



*Томский межвузовский центр
дистанционного образования*

Е.Ф. Жигалова

АВТОМАТИЗАЦИЯ КОНСТРУКТОРСКОГО И ТЕХНОЛОГИЧЕСКОГО ПРОЕКТИРОВАНИЯ

Учебное пособие

ТОМСК - 2002

Министерство образования Российской Федерации

**ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ СИСТЕМ
УПРАВЛЕНИЯ И РАДИОЭЛЕКТРОНИКИ (ТУСУР)**

**Кафедра компьютерных систем в управлении
и проектировании (КСУП)**

Е.Ф. Жигалова

**АВТОМАТИЗАЦИЯ
КОНСТРУКТОРСКОГО И
ТЕХНОЛОГИЧЕСКОГО
ПРОЕКТИРОВАНИЯ**

Учебное пособие

2002

Корректор: Красовская Е.Н.

Жигалова Е.Ф.

Автоматизация конструкторского и технологического проектирования:
Учебное пособие. – Томск: Томский межвузовский центр дистанционного
образования, 2002. – 78 с.

СОДЕРЖАНИЕ

ВВЕДЕНИЕ.....	4
1 ПРОГРАММА ЛЕКЦИОННОГО КУРСА.....	4
2 МАТЕМАТИЧЕСКИЕ МОДЕЛИ ОБЪЕКТОВ КОНСТРУКТОРСКОГО ПРОЕКТИРОВАНИЯ РАДИОЭЛЕКТРОННОЙ И ЭЛЕКТРОННО- ВЫЧИСЛИТЕЛЬНОЙ АППАРАТУРЫ.....	4
3 СТРУКТУРНЫЙ СИНТЕЗ ТЕХНИЧЕСКОГО ОБЪЕКТА.....	9
3.1 Задача компоновки.....	9
3.2 Задача размещения.....	11
3.3 Задача трассировки.....	12
4 АЛГОРИТМЫ РЕШЕНИЯ ЗАДАЧ СТРУКТУРНОГО СИНТЕЗА ТЕХНИЧЕСКИХ ОБЪЕКТОВ.....	14
4.1 Алгоритмы и модели компоновки.....	14
4.2 Размещение конструктивных элементов в монтажном пространстве (задача размещения).....	26
4.3 Системы автоматизированного проектирования.....	35
5 МЕТОДИЧЕСКИЕ УКАЗАНИЯ ПО ВЫПОЛНЕНИЮ КОНТРОЛЬНОЙ РАБОТЫ ПО САПР РСAD 4.5.....	39
5.1 Запуск программ в САПР РСAD 4.5.....	39
5.2. Способы проектирования и подготовки производства печатных плат.....	40
5.3 Проектирование печатной платы по ее топологии.....	44
6 P-CAD 2000 [ACCEL EDA 15.0].....	53
КОНТРОЛЬНАЯ РАБОТА №1.....	59
КОНТРОЛЬНАЯ РАБОТА №2.....	72
СПИСОК ЛИТЕРАТУРЫ.....	76
ПРИЛОЖЕНИЕ А.....	77
ПРИЛОЖЕНИЕ В.....	78

ВВЕДЕНИЕ

Целью изучения курса «Автоматизация конструкторского и технологического проектирования» является изучение общих принципов разработки автоматизированных систем конструкторского проектирования радиоэлектронной и электронно-вычислительной аппаратуры, методов и алгоритмов анализа, оптимизации и синтеза технических объектов, современных программных средств автоматизированного проектирования.

Для изучения автоматизированных систем проектирования студенты в течение семестра выполняют две контрольные работы по базовым алгоритмам задач структурного синтеза технических объектов и по системам автоматизированного проектирования САПР PCAD 4.5, P_CAD 2000, описание которых приводится в данном методическом пособии.

1 ПРОГРАММА ЛЕКЦИОННОГО КУРСА

1.1 Понятие автоматизированного проектирования.

Определение проектирования. Принципы проектирования. Блочнo-иерархический подход. Аспекты и уровни проектирования. Этапы проектирования радиоэлектронных средств. Итерационное проектирование.

1.2 Понятие о математических моделях (ММ) технических объектов. Типовые проектные процедуры. Анализ и синтез. Типичная последовательность проектных процедур. Место моделирования в проектировании. Этапы проектирования, под-дающиеся формализации.

1.3 Математический аппарат автоматизации конструкторского проектирования.

Методы конструкторского проектирования радиоэлектронной и электронно-вычислительной аппаратуры.

1.4 Системы Автоматизированного проектирования (САПР).

2 МАТЕМАТИЧЕСКИЕ МОДЕЛИ ОБЪЕКТОВ КОНСТРУКТОРСКОГО ПРОЕКТИРОВАНИЯ РАДИОЭЛЕКТРОННОЙ И ЭЛЕКТРОННО-ВЫЧИСЛИТЕЛЬНОЙ АППАРАТУРЫ

При решении задач в АСКП (Автоматизированных системах конструкторского проектирования) возникает необходимость выбора совокупности математических моделей, обеспечивающих необходимой информацией каждый алгоритм решения конкретной задачи.

Для каждой задачи АСКП (компоновка, размещение, трассировка) на этапе предшествующем постановке задачи, выбирается (разрабатывается) математическая модель. Это позволяет перейти от содержательной постановки задачи к формальной постановке в терминах выбранной модели.

Поскольку цели отдельных задач АСКП отличаются, то возможно для решения каждой задачи применять свою модель.

Проблема выбора (построения) моделей является важной и актуальной, так как от модели во многом зависит качество решения задач в АСКП и эффективность алгоритмических методов.

Требования, предъявляемые к математической модели:

- простота перехода от объекта к модели и обратно;
- удобство представления и обработки модели в ЭВМ;
- небольшой объем памяти компьютера, отводимый для хранения информации для модели;
- разработанность математического аппарата для работы с данной моделью (дает возможность разрабатывать высокоэффективные алгоритмы проектирования);
- информационная простота модели (возможность перехода от описания данной модели к более простой).

Математические модели, используемые при решении задач в АСКП можно разделить на две группы:

одна - математические модели схем;

другая - математические модели монтажного пространства.

Рассмотрим математические модели, построенные в рамках аппарата теории графов.

Введем понятия:

- коммутационная схема;
- электрическая цепь;
- контакты.

Коммутационная схема

Любая схема (функциональная, логическая, принципиальная, структурная) состоит из набора элементов и соединений, о которых можно сказать, что они находятся в заданном отношении. Поэтому такие схемы можно рассматривать как некоторое множество элементов $X=(X_1, X_2, \dots, X_n)$, $|X|=n$ и некоторое множество соединителей (называемых цепями) $E=(e_1, e_2, \dots, e_m)$ $|E|=m$, находящихся в заданном бинарном отношении.

В схеме имеются внешние и внутренние выводы: C .

$$C = \{c_{0,1}, c_{0,2}, \dots, c_{0,p}, c_{1,1}, c_{1,2}, \dots, c_{n,1}, c_{n,2}, \dots, c_{n,m}\}.$$

Внутренние выводы соединяют элементы схемы между собой.

Внешние выводы – соединяют элементы данной схемы с элементами другой схемы.

Множество C – будем называть множеством контактов.

Два контакта $C_{i,j}$ и $C_{k,e}$ считаются связанными (соединенными), если они объединяются одной электрической цепью.

Электрическая цепь – это некоторое подмножество контактов C' множества контактов C , т.е. $C' \subseteq C$, принадлежащих к одному эквипотенциалу.

Контакты подмножества C' имеют равный потенциал в принципиальной электрической схеме.

Такое представление схемы называют схемой соединений или коммутационной схемой (КС).

При построении математической модели в коммутационной схеме каждой однопотенциальной цепи ставят в соответствие число из натурального ряда.

Для решения задач компоновки, размещения и трассировки коммутационную схему можно представлять структурными математическими моделями различного вида.

Модели схем в зависимости от математического аппарата в рамках которого разработана модель, условно разделяют на три вида:

- модели, использующие аппарат теории графов;
- модели, использующие аппарат теории гиперграфов;
- модели, использующие аппарат теории множеств.

В настоящее время наиболее широко используются модели, построенные в рамках теории графов и в первую очередь при математической постановке оптимизационных задач конструирования. Это позволяет при разработке алгоритмов решения задач конструирования использовать известные математические методы.

Рассмотрим структурные математические модели, использующие аппарат теории графов на примере условного фрагмента схемы электрической принципиальной, представленной на рисунке 1.

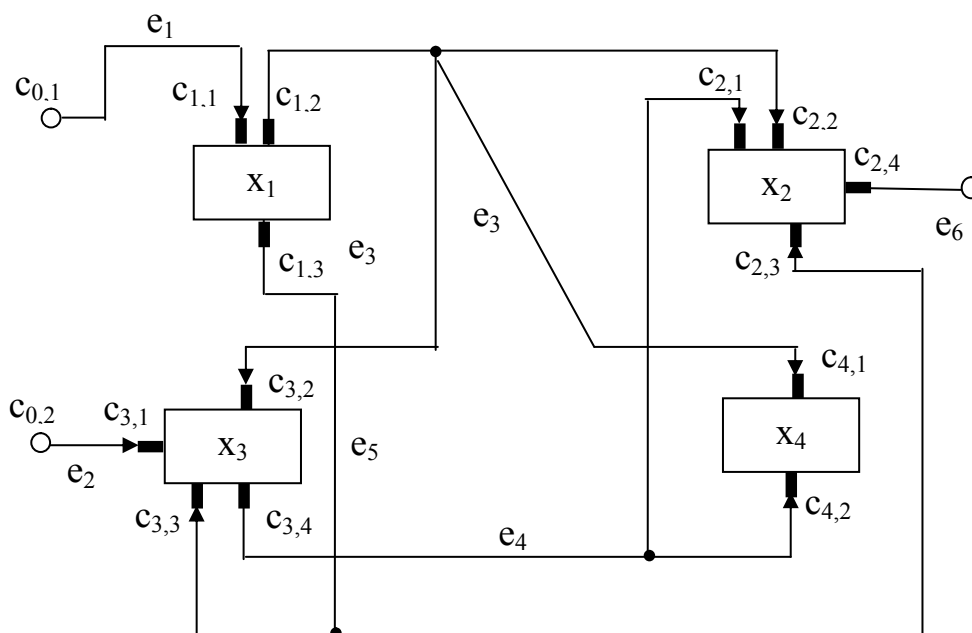


Рисунок 1

Одним из способов представления схемы устройства структурной математической моделью является ее интерпретация неориентированным связным графом $G=(X,U)$, где $X=\{x_{ij}\}$ $i=\overline{1,n}$ множество вершин G , U – множество ребер $U=\{u_{ij}\}$, определяющих отношения между элементами множества X .

Для построения графа G , необходимо задать между элементами множества $X=\{x_{ij}\}$ и элементами схемы (к элементам схемы относятся сами приборы и внешние контакты схемы), взаимоднозначное соответствие, в том числе между элементами множества ребер $U=\{u_{ij}\}$ и элементами множества электрических цепей $E=\{e_{ij}\}$.

