PLACE-3.

5. ПОСТРОЕНИЕ КРАТЧАЙШИХ СОЕДИНЕНИЙ, РАССЛОЕНИЕ МОНТАЖА И УПОРЯДОЧЕНИЕ СОЕДИНЕНИЙ

Лабораторная работа № 3

Задачу трассировки электрических соединений по причине ее сложности разделяют на 4 более простые подзадачи [1, 5, 6] :

1. Построение кратчайших соединений.
2. Расслоение монтажа.
3. Упорядочение соединений.
4. Прокладка трасс.

Эти задачи могут решаться в различном порядке, некоторые из них могут объединяться в одну. Например, задача расслоения монтажа может решаться до прокладки трасс , во время или после прокладки трасс.

В лабораторной работе изучаются первые три подзадачи. Четвертая подзадача рассмотрена в разделе 6.

5.1. Описание проектной задачи построения кратчайших соединений

Обычно в алгоритмах прокладки трасс задаются поочередно пары контактов (2 точки в монтажном поле), которые нужно соединить, т.е. электрические соединения.

Поэтому нужно предварительно преобразовать каждую электрическую цепь схемы в совокупность электрических соединений.

Входные данные

1. Описание СхПрЭ.
2. Конструктивно-технологическое описание монтажного пространства: размеры платы, шаги установки и ориентация элементов (или координаты посадочных мест), число коммутационных слоев, способ межслойных соединений, размеры межслойных соединений, области, запрещенные для трассировки (краевые поля, точки крепления платы, технологические зоны, заранее проложенные трассы и др.), минимальная ширина проводников, минимальные зазоры проводник-проводник, проводник-межслойный переход и др.
3. Описание посадочных мест под конструктивные элементы: это

размеры контактных площадок и координаты их взаимного расположения, а также тип выводов элемента: выводы планарные или штыревые.

4. Результаты размещения элементов и РЦВУ.

Выходные данные

Список электрических соединений, в котором для каждого соединения определены координаты двух его концов, а так же номер цепи, "породившей" это соединение.

Критерии качества

Обычно это минимум суммарной длины соединений, т.е. строятся кратчайшие соединения (соединения кратчайшей длины). В этой задаче под длиной соединения понимается расстояние между его концами.

Ограничения

Если проектируется проводной монтаж, то может быть задано технологическое ограничение на число проводников, которое разрешено присоединять к одному контакту.

Математические модели

Каждая электрическая цепь представляется полным графом, взвешенным по ребрам, который построен на контактах цепи. "Вес" ребер означают расстояние между соответствующей парой контактов.

В результате решения задачи для каждой цепи на полном графе определяется дерево минимальной длины.

Формализованная формулировка

Для полного графа, взвешенного по ребрам, найти покрывающее дерево минимальной длины.

Тип оптимизационной задачи

Это частный очень простой случай задачи ЦЛП.

Алгоритмы решения

1. Алгоритм Прима.
2. Алгоритм Краскала.

Оба алгоритма дают точное решение, если строится обыкновенное дерево (а не дерево Штейнера) и отсутствует указанное выше ограничение.

5.2. Описание проектной задачи расслоения монтажа

Термин "расслоение монтажа" означает распределение элементов монтажа по слоям коммутационной платы.

Таковыми элементами монтажа (объектами расслоения) могут быть следующие четыре :

1. Подсхемы.
2. Электрические цепи.
3. Электрические соединения.
4. Трассы (расслоение после прокладки трасс).

В учебной САПР CROCUS-3 объектами расслоения являются электрические соединения.

Входные данные

1. Список соединений.
2. Число коммутационных слоев платы.

Выходные данные

Вектор (кортеж) распределения соединений по слоям платы.

Критерии качества

1. Минимальная степень конфликтности между элементами монтажа в каждом слое. Под конфликтом понимается потенциальное пересечение проводников.
2. Минимальное число межслойных переходов.
3. Минимальное число слоев платы (если оно не задано в качестве ограничения).

Математические модели

Граф конфликтов.

Формализованная формулировка

Выполнить раскраску графа конфликтов заданным количеством цветов, минимизируя число конфликтных ребер.

Тип оптимизационной задачи

Экстремальная задача комбинаторного типа.

Алгоритмы решения

1. Метод Магу [12].
2. Итерационные алгоритмы (один из них рассмотрен в данном разделе).

5.3. Описание проектной задачи упорядочения соединений

Большинство алгоритмов прокладки трасс используют последовательную стратегию: трассы прокладываются поочередно.

Упорядочение соединений (может быть и цепей) заключается в оп-

ределении очередности, в которой будут прокладываться трассы соединений.

Входные данные

Список соединений.

Выходные данные

Упорядоченный список соединений.

Критерии качества (критерии оценки, сравнения соединений)

1. Длина соединения.
2. Степень конфликтности соединений.
3. Степень отклонения соединения от заданного в слое направления.

Математические модели

1. Прямолинейный отрезок.
2. Минимальный прямоугольник, охватывающий контакты соединения.

Формализованная формулировка

Упорядочение соединений сводится к известной задаче сортировки массивов.

Алгоритмы решения

1. Сортировка перестановкой (метод "пузырька" и быстрая сортировка).
2. Сортировка выбором.
3. Сортировка вставлением.

5.4. Описание программного модуля TLO-3

Программный модуль TLO-3 содержит 3 подпрограммы (TREE, LAY, ORDER), которые решают 3 первые подзадачи трассировки. На рис. 11 представлена схема программы TLO-3, где подробно раскрыта подпрограмма TREE.

На рис.12 представлена схема подпрограммы LAY.

Входные данные

Входные данные программы TLO-3 - пять файлов описания проекта, обработанные программой PLACE-3 (см. раздел 3). Новых файлов программа PLACE-3 не создает, но изменяет файлы BDP.PR и BDP.CIR.

Выходные данные

M - число соединений;

$X1, Y1, X2, Y2, CC, SC (M)$ - упорядоченный список соединений (координаты концов соединений; номер цепи соединений; слой, в который попало соединение).

Промежуточные данные подпрограммы TREE

F - переменная для счета числа цепей, для которых уже построены соединения;

M - переменная для счета числа соединений, включенных в список;

K - переменная для счета числа выбранных соединений (ребер дерева) очередной цепи;

Q - число контактов, соединяемых очередной цепью;

$XC, YC (Q)$ - координаты концов цепи (вершин дерева);

$ML (Q, Q)$ - матрица длин ребер полного графа, на вершинах которого строится дерево;

$RV (Q)$ - вектор локальных степеней вершин дерева;

$MV (Q)$ - вектор меток вершин. Метка каждой вершины может принимать значение 1 или 0 в зависимости от того, включена или нет данная вершина в дерево к данному моменту времени.

Промежуточные данные подпрограммы LAU

$MC (M, M)$ - матрица смежности графа конфликтов соединений;

$SC (M)$ - вектор цветов, в которые окрашены вершины. Например, $SC_5=2$ означает, что 5-я вершина графа конфликтов окрашена во второй цвет (т. е. пятое соединение помещено во второй слой);

F - переменная для счета числа итераций алгоритма раскраски;

I, J - номера вершин графа конфликтов;

K - номер цвета вершин (номер слоя платы);

$KV(S)$ - количество вершин P -го цвета, смежных I -й вершине. Например, $KV_2 = 4$ означает, что из всех вершин, смежных 2-й вершине, четыре окрашены во 2-й цвет;

P, H - переменные для запоминания номера цвета.