Для нашего примера математическая модель М1, интерпретирующая схему (рисунок 1), представляет неориентированный граф $G_I=(X,U)$ с мультичислом равным 3:

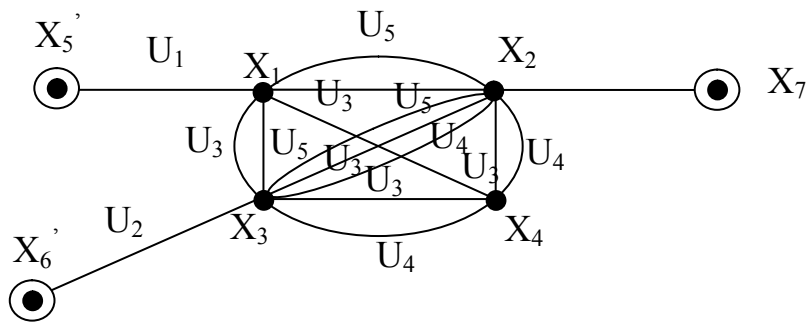


Рисунок 2 – Математической модель М1

Модель М1 удобна для решения задач компоновки и размещения.

Для решения задач определения планарности и трассировки математическая модель М1 неудобна из-за больших искажений между моделью и коммутационной схемой.

Модификацией математической модели М1 является математическая модель М2, в которой полные подграфы, моделирующие цепи $e_i \in E$, заменяются покрывающими их деревьями (рисунок 3).

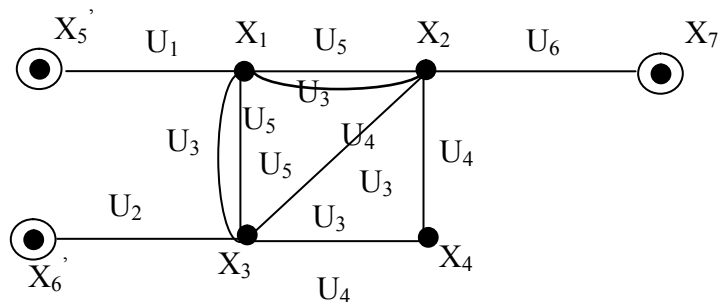


Рисунок 3 – Математической модель М2

Разновидностью математической модели М2 является представление коммутационной схемы графом $G=(X,U)$, в котором все покрывающие деревья являются звездными подграфами с центральной вершиной x_i , соответствующей началу цепи e_i (см. рисунок 4).

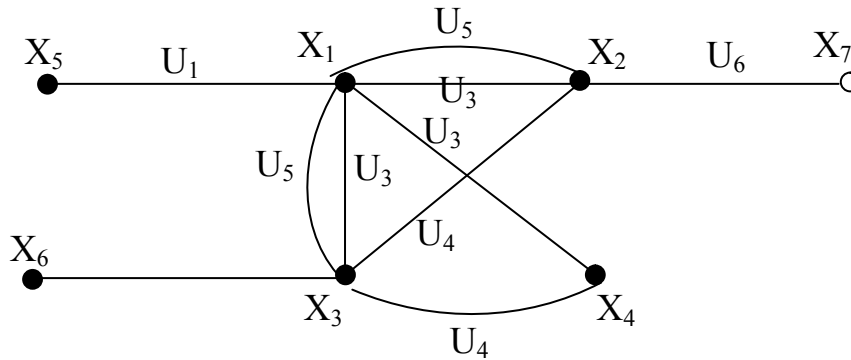


Рисунок 4 – Математическая модель М3

Общий недостаток приведенных математических моделей: *фиксированные деревья могут ухудшить качество компоновки*. Тем не менее приведенные модели М1, М2, М3 наиболее просты при решении практических задач.

Коммутационная схема также может быть представлена в виде двудольного ориентированного графа $D=(X \cup C \cup E, U)$. В этом случае граф задают матрицей инцидентности $I=||i_{k,p}||$, элементы $i_{k,p}$ которой равны либо -1, либо 0, либо 1.

$i_{k,p} = -1$, если цепь e_p – выходная для элемента x_k ; $i_{k,p} = 0$, если цепь e_p не связана с элементом x_k ; $i_{k,p} = 1$, если цепь e_p – входная для элемента x_k .

Для фрагмента коммутационной схемы, представленной на рисунке 1, матрица I запишется:

$$I = \begin{matrix} & e_1 & e_2 & e_3 & e_4 & e_5 & e_6 \\ \begin{matrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{matrix} & \left\| \begin{array}{cccccc} 1 & 0 & -1 & 0 & -1 & 0 \\ 0 & 0 & 1 & 1 & 1 & -1 \\ 0 & 1 & 1 & -1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 \end{array} \right\| \end{matrix}$$

Номера столбцов данной матрицы соответствуют электрическим цепям (e_1, e_2, \dots, e_6), а номера строк соответствуют элементам схемы (x_1, x_2, x_3, x_4). Соответствующая этой интерпретации коммутационной схемы математическая модель М4 представлена на рисунке 5.

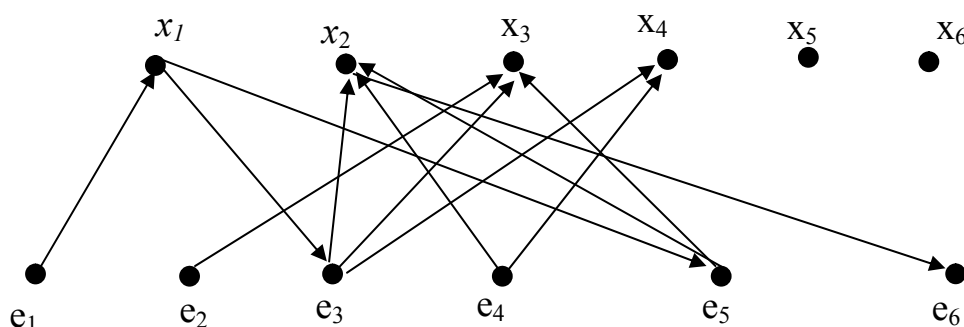


Рисунок 5 – Математическая модель М4

Математическая модель М4 применяется для разработки алгоритмов трассировки соединений и, когда требуется различать входы и выходы логических элементов.

Когда матрицы, представляющие граф, сильно разрежены, т.е. имеют мало ненулевых элементов, хранение информации в матричной форме не эффективно. В этом случае удобной формой представления матриц являются списки соединений.

3 СТРУКТУРНЫЙ СИНТЕЗ ТЕХНИЧЕСКОГО ОБЪЕКТА

Задача синтеза технического объекта включает в себя создание структуры проектируемого объекта и расчет его параметров. Эти две части синтеза соответственно называются структурным и параметрическим синтезом. Задача структурного синтеза заключается в поиске оптимальной или рациональной структуры (схемы) технического объекта для реализации заданных функций в рамках выбранного принципа действия.

Сущность получения математических моделей объектов проектирования электронно-вычислительной и радиоэлектронной аппаратуры, для решения задач структурного синтеза, рассмотрим на примере компоновки, размещения, трассировки.

3.1 Задача компоновки

Под задачами компоновки понимают задачи разбиения множества $D=(d_1, d_2, \dots, d_n)$ из n элементов на ряд непересекающихся подмножеств D_k , $k = \overline{1, N}$, чтобы при этом выполнялись заданные ограничения и достигался экстремум некоторой функции качества $F(x)$.

При заданном числе N подмножеств разбиения задача компоновки формулируется следующим образом:

$$F(x) \rightarrow \min \quad (1)$$

и для $\forall k, l \in \{1, 2, \dots, N\}$ выполняется:

$$D_k \cap D_l = \emptyset; \quad (2)$$

$$\bigcup_{k=1}^N D_k = D, \quad (3)$$

где $D_k \subset D$ – множество элементов, принадлежащих k -му подмножеству разбиения при условии, что мощность $|D_k|$ каждого подмножества из разбиения задана, т.е.

$$|D_k| = n_k; \quad \sum_{k=1}^N n_k = n. \quad (4)$$

Просмотреть все варианты разбиения уже для числа $n \geq 100$ нереально!

Если же сформулировать задачу компоновки в терминах целочисленного программирования, то можно уменьшить число просматриваемых вариантов компоновки.

Итак, пусть требуется распределить n компонентов электронной схемы между N блоками таким образом, чтобы суммарное число связей между блоками было минимально.

Введем вектор $X = \{x_{i,k}\}$ переменных проектирования, где $x_{i,k}$ – элементы вектора X , $i = \overline{1, n}$; $k = \overline{1, N}$;

$$x_{i,k} = \begin{cases} 1, & \text{если компонент } d_i \text{ включается в подмножество } D_k; \\ 0, & \text{в противном случае.} \end{cases}$$

Пусть функция качества $F(x)$ характеризует общее число связей между подмножествами

$$\{D_k\} \text{ для } k = \overline{1, N}.$$

Тогда, задача компоновки запишется:

$$F(x) = \sum_{j=1}^{n-1} \sum_{i=j+1}^n \sum_{k=1}^{N-1} \sum_{l=k+1}^N x_{j,k} \cdot x_{i,l} \cdot \pi_{i,j} \rightarrow \min, \quad (5)$$

при условиях:

$$\sum_{k=1}^N x_{i,k} = 1, i = \overline{1, n}; \quad (6)$$

$$\sum_{i=1}^n x_{i,k} = n_k, k = \overline{1, N}; \quad (7)$$

$$\sum_{i=1}^n V_i^{(S)} \cdot x_{i,k} \leq V_S^{(k)}; \quad (8)$$

$$x_{i,k} = (0;1), \quad (9)$$

где π_{ij} - число связей между компонентами d_i и d_j ;

$V_i^{(S)}$ - значение параметра S для компонента d_i ;

$V_S^{(k)}$ - ограничение по параметру S , накладываемое на подмножество D_k ;

S - может быть любой параметр, подчиняющийся свойству аддитивности: объем; масса; энергоемкость, стоимость и т.п.

В задаче (5-9) условия (6, 7) соответственно означают, что каждый компонент может быть отнесен только к одному из подмножеств D_k и в каждом подмножестве D_k может содержаться компонентов не более, чем заданное число n_k .

3.2 Задача размещения

Высокая плотность размещения элементов ЭВА создает большие трудности при реализации соединений между ними. В этой связи задача размещения элементов на плоскости определяет быстроту и качество трассировки. Оптимальное размещение элементов обеспечивает повышение надежности проектируемого устройства, минимизацию наводок, задержек сигналов, уменьшение общей длины соединений и т.п.

Формально задача размещения заключается в определении оптимального варианта расположения элементов на плоскости в соответствии с введенным критерием. Например, с минимальной взвешенной длиной соединений.

В общем случае требуется найти размещение компонентов d_1, \dots, d_n на множестве q_1, q_2, \dots, q_m ($m \leq n$) позиций монтажного пространства, при котором суммарная длина соединений между компонентами была бы минимальной.

Введем булевы переменные:

$$x_{i,k} = \begin{cases} 1, & \text{если компонент } d_i \text{ назначается на позицию } q_k; \\ 0, & \text{в противном случае.} \end{cases}$$

Тогда математическая модель задачи размещения может быть записана:

$$F(x) = \sum_{i=1}^{n-1} \sum_{j=i+1}^n \sum_{k=1}^m \sum_{s=1}^m x_{i,k} \cdot x_{j,s} \cdot l_{k,s} \cdot p_{i,j} \rightarrow \min \quad (10)$$

при условиях:

$$\sum_{k=1}^m x_{i,k} = 1, \quad i = \overline{1, n}; \quad (11)$$

$$\sum_{i=1}^n x_{i,k} \leq 1, \quad k = \overline{1, m} \quad (12)$$

$$x_{i,k} \in \{0, 1\}, \quad (13)$$

где $l_{k,s}$ — расстояние между позициями q_k и q_s ; $p_{i,j}$ — число связей между компонентами d_i ; d_j .

Условия (11, 12) соответственно означают, что каждый компонент может быть размещен только на одно посадочное место и каждое посадочное место может быть закреплено только за одним компонентом.

3.3 Задача трассировки

Задача трассировки встречается при конструировании печатных плат; разработке систем водоснабжения, электроснабжения и т.д.

Трассировка соединений является, как правило, заключительным этапом конструкторского проектирования ЭВА и состоит в определении линий, соединяющих эквипотенциальные контакты элементов и компонентов, составляющих проектируемое устройство.

Задача трассировки — одна из наиболее трудоемких в общей проблеме автоматизации проектирования ЭВА. С математической точки зрения трассировка — наисложнейшая задача выбора из огромного числа вариантов оптимального решения.

Основная задача трассировки формулируется следующим образом: по заданной схеме соединений проложить необходимые проводники на плоскости (плате, типовом элементе замены, кристалле и т.п.), чтобы реализовать заданные электрические соединения с учетом заранее заданных ограничений. Основными являются ограничения

На ширину проводников и минимальное расстояние между ними.

Исходной информацией для решения задачи трассировки соединений обычно являются список цепей, параметры конструкции элементов и коммутационного поля, а также данные по размещению элементов.

Критериями трассировки, наиболее часто используемые для оценки качества решения задачи трассировки, могут быть: процент реализованных

соединений, суммарная длина проводников, число монтажных слоев, число межслойных переходов, минимальная область трассировки и другие.

Задача трассировки всегда имеет топологический и метрический аспекты. Топологический аспект связан с выбором допустимого пространства расположения отдельных фрагментов соединений без фиксации их конкретного месторасположения при ограничениях на число пересечений и слоев. Метрический аспект предполагает учет конструктивных размеров элементов, соединений и коммутационного поля, а также метрических ограничений на трассировку.

Рассмотрим одну разновидность задачи трассировки - задачу построения связывающих сетей минимальной длины для цепей α_k .

Пусть U_k – множество точек, соединяемых по электрической цепи α_k ; $|U_k| = n_k$, где каждому элементу U_k соответствует одна точка в монтажном пространстве.

Введем понятие трассы.

Трасса – множество связанных отрезков, соединяющих точки электрической цепи.

Определим переменную проектирования x_{ij} .

$$x_{ij} = \begin{cases} 1, & \text{если ребро } (i, j), \text{ длиной } l \text{ включается в связывающую сеть;} \\ 0, & \text{в противном случае,} \end{cases}$$

где x_{ij} - булева переменная.

Тогда математическая модель задачи трассировки запишется:

$$F(x) = \sum_{i=1}^{n_{k-1}} \sum_{j=i+1}^{n_k} l_{i,j} \cdot x_{i,j} \rightarrow \min \quad (14)$$

при условиях:

$$\sum_{j=1}^{n_k} x_{ij} \leq K_0, \quad i = \overline{1, n_k}; \quad (15)$$

$$x_{i,j} = (0 ; 1), \quad (16)$$

где, K_0 – максимально допустимое число соединений в одной точке.

Условие (15) означает, что в одной точке не могут соединяться количество ребер более заданного числа K_0 .

Для контроля связности сети при решении задачи трассировки, математическая модель (17,18) может быть дополнена условиями:

$$y_{i,j}^{(S)} = \sum_{r=1}^{n_k} y_{i,r}^{(S-1)} \cdot x_{r,j}; \quad y_{i,j}^{(1)} = x_{i,j}; \quad (17)$$

$$\sum_{S=2}^{n_k-1} y_{i,j}^{(S)} \geq 1, \quad i = \overline{1, n_k}; \quad j = \overline{2, n_k}, \quad (18)$$

где $y_{i,j}^{(S)}$ – вспомогательные переменные.

Суть ограничений (17,18) в том, что на каждом шаге принятия решения «включать – не включать ребро в трассу» должны рассматриваться точки соединений, принадлежащие одной цепи.

4 АЛГОРИТМЫ РЕШЕНИЯ ЗАДАЧ СТРУКТУРНОГО СИНТЕЗА ТЕХНИЧЕСКИХ ОБЪЕКТОВ

4.1 Алгоритмы и модели компоновки

В настоящее время при разработке современной электронно-вычислительной аппаратуры (ЭВА) основными элементами являются интегральные микросхемы (ИМС). К ИМС относятся микроэлектронные изделия, выполняющие функции преобразования, обработки сигнала или накопления информации и имеющие высокую плотность упаковки электрически соединенных элементов и кристаллов, которые рассматриваются как единое целое. Под элементом ИМС понимается его часть, реализующая функцию произвольного электрорадиоэлемента (ЭРЭ), не выделяемая как самостоятельное изделие. Компонент ИМС также реализует функцию ЭРЭ, но рассматривается как самостоятельное изделие. Сложность ИМС характеризуют числом содержащихся в них элементов и компонентов и называют степенью интеграции (СИ). Величину СИ определяют по формуле $СИ = l_g N$, где N – число элементов и компонентов в ИМС.

При решении задач конструкторского проектирования радиоэлектронной и электронно-вычислительной аппаратуры с помощью САПР должно быть предусмотрено использование унифицированной конструкторско-технологической базы современной микроэлектроники. По этому они строятся по блочному принципу.

Основными элементами ЭВА являются монолитные интегральные БИС, СБИС, для которых значение СИ соответственно равны: 1000 (и более) и 10^6 элементов и компонентов.

Конструкция ЭВА всегда содержит иерархию типовых элементов конструкций (ТЭК). Под ТЭК можно понимать конструктивно законченное изделие любой сложности.

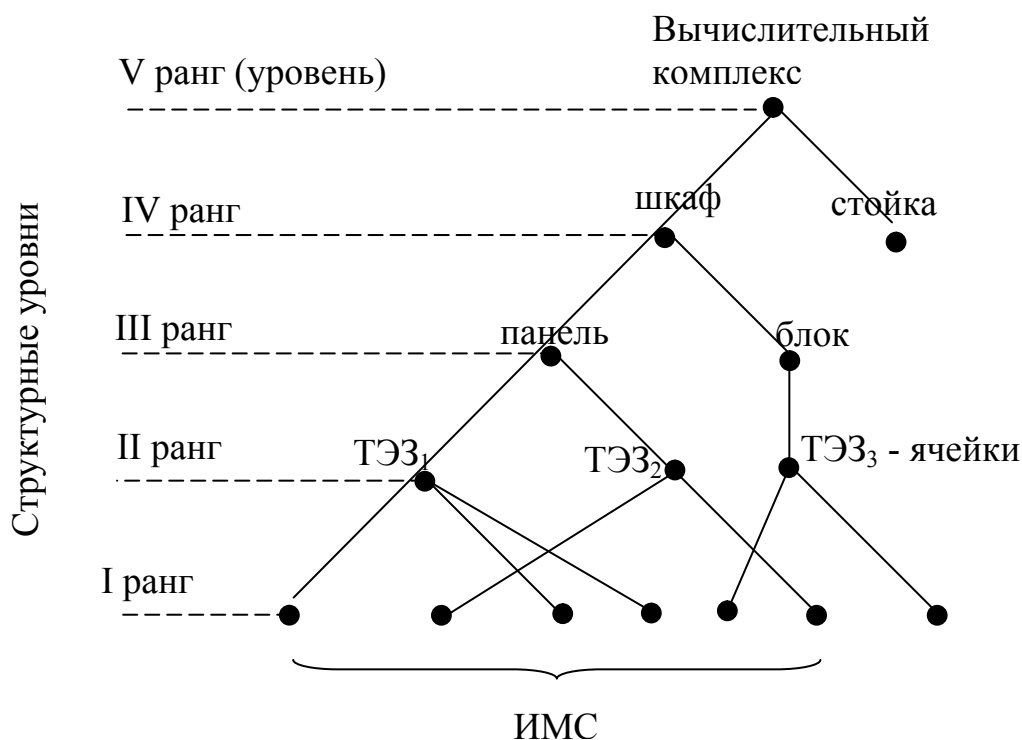
Съемные ТЭК называют типовыми элементами замены (ТЭЗ).

Обычно говорят о конструировании на определенном ранге (уровне).

Например, ТЭК первого (низшего) ранга – это может быть транзистор, резистор, интегральная микросхема (ИМС), модуль, микромодуль и

т.д., т.е. конструктивно неделимый базовый элемент, на котором ведется конструирование.

На втором уровне элементы первого ранга, например ИМС, объединяются в схемные сочетания, образуя ТЭЗы (типовой элемент замены) или ячейки и т.д.



Таким образом, ТЭК i -го ранга объединяются в ТЭК $(i+1)$ -го ранга.

Как правило, конструкторско-техническая база ЭВА состоит из пяти уровней (рангов) (см. схему на рисунке 6).

В общем случае при переходе от ранга i к рангу $(i+1)$ приходится решать задачу компоновки, т.е. разбиение ТЭК уровня i на ТЭК уровня $(i+1)$.

Исключение составляет компоновка элементов самого низкого ранга, которые составляются не из конструктивных элементов, а из функциональных базовых элементов.

Задачи компоновки на всех уровнях иерархии конструктивного оформления ЭВА тесно связаны между собой. Компоновка рассматривается как процесс принятия решения. В результате решения задачи компоновки типовые элементы конструкции (ТЭК) i -го уровня, размещаются в ТЭК $(i+1)$ -го уровня. При этом учитывается выбранный критерий качества.

Преобразование функционального описания аппаратуры в соответствующее ему описанию схемы электрической принципиальной может быть выполнено двумя способами: «снизу-вверх» и «сверху-вниз». На

практике применяются оба способа и их различные комбинации. Например, можно функциональную схему (ΦC_x) устройства покрыть ТЭК второго ранга (ячейками), а затем уже схему каждой ячейки покрывать модулями.

Существует три варианта постановки задачи компоновки, в зависимости от принятых критериев:

- типизация – разбиение ΦC_x на части различных типов по критерию минимума числа разнотипных узлов (минимум номенклатуры узлов);
- покрытие – преобразование ΦC_x в принципиальную электрическую схему, т.е. в схему соединений ТЭК, номенклатура которых заранее известна;
- разрезание – разбиение ΦC (или ПЭС) на части с минимизацией числа связи между этими частями и ограничением числа связей каждой части.

Исходными данными для решения задачи компоновки является коммутационная схема (КС). Для алгоритмизации и формального решения задачи производится переход от КС к графу одним из методов, описанных выше.

Классификация алгоритмов компоновки приведена на рисунке 7.



Рисунок 7

Алгоритмы классифицируются по критериям, по ограничениям на формирование частей или по структуре вычислительной процедуры. С этой точки зрения их делят на *последовательные*, *параллельно-последовательные* и *итерационные*. В алгоритмах первого типа вводится последовательный процесс компоновки частей, на каждом шаге которого в очередную часть добавляется один из элементов математической модели, выбираемый по определенному приоритету. В параллельно-последовательных алгоритмах сначала выделяется некоторое исходное множество групп элементов, которое затем распределяется по частям с учетом критериев и

ограничений на компоновку. Обычно эти алгоритмы используются при решении задач компоновки со специальными требованиями, например, минимизация числа разнотипных блоков. Итерационные алгоритмы служат для улучшения некоторого начального варианта компоновки в соответствии с принятым критерием.

При использовании итерационных алгоритмов сначала граф разбивается на определенное число частей произвольным образом, например с помощью последовательных алгоритмов. Затем по некоторым правилам производится перестановка вершин из одной части в другую с целью минимизации числа внешних ребер.

В алгоритмах разбиения, основанных на математических методах, в основном используются методы ветвей и границ, решение задачи о назначениях. Алгоритмы разбиения с применением методов ветвей и границ состоят из следующих этапов. Сначала определяется нижняя оценка разбиения графа на заданное число частей. Затем производится построение дерева решений и осуществляется поиск оптимального результата.