Описание схемы подпрограммы TREE (рис. 11)

В программе TREE реализован алгоритм Прима, который работает следующим образом. Выбирается очередная электрическая цепь схемы (блоки 2, 3, 18), определяется (блок 4) число Q ее контактов (концов), их координаты $XC, YC(Q)$, и формируется матрица $ML(Q, Q)$ длин ребер полного графа (блок 5), построенного на этих контактах.

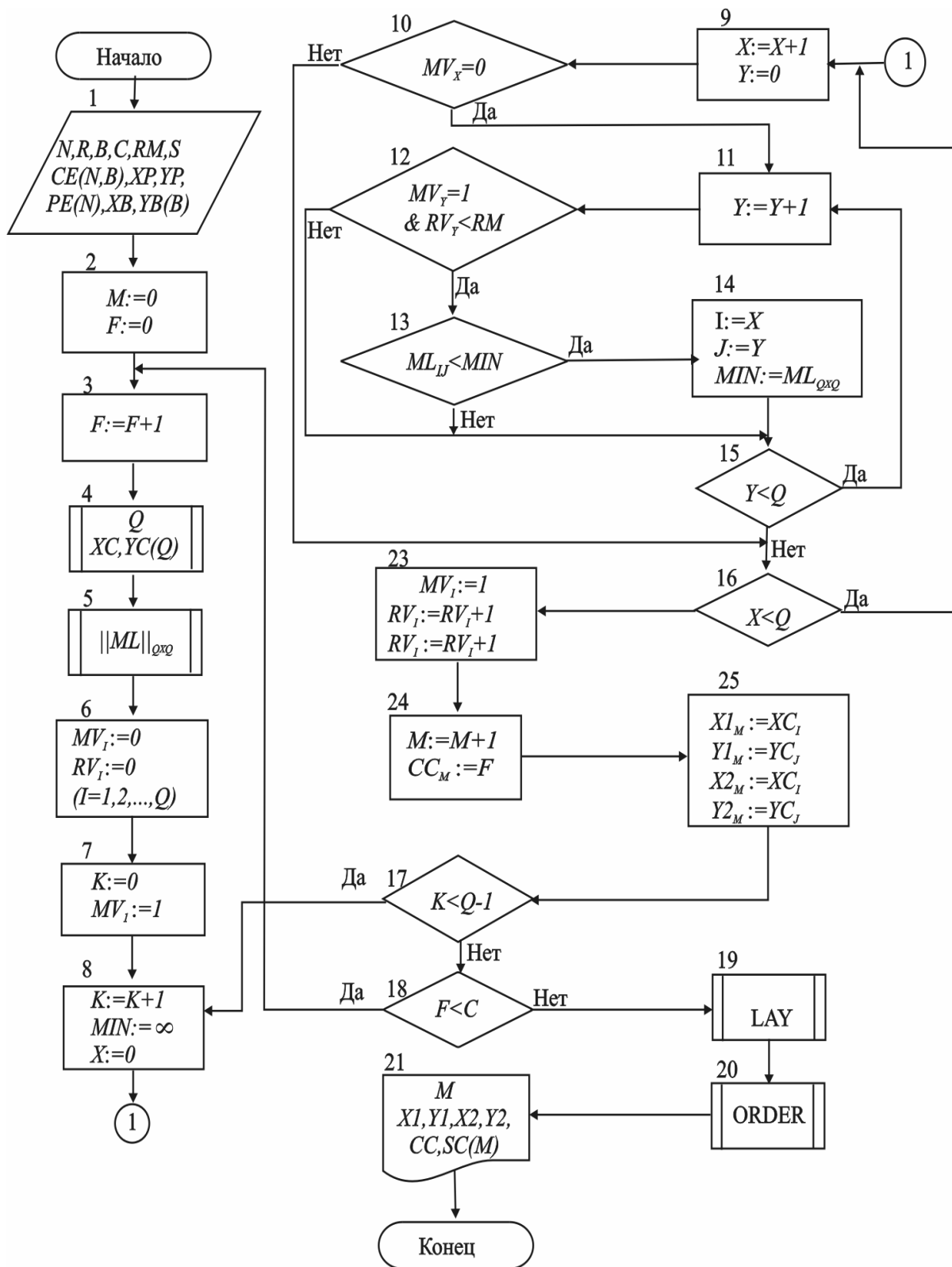


Рис. 11. Схема программного модуля TLO-3

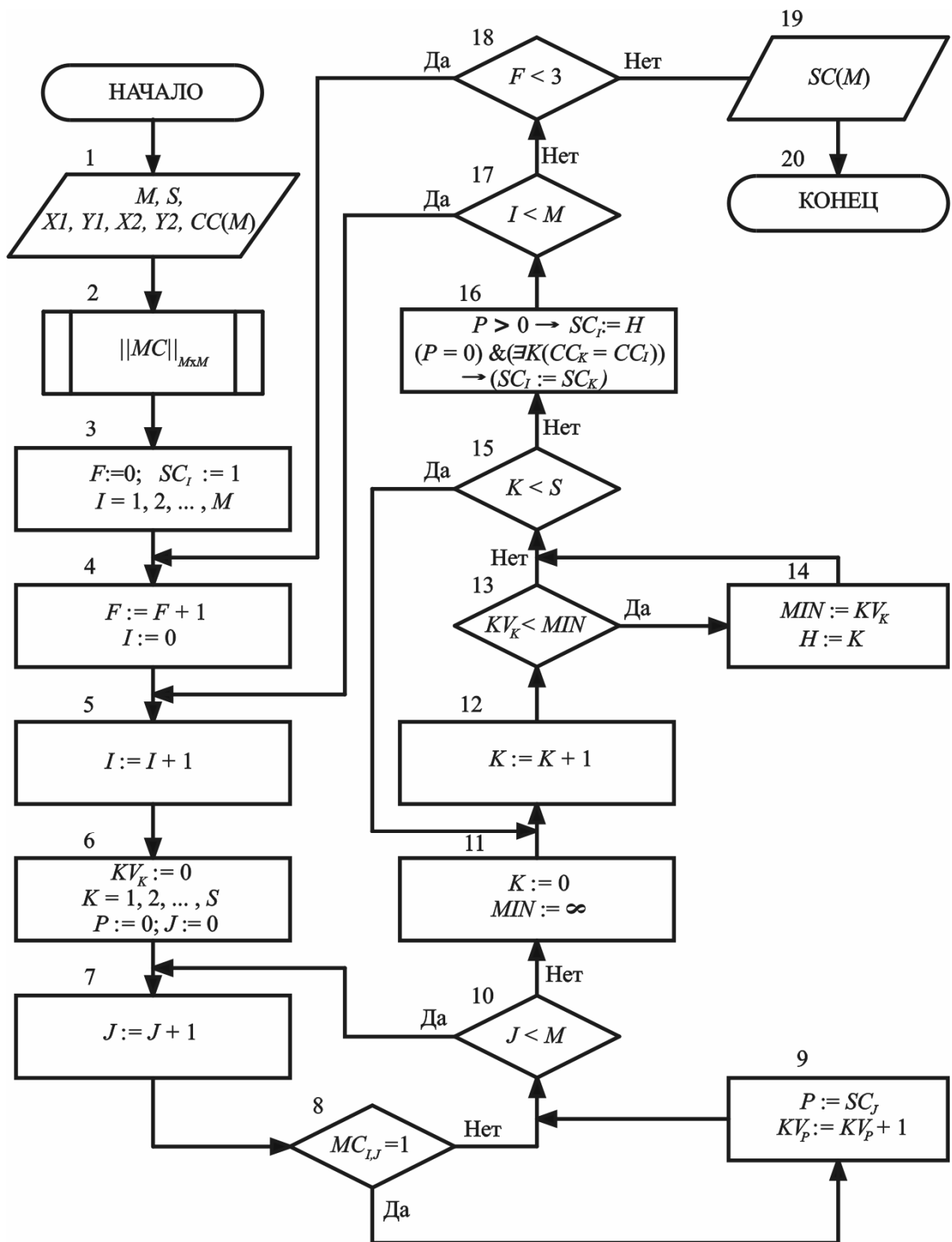


Рис. 12. Схема подпрограммы LAY-3

Первоначально (блок 6) все метки и локальные степени вершин принимают нулевые значения. Затем одна из вершин (здесь первая) включается (блок 7) в дерево. Далее фрагмент дерева разрастается. Выбирается ближайшая к фрагменту I -я вершина (блоки 8...16). Затем I -я вершина включается (блок 23) в дерево, а соответствующее соединение пополняет (блоки 24, 25) список соединений. Процесс продолжается до тех пор, пока не будет выбрано ровно $(Q-1)$ ребер для очередной цепи (блоки 7, 8, 17).

Описание подпрограммы LAУ (рис. 12)