Задачу разбиения графа схемы на части можно свести к **задаче о назначении**. При этом ищется назначение кандидатов (вершин графа) на все части, дающие минимальные суммарные затраты. При этом каждая вершина может быть назначена только на одну часть и в каждой части должны содержаться различные вершины графа.

Алгоритмы компоновки типовых блоков сводятся к задачам **покрытия и типизации**. Под **покрытием** понимается представление функциональной схемы ЭВА типовыми конструктивными элементами, на которых она будет реализована, и связями между ними с выполнением заданных конструктивных ограничений. При покрытии выделяют задачи с несвязными элементами и функциональными ячейками. В первом случае решают задачи определения необходимого числа ячеек для покрытия коммутационной схемы с минимальной суммарной стоимостью (математическая модель 19 - 21). Во втором случае решают задачи покрытия коммутационной схемы заданным классом функциональных ячеек с минимизацией числа ячеек и суммарного числа межъячеечных связей (математическая модель 22 - 26).

$$L(x) = \sum_{j \in J} c_j \cdot x_j \rightarrow \min \quad (19)$$

при условиях (ограничениях):

$$\sum_{j \in J} a_{i,j} \cdot x_j \geq b_i, i = \overline{1, m}; \quad (20)$$

$$x_j \geq 0, x_j - \text{целое для всех } j, \quad (21)$$

где a_{ij} – количество элементов i -го типа, содержащиеся в ячейке j -го типа;
 x_j – количество ячеек j -го типа, используемых при покрытии элементов коммутационной схемы;
 c_j – стоимость ячейки j -го типа.

$$L(x) = \sum_{j \in J} \sum_{i=1}^k x_j \cdot \rho_{i,j} \rightarrow \min \quad (22)$$

при условиях (ограничениях):

$$\sum_{j \in J} a_{i,j} \cdot x_j \geq b_i, \quad i = \overline{1, m}; \quad (23)$$

$$x_j \geq 0, x_j - \text{целое для всех } j, \quad (24)$$

где: a_{ij} – количество элементов i -го типа, содержащиеся в ячейке j -го типа;

x_j – количество ячеек j -го типа, используемых при покрытии коммутационной схемы;

ρ_{ij} – количество внешних связей ячейки типа j при реализации ею элемента схемы i -го типа;

b_i – количество элементов i -го типа, содержащиеся в коммутационной схеме;

Типизация – это разбиение коммутационной схемы на части по критерию оптимальности – минимум номенклатуры частей разбиения или по критерию оптимальности – максимум однотипности используемых ячеек. Сокращение номенклатуры ТЭК позволяет уменьшить затраты на дальнейшее проектирование.

Задача типизации - это компоновка «сверху вниз».

$$L(x) = \sum_{j \in J} \sum_{i \in I} x_{i,j} \rightarrow \min \quad (25)$$

при условиях (ограничениях):

$$\sum_{j \in J} x_{i,j} = 1, \quad i = \overline{1, m}; \quad (26)$$

$$\sum_{i=1}^m n_i \cdot x_{ij} = k_j, \quad j \in J; \quad (27)$$

$$x_{ij} \in \{0; 1\}, \quad (28)$$

где x_{ij} - переменная проектирования; $x_{ij} = 1$, если ячейка типа j -го реализует i -й элемент схемы; $x_{ij} = 0$, если ячейка типа j не может реализовать i -й элемент схемы.

k_j - количество элементов, содержащиеся в ячейке j -го типа;

n_i - количество элементов i -го типа, содержащиеся в ячейке j -го типа.

Ограничение (26) показывает, что каждый элемент коммутационной схемы может быть реализован только одной ячейкой соответствующего типа.

Ограничение (27) показывает, что число элементов схемы, реализуемых ячейкой j -го типа не должно превышать числа элементов, содержащихся в ней.

Важной задачей в общей проблеме компоновки коммутационной схемы является покрытие. Как отмечалось выше, под покрытием схемы понимается представление функциональной схемы типовыми элементами конструкций, на которых она будет реализована, и связями между ними. Покрытие называют компоновкой «снизу в верх».

Решение «задачи покрытия» на первом уровне конструкторского проектирования дает возможность представить функциональную схему ЭВА или ее частей в виде принципиальной электрической схемы соединений элементов. Элементами могут быть резисторы конденсаторы, транзисторы, ИМС и т.д., т.е. ТЭК 1-го уровня. Поэтому при решении задачи покрытия можно рассматривать вопросы выбора класса элементов и минимизации числа типов элементов. Конечной целью покрытия является выбор оптимальной элементно-технической базы объекта проектирования.

Исходной информацией для этого этапа проектирования являются схема функциональная и набор ячеек $Y = \{Y_1, Y_2, \dots, Y_n\}$, параметры которых оказывают решающее влияние на результаты покрытия схемы.

Пусть задана схема, состоящая из элементов x_1, x_2, \dots, x_n , для каждого из которых известен тип $t(x_i)$, $t \in \{1, \dots, l\}$. Состав схемы по типам элементов описывается вектором $M = \{m_1, m_2, \dots, m_l, \dots, m_l\}$, в котором m_t – число элементов типа t . Состав ячеек заданного набора Y описывается матрицей $R = \|r_{v,t}\|$, $v \in \{1..k\}$, $t \in \{1..n\}$, где $r_{v,t}$ число элементов типа t в ячейке Y_v . *Схема считается покрытой ячейками из набора Y , если каждый элемент схемы реализуется элементами, входящими в состав выбранных ячеек.* Пусть известны стоимости c_v ячеек каждого типа. Требуется отыскать по-

крытие с минимальной стоимостью F :
$$F = \sum_{t=1}^n c_v \cdot |R_v|$$
.

Для формализации процесса покрытия схемы ячейками, различные наборы делятся на классы.

1. В состав ячеек входят логические элементы одного типа, не связанные между собой; входы и выходы элементов имеют отдельные выводы на ячейке.

2. Ячейки содержат разнотипные элементы, не связанные между собой; входы и выходы элементов имеют отдельные выводы на ячейке.

3. Состав ячеек аналогичен классу 1, однако часть элементов связана между собой.

4. Состав ячеек аналогичен классу 2, однако часть элементов связана между собой.

Рассмотрим задачу, относящуюся к классу 2. Математическая модель задачи покрытия для данного класса запишется:

$$F = \sum_{t=1}^n c_v \cdot |Y_v| \Rightarrow \min \quad (29)$$

при условиях:

$$\sum_{v \in V} r_{v,t} \cdot |Y_v| - \sum_{a=1}^l S_{t,a} + \sum_{v=1} S_{a,t} \geq m_t; \quad (30)$$

$$|Y_v| \geq 0; \quad (31)$$

$$S_{t,a} \geq 0 - \text{целые числа}, \quad (32)$$

где m_t - число элементов типа t , используемых в схеме;

$S_{t,a}$ - количество элементов типа « t », которые могут быть заменены элементами типа « a »;

$S_{a,t}$ - количество элементов типа « a », которые могут быть заменены элементами типа « t »;

$|Y_v|$ - число ячеек типа Y_v , использованных при реализации схемы.

Задача (29-32) относится к задачам целочисленного программирования. Ее решение точными методами затруднено. По этому применяют приближенную эвристическую процедуру решения поставленной задачи. На практике хорошо зарекомендовал себя алгоритм Селютина. Рассмотрим алгоритм Селютина компоновки ячеек с несвязными элементами. Пример такой ячейки «K155LA3» приведен на рисунке 8. В состав данной ячейки входят однотипные элементы, реализующие функцию «И-НЕ» на два входа.

Для общности примем, что набор ячеек относится к классу 2, т.е. в него входят ячейки с разнотипными элементами.

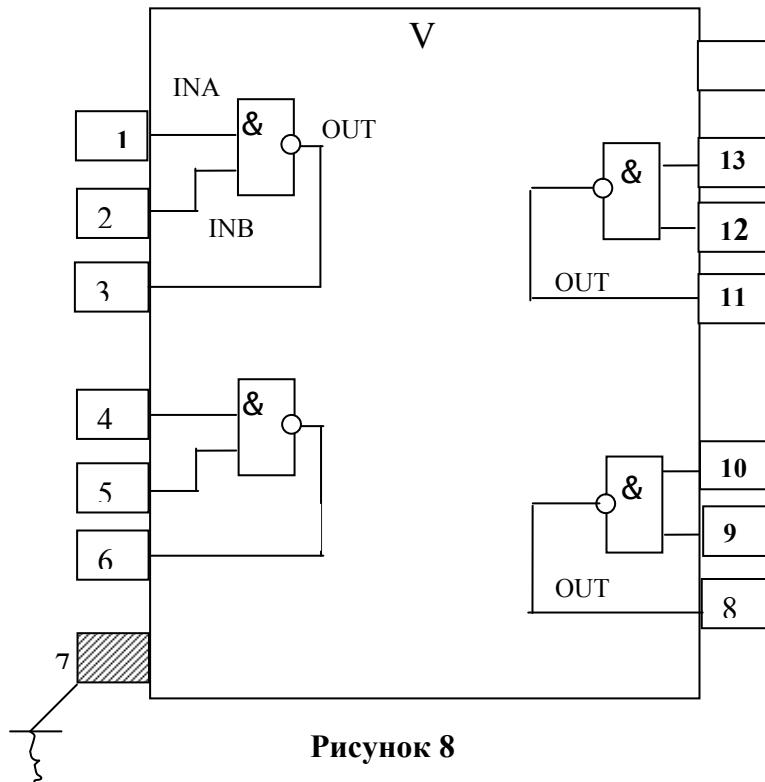


Рисунок 8

При решении задачи будем считать, что каждый элемент схемы реализуется элементом того же типа в ячейках набора Я. В качестве дополнительного критерия оптимизации принимается число межъячеечных соединений. Решение задачи разбивается на два этапа: 1) определение необходимого числа ячеек с минимальной суммарной стоимостью; 2) минимизация числа связей между ячейками. Пусть дана матрица $R = \parallel r_{v,t} \parallel$, $v \in \{1..k\}$, $t \in \{1..n\}$, и вектор числа элементов в схеме по типам $M_0 = \parallel m_t \parallel$.

Алгоритм решения задачи состоит в следующем.

1. Упорядочить ячейки набора $\{Y_v\}$ по возрастанию их стоимости: $Y_1; Y_2; Y_3; \dots; Y_v$ ($C_1 \leq C_2 \leq \dots \leq C_k \leq \dots \leq C_v$). Выбирается ячейка Y_k с наименьшим значением стоимости C_k . Задаем начальное значение номера итерации $i=0$. По результатам структурного анализа покрываемой схемы определяем вектор M_0 , элементы которого равны числу каждого типа элементов, использованных в схеме.

2. Определить «локально-минимальное» число ячеек Y_k для итерации $i = i+1$: $A_k^{(i)} = \min \{m_t / r_{k,t}\}$, где $r_{k,t}$ – количество элементов типа «t» в ячейке типа «k».

3. Вычислить вектор «непокрытых элементов» M_i : $M_i = M_{i-1} - A_k^{(i)} R_k$.

4. Если $M_i > 0$, перейти к п.2, если $M_i = 0$, перейти к п.5.

5. Подсчитать число ячеек каждого типа: $|Y_v| = \sum_{i=1}^p A_v^{(i)}$, где p – число выполненных при покрытии итераций; $v = \{1, 2, 3, \dots, k, \dots, v\}$.

Рассмотрим пример. Дана логическая схема (рисунок 9), типы используемых элементов (рисунок 10).

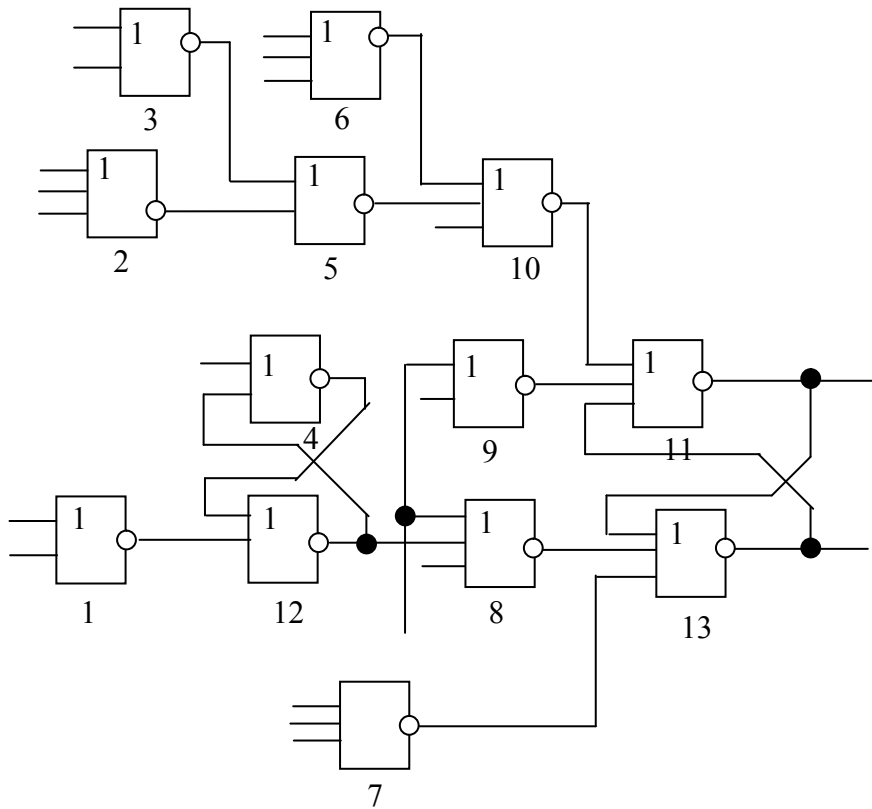


Рисунок 9

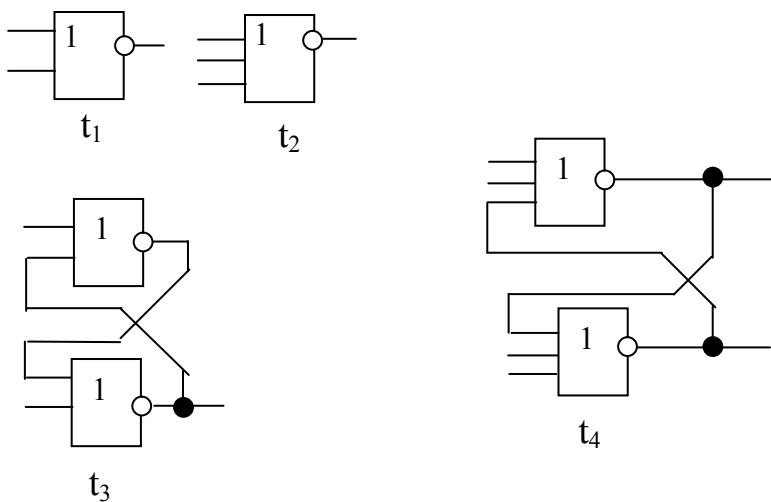


Рисунок 10

Типовые элементы проектирования t_1, t_2, t_3, t_4 входят в набор ячеек $Я_1, Я_2, Я_3$ (рисунок 11):

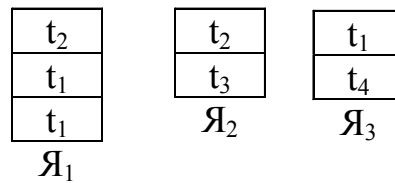


Рисунок 11

Составим матрицу R_v описания состава данного набора ячеек:

$$R_v = \begin{array}{c|cccc} & t_1 & t_2 & t_3 & t_4 \\ \hline & 2 & 1 & 0 & 0 \\ & 0 & 1 & 1 & 0 \\ & 1 & 0 & 0 & 1 \end{array}$$

Строки матрицы R_v соответствуют ячейкам заданного набора – $Я_1, Я_2, Я_3$. Столбцы – типовым элементам проектирования: t_1, t_2, t_3, t_4

Данные структурного анализа схемы приведены в таблице 1.

Таблица 1

Тип элемента схемы	t_1	t_2	t_3	t_4	t_5
число использования элемента в схеме	5	4	1	1	0

В соответствии с исходными данными, математическая модель задачи покрытия запишется:

$$L = c_1x_1 + c_2x_2 + c_3x_3 \rightarrow \min \quad (33)$$

при условиях:

$$2x_1 + x_3 \geq 5 ; \quad (32)$$

$$x_1 + x_2 \geq 4 ; \quad (33)$$

$$x_2 \geq 1 ; \quad (34)$$

$$x_3 \geq 1 ; \quad (35)$$

$$\forall x_j \geq 0, x_j - \text{целое}, \quad (36)$$

где: x_j – количество ячеек j -го типа, выбранных для покрытия схемы.

Решение задачи покрытия по алгоритму Селютина.

1. По данным таблицы 1 запишем вектор $M_0 = (5, 4, 1, 1)$.
2. Упорядочим ячейки заданного набора $\{Y_v\}$ по возрастанию их стоимости:

$$Y_1; Y_2; Y_3 \quad (C_1 \leq C_2 \leq C_3).$$

3. Выбираем ячейку Y_1 , т.к. она имеет наименьшую стоимость.

Определяем «локально-минимальное» число $A_1^{(1)}$ использования ячейки Y_1 . Ячейка Y_1 может реализовать элементы схемы типа t_1, t_2 (см. матрицу R_v).

Для реализации элементов схемы типа t_1 требуется две ячейки $Y_1 \rightarrow A_{1,1} = (5 \div 2 = 2)$;

для реализации элементов схемы типа t_2 требуется четыре ячейки $Y_1 \rightarrow A_{1,2} = (4 \div 1 = 4)$.

Для нашего примера $A_1^{(1)} = \min \{A_{1,1}, A_{1,2}\}$; $A_1^{(1)} = 2$. Следовательно, $|Y_1| = 2$ шт.

Переход к следующей итерации: $i = i + 1, i = 0 + 1, i = 1$.

4. Вычисляем вектор «непокрытых элементов» M_i , где i – номер выполненной итерации ($i = 1$).

$$M_1 = M_0 - A_1^{(1)} \cdot r_{1,1}; \text{ получаем } M_1 = (1, 2, 1, 1).$$

5. Так как вектор $M_1 \neq 0$, то выполняем вторую итерацию ($i = 2$) для ячейки Y_2 . Все вычисления проводятся аналогично вычислениям на первой итерации, но с новыми значениями вектора M_i .

$$6. \quad A_2^2 = \min \begin{cases} A_{2,2} = 2 \\ A_{2,3} = 1 \end{cases} ; \quad A_2^2 = 1; \quad |Y_2| = 1; \quad M_2 = (1, 1, 0, 1).$$

Так как $M_2 > 0$, - выполняем 3-ю итерацию.

7. Третья итерация выполняется для ячейки Y_3 .

$$A_3^3 = \min \begin{cases} A_{3,1} = 1 \\ A_{3,4} = 1 \end{cases} ; \quad |Y_3| = 1.$$

Вычисляем $M_3 = (0, 1, 0, 0)$.

Так как $M_3 > 0$, - выполняем 4-ю итерацию.

8. Вектор M_3 содержит только один ненулевой элемент – m_2 . Следовательно в схеме остался только один нереализованный элемент типа t_2 , который может быть реализован ячейками Y_1 и Y_2 .

Так как $C_1 < C_2$ выбираем ячейку Y_1 .

9. Вычисляем вектор M_4 : $M_4 = (0, 0, 0, 0)$.

Так как $M_4 = 0$, решение задачи заканчивается. Остается подсчитать суммарное количество ячеек каждого типа, вычисленное на каждой итерации.

Для данного примера $|Y_1| = 3, |Y_2| = 1, |Y_3| = 1$.

Выше говорилось о том, что ячейка – это неделимый базовый элемент конструкции, по этому при покрытии допускается некоторая избыточность элементов ячеек в виду их недоиспользования, на пример одна из трех ячеек Y_l имеет избыточность по элементам типа t_l .

Если функциональную схему представить графом $G=(X,U)$, где X – множество вершин, интерпретирующих логические элементы схемы, а множество ребер U – связи.

Пусть функциональную схему электронной радиоаппаратуры необходимо покрыть заданным комплексом интегральных микросхем (БИС). Тогда каждый компонент БИС можно представить в виде подграфа G' . В результате получим некоторое множество подграфов, соответствующих ИМС заданного комплекса. Задача покрытия формулируется теперь как покрытие графа G подграфами из множества $\{G'_1, G'_2, \dots, G'_p\}$. Один из возможных путей решения этой задачи следующий. Каждой вершине графа G присвоить код $S_{i,p}$, где i – номер вершины; p – тип элемента логической схемы.

Каждому ребру графа ставится в соответствие значение функции веса $g_{i,k}$.

$$g_{i,k} = \begin{cases} 1, & \text{если ребро } u_i, \text{ инцидентно однотипным вершинам;} \\ 0, & \text{в противном случае.} \end{cases}$$

Внутри ИМС, как правило, объединяются однотипные компоненты. Из нескольких вариантов покрытия схемы выбирают тот, для которого:

$$g = \frac{1}{2} \sum_i^n \sum_k^n g_{ik} \Rightarrow \min.$$

Рассмотрим задачу типизации. *Типизация* – это разбиение коммутационной схемы на части с минимизацией номенклатуры частей разбиения. Однотипными называются ТЭК, имеющие одинаковый состав элементов и одинаковую коммутационную схему. В этом случае задача типизации формулируется как задача выделения в графе изоморфных подграфов (А.М. Бершадский). Задача выделения в графе изоморфных подграфов формулируется следующим образом: найти разбиение $\phi(G) \in \Phi$ графа G на множество групп изоморфных подграфов, удовлетворяющих следующим условиям:

- 1) любые два подграфа G_i, G_k , принадлежащие одной группе разбиения, должны быть изоморфны;
- 2) множества вершин любых двух подграфов не должны пересекаться;
- 3) число вершин любого подграфа разбиения не должно превышать заданное (конструктивное ограничение, связанное с числом элементов на типовом элементе конструкции);

4) суммарное число внешних ребер каждого подграфа не должно превышать заданное (конструктивное ограничение, связанное с числом элементов разъема или длиной параметра корпуса ТЭК).

Критерием оптимальности при типизации является минимальное число групп изоморфных подграфов, полученных в результате разбиения.

Для решения задачи типизации путем сведения ее к задаче выделения в графе изоморфных подграфов необходимо:

1) описать исходную коммутационную схему с помощью графовой модели;

2) решить для графа задачу выделения в нем изоморфных подграфов;

3) от полученных результатов решения задачи на графе необходимо перейти к результатам решения задачи типизации в коммутационной схеме.

4.2 Размещение конструктивных элементов в монтажном пространстве (задача размещения)

Размещение определяет качество трассировки, быстродействие, надежность. Из-за малого времени переключения схем ЦВМ – длина соединений влияет на задержки в распространении сигнала, затрудняет синхронизацию.