Блок 1 предназначен для формирования матрицы смежности $\|MC_{IJ}\|_{M \times M}$ графа конфликтов соединений. Предварительно все вершины окрашиваются одинаково (блок 3). Далее выбирается очередная I -я вершина (блоки 5, 17), отыскиваются смежные с ней J -е вершины (блоки 7, 8, 10) и среди них подсчитывается число KV_p вершин каждого (P -го) цвета (блок 9). Затем определяется цвет H , в который окрашено минимальное число вершин, смежных с I -й (блоки 11, ..., 15), и в этот цвет окрашивается I -я вершина (блок 16). В случае если I -я вершина изолирована ($P=0$), т.е. соединение не конфликтует с другими, то его помещают (блок 16) в тот слой, где находятся другие соединения той же цепи. Перекраска вершин выполняется заданное число раз (блоки 4, 18).

Контрольная задача Test3-4

Входные данные

Входными данными программы TLO-3 являются 5 файлов проекта Test3x4, обработанные программой PLACE-3 (см. раздел 3).

Выходные данные (упорядоченный список соединений).

49	}	M
27 21 28 21 6 1	}	$X1, Y1, X2, Y2, CC, SC (49)$ (Файл BDP.SO)
28 13 29 13 7 1		
11 18 10 13 17 1		
.....		
9 4 28 29 3 2		
8 4 29 29 5 2		

Результаты представлены на рис. 13. Штриховыми линиями показаны соединения, помещенные во 2-й слой.

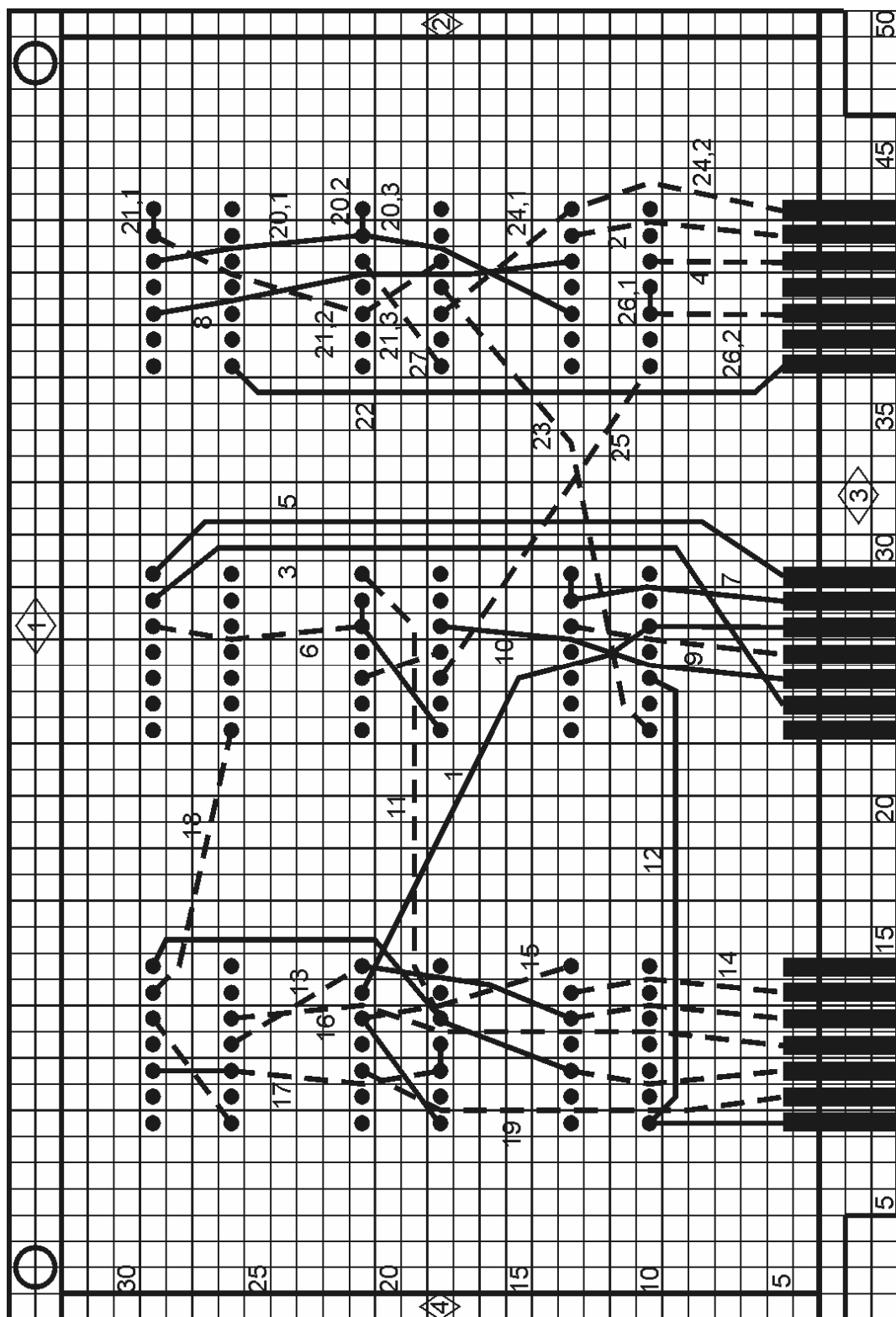


Рис. 13. Крагчайшие соединения

Раскрашенный граф конфликтов (для правого вертикального ряда элементов) представлен на рис. 14.

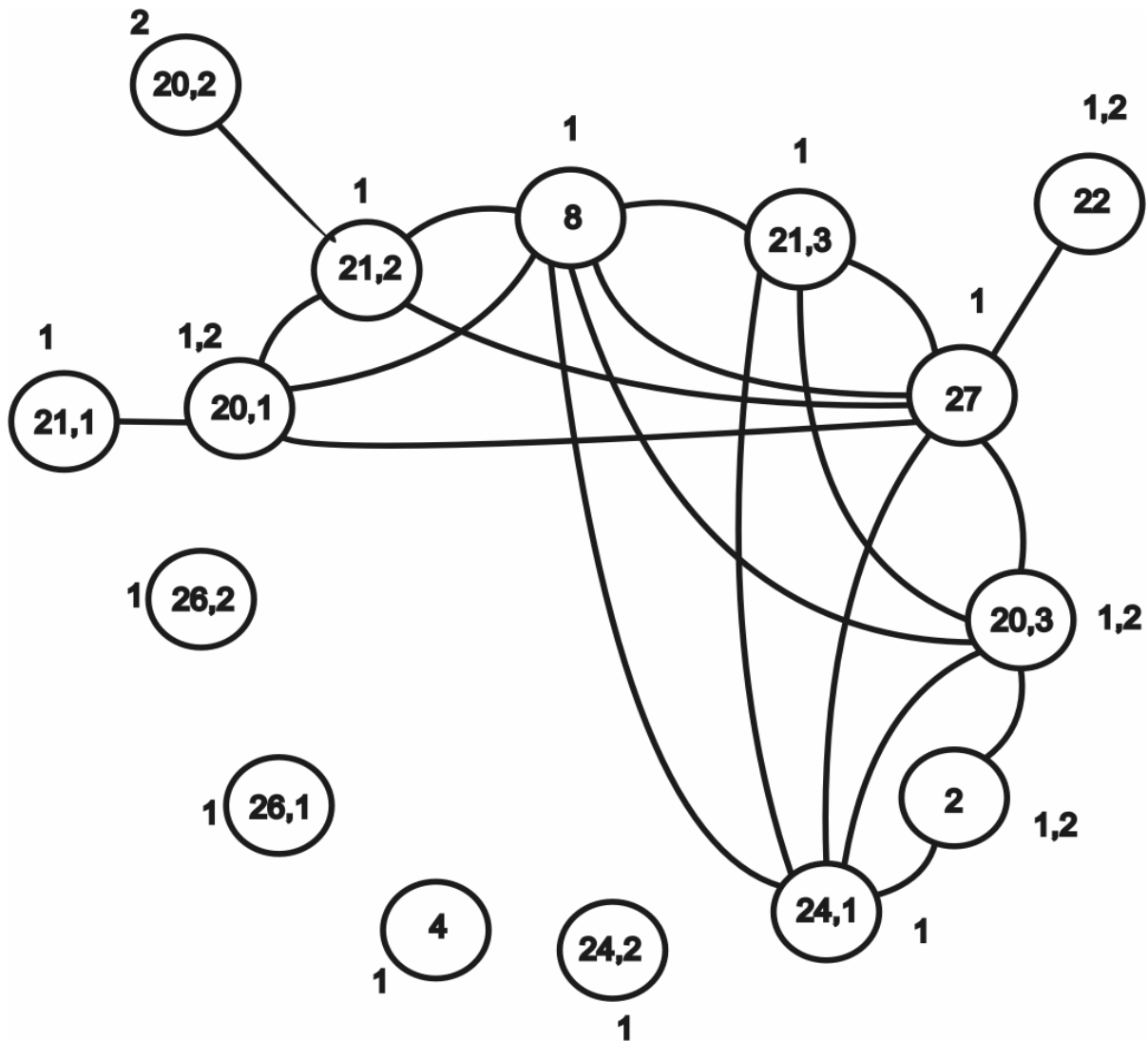


Рис.14. Граф конфликтов

5.5. Задания на лабораторную работу и УИРС

1. При теоретической подготовки использовать данный практикум, контрольные вопросы и цитированную литературу. При подготовке уяснить содержание и сущность каждой из 4 подзадач трассировки соединений. Изучить алгоритмы построения кратчайших соединений, методы раскраски графов (Магу и эвристический), алгоритмы упорядочения соединений. Вычертить схемы подпрограмм TREE и LAY и разобраться в их рабо-

те.

2. Решить задачу с помощью программы TLO-3. В качестве исходных данных использовать исходное описание проекта (см. раздел 3), обработанное программой PLACE-3.

3. Изобразить автоматизированное проектное решение аналогично рис.13. Проанализировать решение.

Изобразить граф конфликтов (для фрагмента проекта, например для левого вертикального ряда элементов на плате) и выполнить раскраску (аналогично рис.14).

4. Задания УИРС.

1) Исследовать временную сложность $O(N)$ используемых алгоритмов.

2) Сравнить результаты работы программ расслоения монтажа, использующие разные способы определения конфликтов (метод отрезков и метод прямоугольников).

5. Отчет должен содержать:

- краткие теоретические сведения, которые можно использовать при защите лабораторной работы;
- схемы программы TLO-3 и подпрограммы LAY;
- входные данные и распечатку результатов машинного решения (список соединений);
- полученное проектное решение в виде рисунка, аналогичному рис.13;
- раскрашенный граф конфликтов;
- анализ полученных результатов.

КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Исчисление высказываний. Предикаты. Математическая логика.
2. Алгебра логики. Законы алгебры логики. Функциональная полнота. Нормальные формы.
3. Методы минимизации булевых функций.
4. Числа графов. Метод Магу.
5. Четыре основных подзадачи трассировки соединений и последовательность их решения. Возможность объединения некоторых подзадач.
6. Построение кратчайших соединений: содержательная формулировка задачи, входные и выходные данные, математические модели объектов проектирования, формализованная формулировка и алгоритмы решения.
7. Типы деревьев. Алгоритм построения дерева Штейнера в ортогональной метрике.
8. Расслоение монтажа. Четыре типа объектов расслоения (элементов монтажа). Содержательная формулировка задачи, входные и выходные данные, математические модели объектов проектирования, формализованная формулировка и алгоритмы решения.
9. Упорядочение соединений: содержательная формулировка задачи, входные и выходные данные, математические модели объектов проектирования, формализованная формулировка и алгоритмы решения.
10. Поясните работу каждого блока в программе TLO-3 и в подпрограмме LAY.

6. ПРОКЛАДКА ТРАСС ЭЛЕКТРИЧЕСКИХ СОЕДИНЕНИЙ

Лабораторная работа № 4

При трассировке электрических соединений прокладываются трассы соединений в каждом слое монтажного пространства, т.е. определяются точное расположение и форма проводников.

6.1. Описание проектной задачи прокладки трасс

Входные данные

1. Упорядоченный список соединений с известной принадлежностью каждого соединения определенному слою платы.
2. Конструктивно-технологическое описание монтажного пространства платы (см. раздел 3.).
3. Описание посадочных мест под конструктивные элементы.
4. Могут быть заданы преимущественные направления трасс в слоях платы.
5. Схемотехнические требования, например ограничение на длину отдельных трасс, на взаимное расположение некоторых трасс и т.п.

Выходные данные

1. Послойные чертежи платы.
2. Управляющая информация (файлы) для программно-управляемого технологического оборудования: фотоплоттеры, сверлильные станки, контролирующие автоматы.