Исходные данные для размещения:

1) прямоугольная конструкция (ячейка, кристалл, панель);

2) число элементов для размещения;

3) граф схемы соединений элементов $G=(X, U)$.

На заданную конструкцию накладывается координатная сетка (решетка) и оси координат s, t , представляя ее таким образом в виде графа G_r . Теперь задача размещения сводится к отображению заданного графа схемы $G=(X, U)$ в координатную сетку G_r , таким образом, чтобы вершины множества X размещались в ее узлах и суммарная длина связей

$$L(G) = \frac{1}{2} \sum_{i,j} d_{i,j} r_{i,j} \text{ или суммарное число пересечений } P(G) = \frac{1}{2} \sum_{u_{i,j} \in U} P(u_{i,j})$$

были наименьшими для возможных способов отождествления вершин графа и узлов сетки. В приведенных формулах переменная $r_{i,j}$ равна числу кратных ребер между вершинами x_i, x_j , а $P(u_{i,j})$ - числу пересечений ребра $u_{i,j}$ со всеми остальными ребрами графа, $d_{i,j}$ расстояние между узлами сетки, в которых размещены вершины x_i, x_j .

Расстояние $d_{i,j}$ между узлами сетки, в которых размещены вершины x_i, x_j определяется выражением - $d_{ij}=|s_i - s_j|+|t_i - t_j|$.

Функция расстояний для узлов сетки G_r задается матрицей расстояний D .

$$D = \|d_{ij}\|; \quad d_{ij} = \begin{cases} 0, & \text{если } x_i = x_j; \\ d_{ij}, & \text{если } x_i \neq x_j. \end{cases}$$

Существует большое число алгоритмов размещения графа схемы с минимизацией суммарной длины соединений и внутрисхемных пересечений. Все алгоритмы можно разделить на две группы: непрерывно-дискретные и дискретные. К первой группе относятся алгоритмы, основанные на градиентных методах. Ко второй группе относятся

Итерационные, последовательные, смешанные, а также алгоритмы, основанные на идеях метода ветвей и границ.

Последовательные алгоритмы размещения заключаются в выборе первоначально размещенного элемента или группы элементов с последующим подсоединением не размещенных элементов. После размещения элементов они уже не перемещаются. Правила выбора и расстановки элементов зависят от конкретных методов.

Итерационные алгоритмы размещения с улучшением качества работают в итеративном режиме. Для изменения позиций размещения элементов выбираются одиночные элементы или группы элементов. Затем по заданным правилам производится перемещение элементов для уменьшения общей длины соединений, что позволяет получать более качественные результаты, чем последовательные за счет больших затрат машинного времени. К группе итерационных алгоритмов относятся *стохастические методы* размещения. Основная идея этих методов следующая. Случайным образом распределяются элементы по посадочным местам плоскости с учетом плотности распределения вероятности, которую считают равномерной. Далее определяется суммарная длина соединений и в полученном размещении и сравнивается с предыдущим. Лучшее размещение сохраняется. Процесс продолжается до тех пор, пока не окончится отведенное время или не будет просмотрено заданное число размещений.

Алгоритмы, основанные на идеях *метода ветвей и границ*, относятся к точным. При этом множество всех допустимых решений разбивается на меньшие по мощности подмножества, в которых производится поиск оптимального размещения. Метод сопровождается вычислением нижних границ. Поиск оптимального решения прекращается, когда граничное значение начинает превышать значение при найденном допустимом размещении. Процесс продолжается до тех пор, пока не будет закончен поиск в каждом подмножестве разбиения или не будет найдено оптимальное размещение.

При реализации алгоритмов в общем случае могут получаться локальные минимумы целевой функции.

Рассмотрим *последовательно-итерационный алгоритм* размещения вершин графа на плоскости с минимизацией суммарной длины соединений. *Последовательная часть*, которая применяется для упорядочи-

вания множества вершин и получения начального размещения, использует понятие коэффициента связности для каждой вершины.

Исходными данными для решения размещения являются граф $G=(X,U)$, интерпретирующий схему электрическую принципиальную, и монтажная плоскость заданных размеров, на которой размещен граф сетки G_r . В общем случае считают, что монтажная плоскость имеет прямоугольную форму. В частном случае форма монтажной плоскости может быть любой формы. Требуется разместить вершины $x_i \in X$ графа G в узлы сетки G_r с минимизацией суммарной длины ребер $u_j \in U$. Число узлов сетки G_r больше или равно числу вершин графа G . Предположим, что часть узлов сетки G_r уже отождествлена с некоторыми вершинами графа схемы.

Первый этап размещения

Пусть вершина $x_i \in X$ помещена в $(j+1)$ -ю позицию, где j – число занятых позиций, тогда коэффициентом связности $\Delta(x_i)$ называется выражение: $\Delta(x_i) = a_{i,h} - a_{i,z}$, где $a_{i,h}$ – число ребер, связывающих x_i -ю вершину с ранее размещенными; $a_{i,z} = p(x_i) - a_{i,h}$, т.е. число ребер, связывающих x_i с не размещенными вершинами, где $p(x_i)$ – число ребер, инцидентных вершине x_i (локальная степень x_i). Тогда $\Delta(x_i) = 2a_{i,h} - p(x_i)$, где $a_{i,h}$ число ребер, соединяющих x_i с ранее размещенными вершинами. Значение $\Delta(x_i)$ определяется для всех не размещенных вершин и выбирается x_i с $\max \Delta(x_i)$. Вершина x_i помещается в первую свободную позицию (узел сетки G_r). Как правило размещение вершин начинается с крайнего левого узла сетки. Процесс последовательно продолжается до тех пор, пока не будут размещены все вершины графа G .

Второй этап размещения (итерационная часть алгоритма размещения)

Итерационная часть алгоритма размещения основана на понятии центра тяжести вершины.

Для оценки степени «предпочтительности» v -й позиции для каждой вершины графа x_j вводится понятие *средней длины ребра* – L_j .

$$L_j = \sum_{i=1}^n d_{i,j} \cdot r_{i,j} / \sum_{i=1}^n r_{i,j}, \quad i \neq j, \quad \text{где } d_{i,j} \text{ – расстояние между узлами}$$

сетки, в которые помещены вершины x_i и все смежные ей j -е вершины; $r_{i,j}$ – число связей вершины x_j со всеми смежными с ней j – ми вершинами графа G ; L_j – средняя длина ребер, инцидентных вершине x_j , помещенной в позицию v .

Очевидно, что $(\forall u_{ij} \in U_j)[(d_{i,j} = 1) \Rightarrow L_j = 1]$, где U_j – множество ребер, инцидентных вершине x_j , помещенной в позицию v . *Необходимо*

стремиться к такому размещению вершин графа, когда средняя длина ребер стремится к минимальному значению.

Из множества вершин графа, расположенных в сетке, выбирается вершина x_j , имеющая

максимальное значение L_j и производится ее перестановка с другой вершиной с целью минимизации $L(G) = \frac{1}{2} \sum_{i,j} d_{i,j} r_{i,j}$. Очевидно, что в перестановках должны участвовать только смежные вершины.

Для характеристики системы точек, связанных с вершиной x_j вводится понятие **центра тяжести**, что позволяет рассматривать их как систему связанных материальных точек и отыскать такое положение для x_j , которое обеспечит равновесие в сетке с точки зрения минимизации $L(G)$. Вычисление координат центра тяжести (s_c и t_c) системы материальных точек производится по формулам:

$$\left. \begin{aligned} S_c &= \sum_{j=1}^n S_j \cdot r_{i,j} / \rho(x_i) \\ t_c &= \sum_{j=1}^n t_j \cdot r_{i,j} / \rho(x_i) \end{aligned} \right\} \text{координаты "центра тяжести" вершины } x_j,$$

где $r_{i,j}$ – количество связей между вершиной x_o и смежными с ней вершинами $\{x_i\}$; вершина x_i – это вершина с максимальным значением L_j ; $\rho(x_i)$ – степень вершины x_i .

Оценка качества решения задачи размещения в целом производится по формуле:

$$L_{\text{общ}} = \frac{1}{2} \sum_i^N \sum_j^N d_{i,j} \cdot r_{i,j}.$$

Данная оценка вычисляется после выполнения каждой итерации алгоритма размещения и ее значение сравнивается со значением, вычисленным на предыдущей итерации. Принятая перестановка вершин должна приводить к уменьшению величины суммарной длины соединений, в противном случае вершины не меняют свое местоположение.

На основании рассмотренного математического аппарата решения задачи размещения с помощью последовательно-итерационного алгоритма запишем основные его шаги и процедуры.

1. На заданную монтажную конструкцию наложить координатную сетку с осями координат s, t , т.е. построить граф G_r .

2. Выполнить первоначальное размещение вершин графа схемы G , совместив их с узлами сетки G_r .

3. Для всех вершин графа G вычислить значения суммарных длин связей L_j .

4. Выбрать вершину x_i с $\max L_j$.

5. Вычислить для вершины x_i координаты центра тяжести s_c и t_c .

6. Определить подмножество вершин X' , вершины которого определяют область возможных перестановок с вершиной x_i .

Произвести парные перестановки вершин множества X' с вершиной x_i и вычислить для каждой вершины $x_j \in X'$, а также для вершины x_i величины L_j^v и L_i^v с учетом нового местоположения (L_j^v - это длина связей j -ой вершины, совмещенной с узлом сетки v).

7. Вычислить отклонение $\sigma_j^v = L_j^v - L_j^k$ для $x_j \in X'$ для всех вершин, выбранных в соответствии с п.п.4,6, где v и k соответственно «старая» и «новая» позиции местоположения вершин.

8. Вычислить значения $\delta_{ij} = \sigma_i^v + \sigma_j^a$ парных перестановок вершин множества X' и вершины x_i . Выбрать такую пару вершин, для которой значение δ_{ij} максимальное.

9. Произвести перестановку вершин.

Пример решения задачи размещения

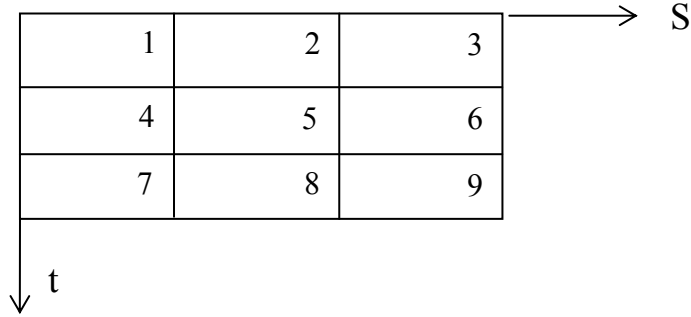
Дан граф $G=(X,U)$. Его матрица смежности $R=||r_{ij}||_{9 \times 9}$ имеет следующие значения элементов:

$r_{1,4}=3$	$r_{4,1}=3$	$r_{2,3}=2$	$r_{4,2}=1$
$r_{1,7}=2$	$r_{7,1}=2$	$r_{2,4}=1$	$r_{3,5}=2$
$r_{1,8}=3$	$r_{8,1}=3$	$r_{3,2}=2$	$r_{5,3}=2$
$r_{4,6}=5$	$r_{6,7}=5$	$r_{7,9}=2$	$r_{7,8}=3$
$r_{6,4}=5$	$r_{7,6}=5$	$r_{9,7}=2$	$r_{8,7}=3$
$r_{5,6}=2$	$r_{6,8}=6$	$r_{5,9}=4$	
$r_{6,5}=2$	$r_{8,6}=6$	$r_{9,5}=4$	

Значение элемента r_{ij} матрицы R равно числу ребер, связывающих i -ю и j -ю вершины графа G .