Критерии качества

1. Минимум длины трассы.
2. Наиболее простая конфигурация трассы.
3. Минимум числа пересечений трасс (случай, когда расслоение монтажа двусторонней печатной платы выполняется после прокладки трасс).

Математические модели

Применяемая для трассировки математическая модель монтажного пространства зависит от алгоритма трассировки [2]:

1. Регулярный граф степени 4 с ребрами единичной длины или дискретное рабочее поле (ДРП) – для волнового алгоритма.
2. Канальная модель – для канального алгоритма.

3. Аналоговое рабочее поле – для алгоритма гибкой трассировки и т.д.

Формализованная формулировка задачи

(применительно к волновому алгоритму)

Найти путь кратчайшей длины между 2-мя вершинами в регулярном графе степени 4 с ребрами единичной длины.

Тип оптимизационной задачи

Это задача динамического программирования (применительно к волновому алгоритму).

Алгоритмы трассировки

1. Волновой алгоритм и его модификации.
2. Алгоритмы трассировки на канальных моделях монтажного пространства.
3. Алгоритмы трассировки по магистралям.
4. Топологические алгоритмы трассировки (алгоритмы, основанные на теории планарных графов и алгоритмы гибкой трассировки (на аналоговом рабочем поле)).

6.2. Описание программы TRACE-3

В лабораторной работе исследуется волновой алгоритм трассировки. Схема программы трассировки TRACE-3 представлена на рис. 15, подпрограммы WAVE - на рис. 16.

Входные данные

Входными данными являются 5 файлов исходного описания проекта, обработанные программой PLACE-3 и шестой файл BDP.SO (список соединений), созданный программой TLO-3.

Выходные данные

Выходным является файл BDP.LIN, содержащий описание дискретного рабочего поля: это массив $DP(DX, DY)$, изображения слоев печатной платы, сформированные на основе массива $DP(DX, DY)$, а также файл BDP.REP, содержащий список не проведенных соединений.

Промежуточные данные

D - номер очередного слоя платы;

Q - номер очередного соединения;

A - номер очередного фронта волны;

I - номер дискрета фронта волны;
J - номер дискрета ДРП соседнего с *I*-м дискретом;
L - номер дискрета в списках *XL*, *YL* (*L*);
F - количество дискрет в списках *XF*, *YF* (*F*);
XL, *YL* (*L*) - список координат дискрет *A*-го фронта;
XF, *YF* (*F*) - список координат дискрет (*A*+1)-го фронта;
TX, *TY* (4) - вспомогательные массивы для организации осмотра дискретов, соседних с *I*-м дискретом;
X, *Y* (*G*, *H*) - координаты рассматриваемого дискрета.

Описание схемы программы TRACE-3 (рис.15)

Выбирается очередной слой платы (блоки 2, 3, 11) и формируется исходное ДРП (блок 4). При этом в соответствующие дискреты ДРП заносятся коды запрещенных областей, контактных площадок, незанятых дискретов. Затем поочередно для каждого соединения (блоки 5, 6, 9) проверяется принадлежность его выбранному слою (блок 7), и прокладывается трасса соединения (блок 8). После окончания трассировки слоя его эскиз формируется и выводится на экран видеотерминала (блок 10).

Описание схемы подпрограммы WAVE (рис.16)

Подпрограмма WAVE работает в 2 стадии: распространение числовой волны (блоки 1...14) и проведение трассы (блоки 15...22). Вначале формируется фронт волны, состоящий из всех дискретов контактной площадки - источника волны (блок 2). Затем начинает формироваться очередной *A*+1 фронт волны (блок 4). Числа 1, 2... выбраны в качестве кодов (номеров цепей) проведенных трасс, поэтому для кодирования фронтов волны применяются числа 1001, 1002, 1003... Затем начинает формироваться очередной *A*+1 фронт волны (блок 4). Выбирается очередной *I*-й дискрет предыдущего *A*-го фронта (блоки 5, 11), осматриваются поочередно 4 дискрета, соседних с *I*-м (блоки 6, 10), отыскивается свободный дискрет (блок 7). Если таковой найден, то в него записывается номер *A*+1 фронта, а в списки *XL*, *YL*(*L*) заносятся координаты этого дискрета. Если дискрет занят, то проверяется (блок 8), не достигла ли волна конечной контактной площадки или ранее протрассированного соединения "своей" цепи:

$$DP_{X,Y} = CC_Q \vee (X \geq X2_Q \& X < X2_Q + BX \& Y \geq Y2_Q \& Y < Y2_Q + BY) \quad (2)$$

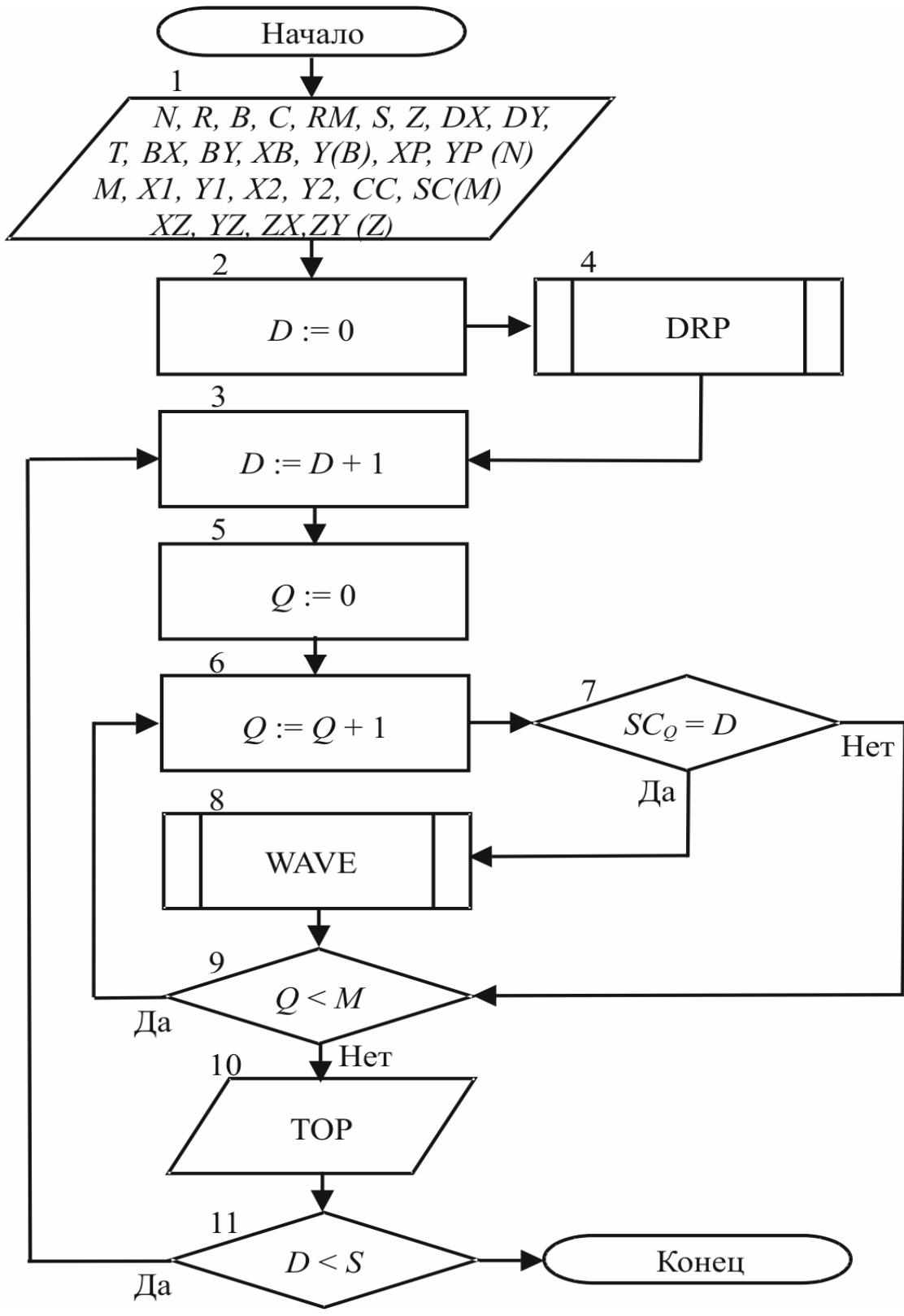


Рис. 15. Схема программы TRACE-3

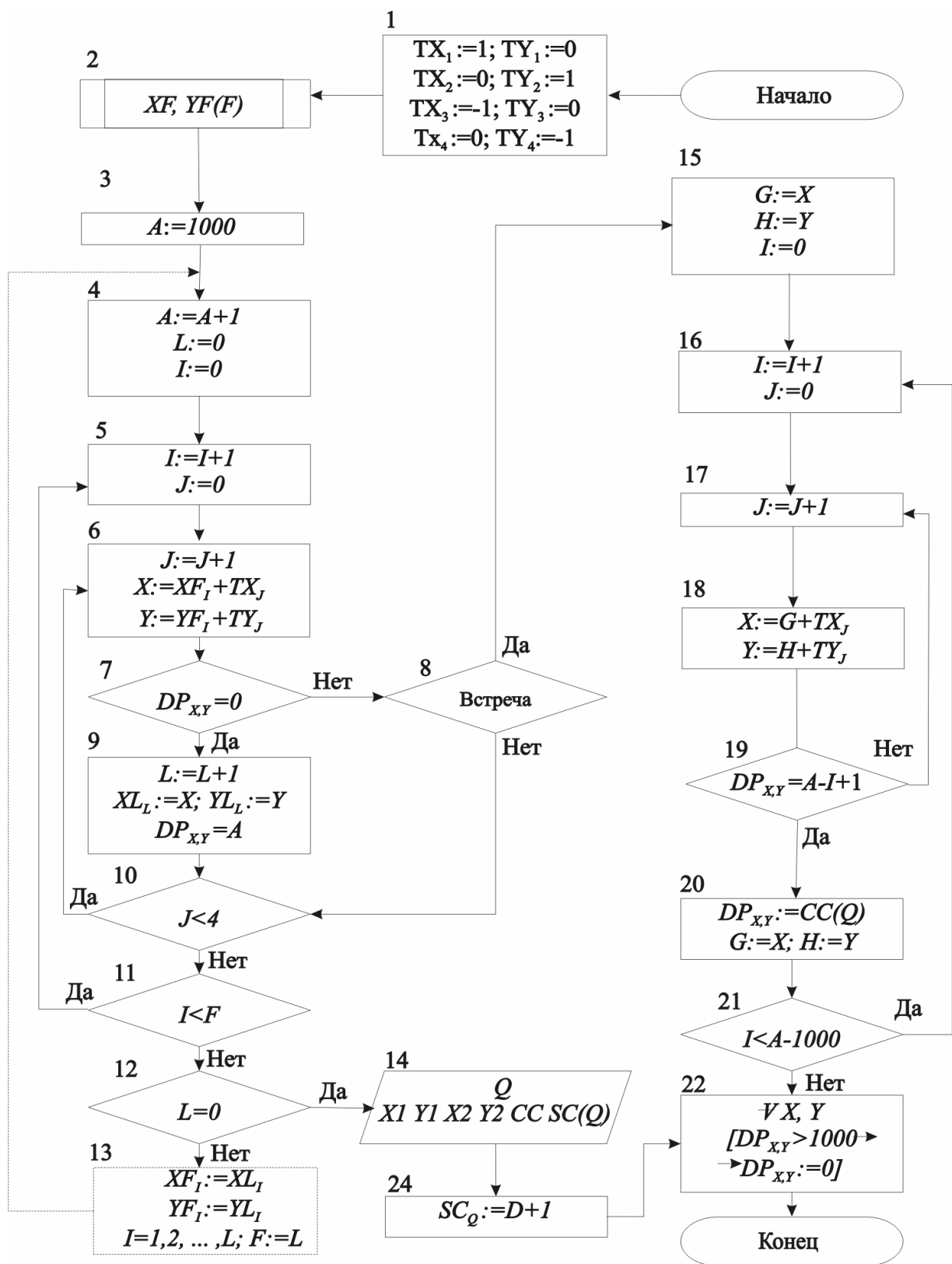


Рис.16. Схема подпрограммы WAVE

Если нет, то продолжается осмотр дискретов ДРП до тех пор, пока не будут просмотрены все L дискрет (блок 11) A -го фронта. Затем проверяется (блок 12), удалось ли сформировать $(A+1)$ -й фронт. Если удалось, то теперь $(A+1)$ -й фронт становится (блок 13) опорным, и процедура повторяется. Если нет, то трассу провести не удалось, и выводится сообщение (блок 14) о номере не проведенного соединения и координатах его концов.

Если логическое высказывание (2) оказалось истинным, то управление передается блоку 15 и начинается стадия проведения трассы. На этой стадии последовательно отыскивается новый дискрет (с координатами G и H), имеющий "вес", на единицу меньше "веса" последнего из найденных дискретов трассы, путем осмотра соседних дискретов (блоки 16, 17, 18). Когда такой дискрет найден (блок 19), в него записывается номер цепи и фиксируются его координаты (блок 20) Проведение трассы завершается "стиранием" остатков числовой волны (блок 22).

Контрольная задача (проект Test3-4)

Входные данные для программы TRACE-3 – 6 файлов описания проекта Test3-4 (см. разделы 3 и 4).

Выходным данные – файл BDP.LIN, содержащий описание ДРП: массив $DP(DX, DY)$. На основе кодов ДРП программа сформировала топологию слоев печатной платы, которая представлена на рис. 17 и 18.

В файле BDP.REP, содержится 6 не проведенных соединений:

N	$X1$	$Y1$	$X2$	$Y2$	CC	SC
30	12	21	14	13	15	1
40	9	4	10	21	19	1
42	38	10	25	18	25	1
44	11	4	12	26	16	1
48	23	10	41	18	23	1
30	12	21	14	13	15	2
38	25	10	8	10	12	2
40	9	4	10	21	19	2
43	38	4	38	26	22	2
45	27	10	13	21	1	2
48	23	10	41	18	23	2

Два соединения (42-е и 44-е), которые не удалось провести в 1-м слое, были проведены во 2-м слое. Не проведенные соединения показаны тонкими линиями на рис. 17 и 18.