Требуется элементы графа разместить на монтажной плоскости прямоугольной формы так, чтобы суммарная длина соединений была наименьшей.

Шаг 1. На монтажную плоскость накладываем граф-сетку G_r из 9 – узлов с координатами $(S \times T)$. Примем, что координаты узла 1 имеют значения $(0,0)$.



Шаг 2. Отобразим граф G в координатную сетку G_r так, чтобы номера его вершин совпадали с номерами узлов координатной сетки. В общем случае номер вершины, может не совпадать с номером узла (рисунок 11а).

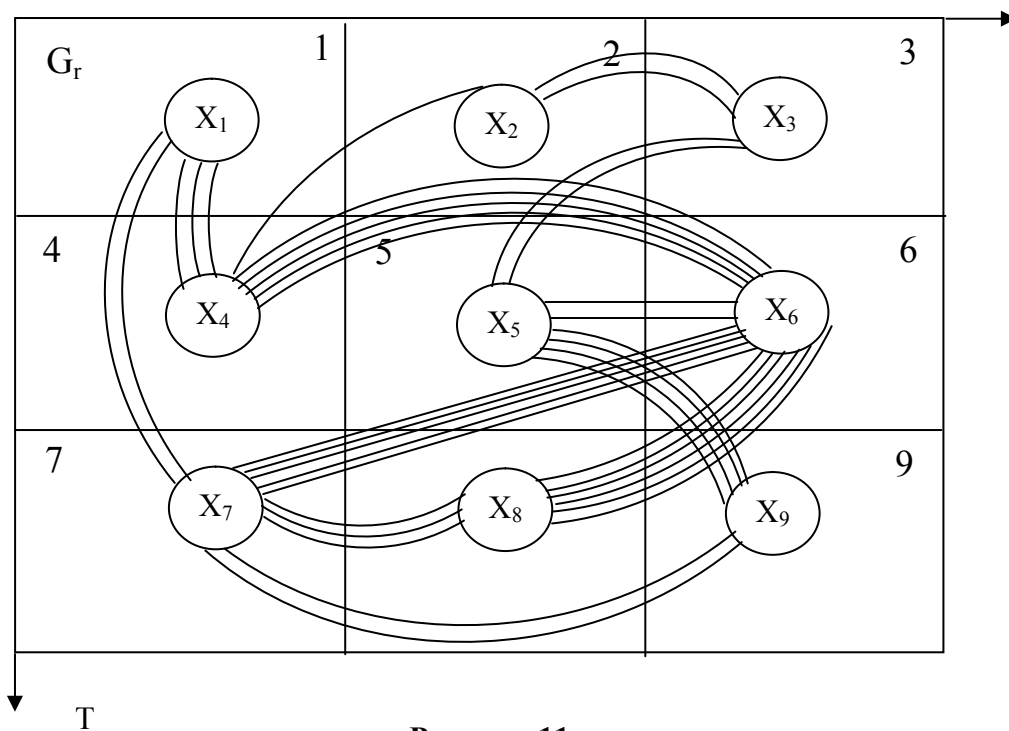


Рисунок 11а

Шаг 3. Определяем L_j для всех вершин $x_j \in X$. Значения d_{ij} – определяем по формуле:

$$d_{ij} = |S_i - S_j| + |t_i - t_j|;$$

$$L_j = \sum_{i=1}^n d_{ij} r_{ij} \bigg/ \sum_{i=1}^n r_{ij}, i \neq j.$$

$$\begin{aligned}
L_1 &= d_{14} \cdot r_{14} + d_{18} \cdot r_{18} + r_{17} \cdot d_{17} / r_{14} + r_{17} + r_{18} = \{(10 - 01^1 + 10 - 11) \cdot 3 + \\
&+ (10 - 21^3 + 10 - 11) \cdot 3 + (10 - 01^2 + 10 - 21) \cdot 2\} / 3 + 3 + 2 = \\
&= \{3 + 9 + 4\} / 8 = 16 / 8 = 2.
\end{aligned}$$

Аналогично вычисляем:

$$L_2 \approx 1,3; L_3 \approx 1,5; L_4 \approx 1,7; L_5 \approx 1,7; L_6 \approx 2,2; L_7 \approx 2,2; L_8 = 2; L_9 = 2.$$

Вершина x_6 , имеет наибольшую суммарную длину связей :
 $L_j = L_6 = 2,2$.

Шаг 4. Вычисляем координаты центра тяжести t_c , s_c для вершины x_6 , так как она имеет наибольшую суммарную длину связей L_j :

$$\left. \begin{aligned} t_c &= \sum_{j=1}^n t_j r_{i,j} / \rho(x_j) \\ s_c &= \sum_{j=1}^n S_j \cdot r_{i,j} / \rho(x_j) \end{aligned} \right\} \quad (*)$$

Для вычисления по формуле (*) выбираются все вершины, смежные x_6 . Это x_4, x_5, x_7, x_8 , т.е. $j = \{4; 5; 7; 8\}$.

$$t_c^6 = (1 \cdot 5 + 1 \cdot 2 + 2 \cdot 5 + 2 \cdot 6) / (5 + 2 + 5 + 6) = 29 / 18 \approx 1,4;$$

$$s_c^6 = (0 \cdot 5 + 1 \cdot 2 + 0 \cdot 5 + 1 \cdot 6) / (5 + 2 + 5 + 6) = 8 / 18 \approx 0,4.$$

Шаг 5. По значениям координат (s_c^6 и t_c^6) определяем подмножество вершин, которые можно поменять местами с вершиной x_6 . Это все вершины, попавшие в радиус равный 1 (из узла 4), - x_4, x_5, x_7, x_8 , так как они смежны вершине x_6 .

Шаг 6. Вычисляем новые значения $L_5^6, L_4^6, L_8^6, L_7^6$ для вершин x_4, x_5, x_7, x_8 , условно переставив каждую из них на место вершины x_6 , т.е. в узел 6, и значения $L_6^4, L_6^5, L_6^7, L_6^8$ для x_6 , условно поместив ее в узлы размещения вершин x_4, x_5, x_7, x_8 .

Пример вычисления L_4^6 :

$$L_4^6 = \frac{d_{64} r_{46} + d_{61} r_{41} + d_{62} r_{42}}{r_{46} + r_{41} + r_{42}} = \frac{2 \cdot 5 + 3 \cdot 3 + 2 \cdot 1}{5 + 3 + 1} = \frac{21}{9} \approx 2,3.$$

Аналогично вычисляются значения: $L_5^6 \approx 1; L_7^6 \approx 2,4; L_8^6 \approx 2,5$.

Пример вычисления L_6^4 :

$$L_6^4 = \frac{d_{46}r_{46} + d_{45}r_{65} + d_{47}r_{67} + d_{48}r_{68}}{r_{46} + r_{65} + r_{67} + r_{68}} \approx 1,6.$$

Аналогично вычисляются значения:

$$L_6^4 \approx 1,6; L_6^5 \approx 1,3; L_6^7 \approx 1,7; L_6^8 \approx 1,6.$$

Шаг 7. Вычисляем значения отклонений σ_j^V для вершин x_6 и x_5 ; x_4 ; x_7 ; x_8 .

Для вершины x_6 :

$$\sigma_6^4 = L_6 - L_6^4 = 2,2 - 1,6 = 0,6;$$

$$\sigma_6^5 = L_6 - L_6^5 = 2,2 - 1,3 = 0,9;$$

$$\sigma_6^7 = L_6 - L_6^7 = 2,2 - 1,7 = 0,5;$$

$$\sigma_6^8 = L_6 - L_6^8 = 2,2 - 1,6 = 0,6;$$

для вершин x_4 ; x_5 ; x_7 ; x_8

$$\sigma_4^6 = L_4 - L_4^6 = 1,7 - 2,3 = -0,6;$$

$$\sigma_5^6 = L_5 - L_5^6 = 1,7 - 1 = 0,7;$$

$$\sigma_7^6 = L_7 - L_7^6 = 2,2 - 2,4 = -0,2;$$

$$\sigma_8^6 = L_8 - L_8^6 = 2 - 2,5 = -0,5.$$

Шаг 8. Вычисляем коэффициент $\delta_{ij} = \sigma_i^V + \sigma_j^a$ для вершин x_6 и x_5 ; x_4 ; x_7 ; x_8 .

$$\delta_{4,6} = \sigma_6^4 + \sigma_4^6 = 0,6 + (-0,6) = 0;$$

$$\delta_{5,6} = \sigma_6^5 + \sigma_5^6 = 0,9 + 0,7 = 1,6;$$

$$\delta_{7,6} = \sigma_6^7 + \sigma_7^6 = 0,5 + (-0,2) = 0,3;$$

$$\delta_{8,6} = \sigma_6^8 + \sigma_8^6 = 0,6 + (-0,5) = 0,1.$$

Выбираем вершины с максимальным значением δ_{ij} .

Максимальное значение $\delta_{ij} = \delta_{5,6}$ имеет взаимная перестановка вершин x_6 и x_5 . Следовательно, меняем местами вершины x_6 и x_5 .

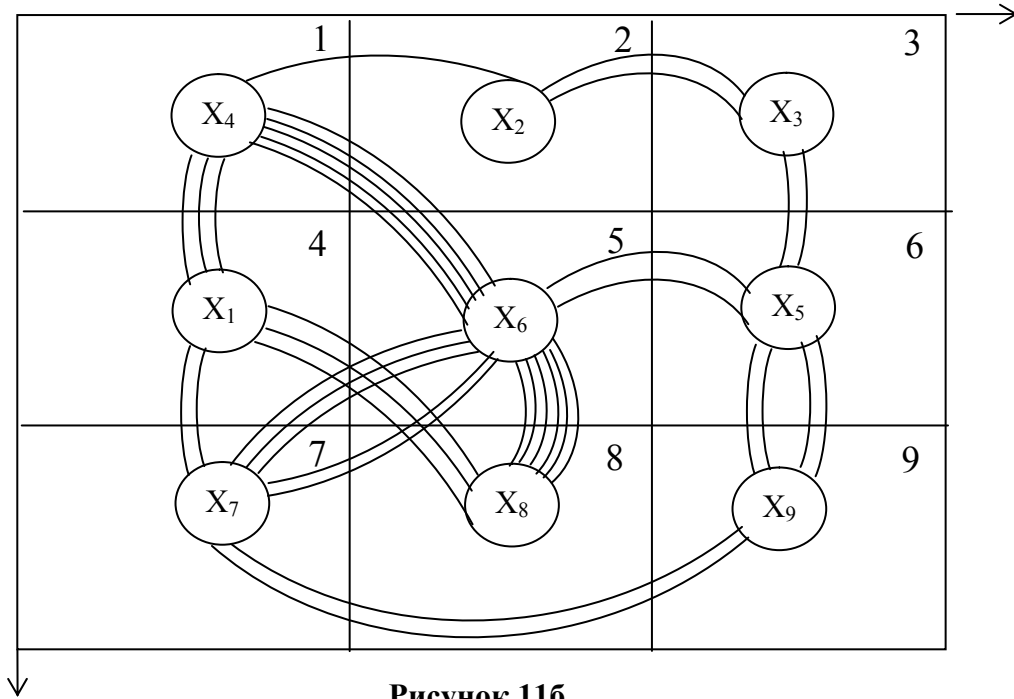


Рисунок 11б

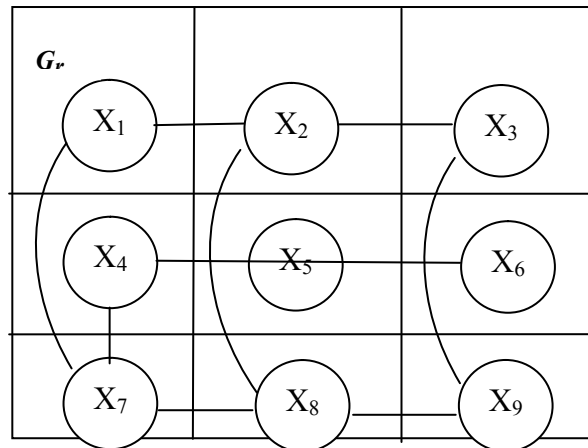
Т

На этом первая итерация алгоритма размещения заканчивается. Переходим к новой итерации, повторяя аналогичные вычисления, в соответствии с шагами 3-8.