6.3. Задания на лабораторную работу и УИРС

1. При теоретической подготовке использовать данные методические указания, контрольные вопросы и цитированную литературу. Вычертить схемы программы TRACE-3 и подпрограммы WAVE и разобраться в работе алгоритма. Выполнить трассировку платы вручную.

2. Подготовить входные данные (использовать результаты работы № 3) и решить задачу с помощью программы TRACE-3. Оценить качество трассировки, выполненной САПР. Сравнить результаты с ручными. Попытаться проложить неразведенные соединения. Указать более оптимальные варианты трассировки и способы, которыми они могут быть получены. Проанализировать, как изменится трассировка при изменении порядка проведения соединений, при другом варианте распределения соединений по слоям.

3. Задания УИРС:

- 1) исследовать временную сложность $O(M)$ алгоритма трассировки;
- 2) исследовать влияние очередности выбора соединений на результаты трассировки;
- 3) исследовать влияние расслоения соединений на результаты трассировки.

4. Отчет должен содержать:

- краткие теоретические сведения;
- схемы программы TRACE-3 и подпрограммы WAVE;
- эскиз топологии, полученный вручную (совмещенная топология);
- входные данные для программы TRACE-3;
- листинг автоматизированного решения (изображение слоев печатной платы
 - список не проведенных соединений. Не проведенные соединения нарисовать на распечатках слоев прямыми линиями;
 - анализ полученных результатов.

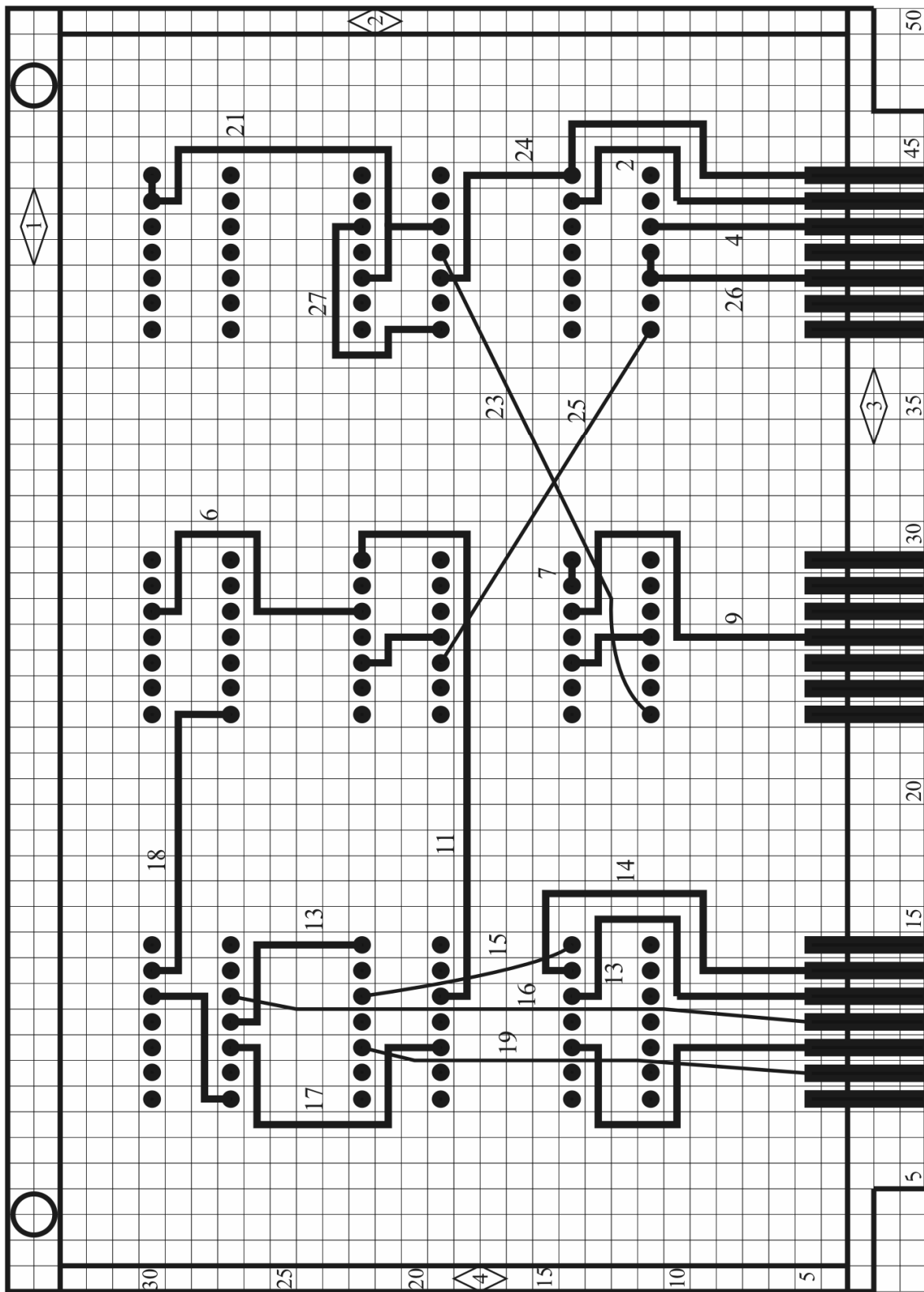


Рис.17 Прокладка трасс в 1-м слое

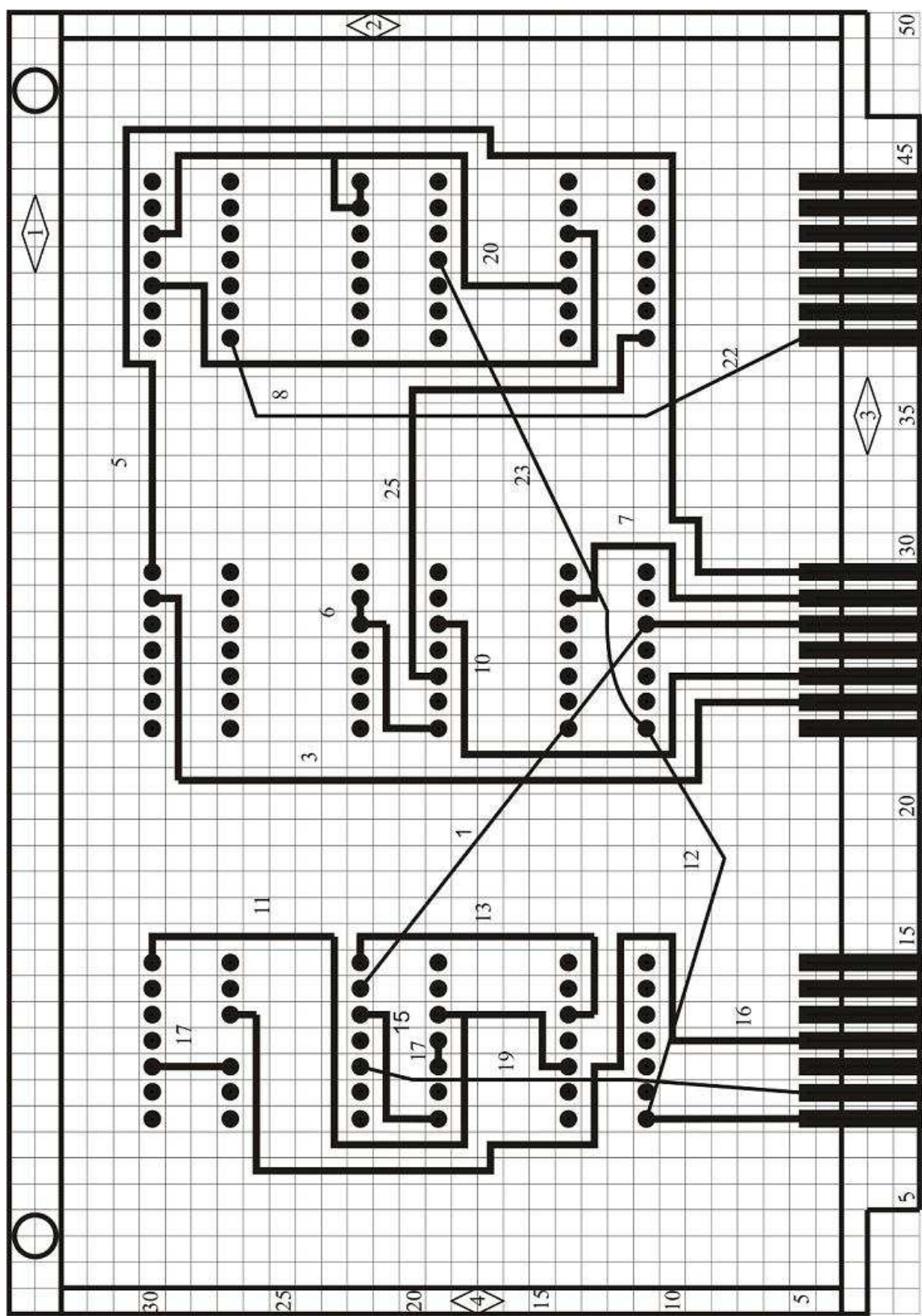


Рис. 18 Прокладка трасс в 2-м слое

Контрольные вопросы

1. Планарные графы. Критерий планарности Понтрягина-Куратовского. Алгоритм Аусландера-Портера. Алгоритм Демукрона.

2. Прокладка трасс в монтажном пространстве: содержательная формулировка задачи, входные и выходные данные, математические модели объектов проектирования, формализованная формулировка и алгоритмы решения.

3. Опишите способы сокращения объема памяти ЭВМ при трассировке с помощью волнового алгоритма.

4. Опишите способы сокращения затрат машинного времени в волновом алгоритме.

5. Особенности автоматизированного конструкторского проектирования узлов ЭС с различным типом монтажа: МПП, ДПП, ОПП, проводной монтаж.

6. Динамическое программирование: формулировка задачи и методы решения.

7. Поясните работу каждого блока в схеме программы TRACE-3 и подпрограммы WAVE.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Курейчик В. М. Математическое обеспечение конструкторского и технологического проектирования с применением САПР: Учебник для вузов. - М.: Радио и связь, 1990. - 352 с.

2. Асланянц В. Р. Модели в автоматизированном конструкторском проектировании РЭА: Методические указания к практическим занятиям. - Владимир, ВПИ, 1987. - 48 с.

3. Автоматизированное проектирование узлов и блоков РЭС средствами современных САПР: Учебное пособие для вузов/ И. Г. Мироненко, В. Ю. Суходольский, К. К. Холуянов.; Под редакцией И. Г. Мироненко. - М.: Высш. шк., 2002. - 391 с.

4. Автоматизация проектирования радиоэлектронных средств: Учебное пособие для вузов. / О. В. Алексеев, А. А. Головков, Н. Ю. Пивоваров и др. ; Под редакцией О. В. Алексеева . - М. : Высш. шк, 2000. - 479 с.

5. Норенков И. П., Маничев В. Б. Основы теории и проектирования САПР: Учебник для вузов. - М.: Высш. шк., 1990. - 335 с.

6. Селютин В. А. Автоматизированное проектирование топологии БИС. - М.: Радио и связь, 1983. - 112 с.

7. Асланянц В. Р. Методы оптимизации в автоматизированном конструировании: Учеб. пособие. - Владимир, ВПИ, 1980. - 106 с.

8. Кофман А. Введение в прикладную комбинаторику. - М.: Мир., 1985. - 480 с.

9. Савельев М. В. Конструкторско-технологическое обеспечение производства ЭВМ: Учеб. пособие для вузов .- М.: Высш. шк.; 2001.- 319 с.

10. Конструкторско-технологическое проектирование электронной аппаратуры: Учебник для вузов. / К. И. Билибин, А. И. Власов, Л. В. Журавлева и др. Под общ. ред. В. А. Шахнова .- М. : Издательство МГТУ им. Н. Э. Баумана, 2002. - 528 с.

11. Бернюков А. С., Никитин А. И. Цифровые устройства: Учеб. пособие / Владим. гос. ун-т. Владимир, 2000. - 176 с.

12. Асланянц В. Р. Оптимизационные задачи на графах/ Метод. указания к практ. занятиям. - Владимир, ВПИ, 1980. - 40 с.

ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ.....	3
1. ДИСКРЕТНАЯ МАТЕМАТИКА	3
2. АРХИТЕКТУРА УЧЕБНОЙ САПР CROCUS-3.....	4
3. ПОКРЫТИЕ ФУНКЦИОНАЛЬНОЙ СХЕМЫ ЭС НАБОРОМ ФТУ И РАЗБИЕНИЕ СХЕМЫ ЭС.....	14
3.1. Описание проектной задачи покрытия электрических функциональных схем ЭС (СхЭФ) набором функционально- типизированных узлов (ФТУ).....	14
3.2. Описание проектной задачи разбиения схем ЭС	16
3.3. Описание программы DECOM-3.....	17
3.4. Описание программы CODER-3	25
3.5. Задания на лабораторную работу и УИРС	26
4. РАЗМЕЩЕНИЕ ЭЛЕМЕНТОВ НА КОММУТАЦИОННОЙ ПЛАТЕ И РАСПРЕДЕЛЕНИЕ ЦЕПЕЙ ПО ВЫВОДАМ УЗЛА.....	28
4.1. Описание проектной задачи размещения элементов на коммутационной плате.....	28
4.2. Описание проектной задачи распределения электрических цепей по выводам конструктивного узла (РЦВУ).....	30
4.3. Описание программы PLACE-3.....	31
4.4. Задания на лабораторную работу и УИРС.....	40
5. ПОСТРОЕНИЕ КРАТЧАЙШИХ СОЕДИНЕНИЙ, РАССЛОЕНИЕ МОНТАЖА И УПОРЯДОЧЕНИЕ СОЕДИНЕНИЙ.....	43
5.1. Описание проектной задачи построения кратчайших соедине- ний.....	43
5.2. Описание проектной задачи расслоения монтажа.....	44
5.3. Описание проектной задачи упорядочения соединений.....	45
5.4. Описание программного модуля TLO-3.....	46
5.5. Задания на лабораторную работу и УИРС.....	52
6. ПРОКЛАДКА ТРАСС ЭЛЕКТРИЧЕСКИХ СОЕДИНЕНИЙ.....	55
6.1. Описание проектной задачи прокладки трасс.....	55
6.2. Описание программы TRACE-3.....	56
6.3. Задания на лабораторную работу и УИРС.....	61
БИБЛИОГРАФИЧЕСКИЙ СПИСОК.....	65