Окончательный вариант размещения (рисунок 11б) имеет суммарную длину $L(G)=55$. Таким образом, суммарная длина уменьшилась с $L(G)=76$ до $L(G)=55$.

В случае, когда не возможно выбрать перестановку, уменьшающую длину соединений $L(G)$, делают перестановку групп вершин, которая выполняется с помощью, так называемой, факторизации графа по строкам и столбцам. Для этого каждой строке графа G_r ставят в соответствие вершину мультиграфа G_μ .

Так графу, представленному на рисунке 12, соответствует мультиграф G_μ на рисунках 13-14.



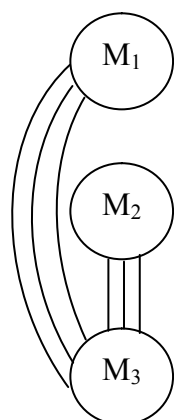


Рисунок 13

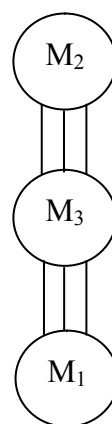


Рисунок 14

Далее «развернем» вершины мультиграфа, представленного на рисунке 14, соответственно вершинам графа на рисунке 12. Полученный граф, представленный на рисунке 15, соответствует лучшему размещению вершин графа с точки зрения меньшей суммарной длины соединений.

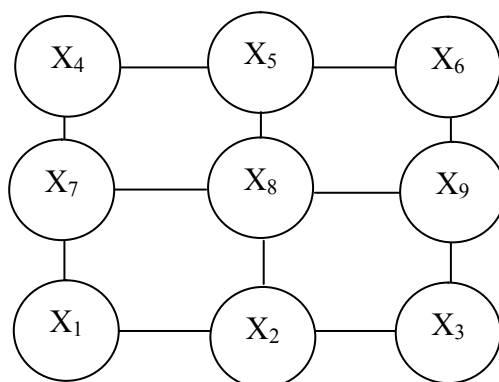


Рисунок 15

Применение факторизации позволяет уменьшить общую длину соединений L_G дополнительно на (5-7)%.

4.3 Системы автоматизированного проектирования

Сложность и трудоемкость проектируемых изделий, сжатые сроки проектирования, высокие требования к качеству разработок требуют новых совершенных технологий и систем для выполнения задач проектирования.

Это прежде всего комплексное применение компьютеров, современных математических методов в проектировании, на базе которых возможно создание Систем Автоматизированного Проектирования (САПР).

Применение САПР особенно эффективно при создании сложных объектов и технических систем.

Общая характеристика процесса проектирования и базовые определения

Проектирование – это не только создание идеи построения объекта, но и обоснование способа его реализации.

Проектирование можно рассматривать как целенаправленную последовательность актов принятия проектных решений, приводящих к построению описания проектируемого объекта с заданной степенью детализации.

Некоторые особенности проектной деятельности:

- 1) продуктом проектирования является упорядоченная совокупность сведений, служащих знаковой моделью объекта, реально еще не существующей в момент проектирования;
- 2) нестабильность и неопределенность постановки задачи, ее изменчивость до завершения процесса проектирования;
- 3) проектирование, как правило, носит итерационный многовариантный характер.

Процесс проектирования реализуется в соответствии с определенным планом, который принято представлять в виде логической схемы (логического графа) построения проекта, на которой видна очередность проектных процедур. Под проектной процедурой понимается элементарное действие, совершаемое конструктором в процессе проектирования.

Стратегия проектирования – это поиск и определение последовательности операций с целью решения поставленной задачи. Стратегия подразумевает использование методов и алгоритмов проектирования.

Технология проектирования – это опробованная последовательность действий или операций, позволяющая технически выполнить проектирование заданного объекта.

При проектировании сложных объектов применяется метод **декомпозиции**, т.е. разбиения.

Уровни декомпозиции:

- **системный** – наиболее общее описание назначения объекта и его связей;
- **архитектурный** – описание структуры объекта;
- **функциональный** – описание законов функционирования подсистем объекта;

- **конструктивный** – детальный выбор и описание всех элементов системы (объекта).

В составе САПР можно выделить следующие основные подсистемы:

- **информационная подсистема**, предназначена для сбора, хранения, поиска, упорядочения, пополнения и выдачи всей необходимой для обеспечения процесса проектирования информации;
- **подсистема поиска решений технической задачи обеспечивает:**
 - анализ проектной ситуации и определение направления поиска решения;
 - поиск решения задачи;
 - форматирование критерия оценки вариантов;
 - выбор рационального или оптимального варианта;
- **подсистема инженерного анализа** (моделирование объекта и оптимизация его характеристик). Ее назначение:
 - выполнение всех вычислительных работ, связанных с детализацией выбранного варианта решения проектной задачи.

• **Подсистема ведения и изготовления документации.**

Назначение – обеспечение получения проектных документов (чертежей, технических описаний, схем, графиков, таблиц и т. пр.), необходимых для создания объекта проектирования. (Это составляет обычно 45-60% всех затрат. В тоже время это нетворческая, рутинная, однообразная работа). Автоматизация этого этапа дает наибольший эффект.

Автоматизация проектирования технических систем особенно эффективна, когда от автоматизации выполнения отдельных инженерных расчетов переходят к комплексной автоматизации, создавая для этой цели САПР. Примером подобной САПР может служить интегрированная САПР PCAD для автоматизации проектирования и подготовки производства печатных плат. Она позволяет выполнить операции по созданию библиотек компонентов; графического ввода чертежей принципиальной схемы и чертежа печатной платы проектируемого устройства; проектирование программируемых логических матриц; выпуск конструкторской документации и технологической информации на проектируемую печатную плату и многое другое.

Перечислим важнейшие, технические требования, которым должна удовлетворять современная САПР печатных плат, в порядке их значимости:

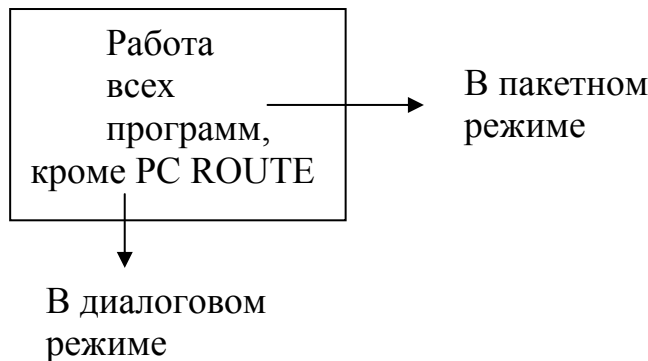
- Полная русификация системы, включая диалоговые окна, меню, подсказки, системные сообщения, руководство пользователя.
- Поддержка системой сквозного проектирования – от создания принципиальной схемы и трассировки платы до подготовки выходных данных (фотошаблонов, программ для сверлильных и сборочных автоматов) в реальном времени и в рамках единого проекта.

- Возможность адаптации к внедренной на предприятии технологии проектирования и производства печатных плат и устройств.
- Наличие импортных и отечественных баз данных элементов.
- Наличие функций автоматизированного размещения элементов и автоматической трассировки печатной платы.
- Наличие интерфейса с имеющимся оборудованием.
- Автоматизированный выпуск комплекта конструкторской документации в полном соответствии с ЕСКД.
- Возможности импорта/экспорта с другими аналогичными системами (в том числе – системами для моделирования печатных плат) через перечень элементов и список цепей.

Большинство современных САПР в той или иной степени удовлетворяют вышеперечисленным требованиям. Поэтому следующее требование, которое прямо или косвенно предъявляется к САПР печатных плат – «экономическое», учитывающее соотношение «цена/возможности». Интегрированная САПР PCAD 4.5 - это наиболее популярная в мире система автоматизации проектирования и подготовки производства печатных плат. Она позволяет выполнить операции: по созданию библиотек компонентов; графический ввод чертежа принципиальной схемы и чертежа печатной платы проектируемого устройства; выпуск конструкторской документации и технологической информации на проектируемую печатную плату.

5 МЕТОДИЧЕСКИЕ УКАЗАНИЯ ПО ВЫПОЛНЕНИЮ КОНТРОЛЬНОЙ РАБОТЫ ПО САПР PCAD 4.5

5.1 Запуск программ в САПР PCAD 4.5



Программы могут запускаться непосредственно из среды MS DOS или под управлением специальной программы PCSHELL. PCSHELL – это универсальный монитор.

Для вызова программы из среды MS-DOS нужно набрать имя программы и нажать [Enter]. В этом случае программа будет вызвана в диалоговом режиме и содержимое экрана сотрется, появится фирменная заставка САПР PCAD с названием программы и в диалоговом режиме будут запрошены имя входного и выходного файлов.

Если запуск программы выполняется в виде:

>PCNODES TEST [ENTER],

то программа будет вызвана в «пакетном режиме», т.е. «PCNODES» поймет, что имя входного файла TEST.SCH; имя выходного файла TEST.NLT. В этом режиме стирание экрана не происходит и выводится служебная информация о работе программы.

Программы PCAD: PCCAPS, .. RDS, PC-PLACE; PCPRINT; ... PLOTS; PC-PACK – имеют открытое меню, где можно задавать конфигурацию программы.

Примеры:

>PCCAPS h – ключ.

- h – подсказка о правильности ввода программы в пакетном режиме.

- r – запуск программы без открытого меню.

Вызов программ САПР PCAD при помощи монитора PC-SHELL.

Запуск монитора:

набрать PCAD <Enter> высвечивается главное системное меню, в том числе:

- подсистема задания исходной информации;

- подсистема проектирования топологии печатной платы.
- D* - подсистема задания исходной информации.
- P* - подсистема проектирования топологии печатной платы.
- C* - подсистема корректирования топологии печатной платы.
- S* - моделирование.
- I* - системные интерфейсы.
- U* - пользовательское меню.
- C: - редактор библиотечных элементов.
- S: - проектирование схем электрических принципиальных.
- N: - выделение списка цепей.
- L: - объединение многолиствого проекта.
- R: - электрический контроль.
- P* - подсистема связи с печатной платой.
- K: - трансляция FIL – файла.
- P: - упаковка.
- B: - корректировка схемы электрической принципиальной.

Здесь обозначения:

«*» – будет вызвано подменю (меню нижнего уровня);

«:» – будет вызвана программа.

Можно вызов программ осуществить нажатием клавиш: например, [D] и [S] это сразу вызовет редактор PCCAPS.

Еще можно вызывать через «/»:

/DS – PC CAPS;

/DN – PC NODES;

/PR – PC ROUTE;

/PL – PC CARDS;

/PP – PC PLACE

и т.д. (всего №№ 17).

Из монитора PC SHELL все программы вызываются с ключом /r, т.е. без открытого меню.

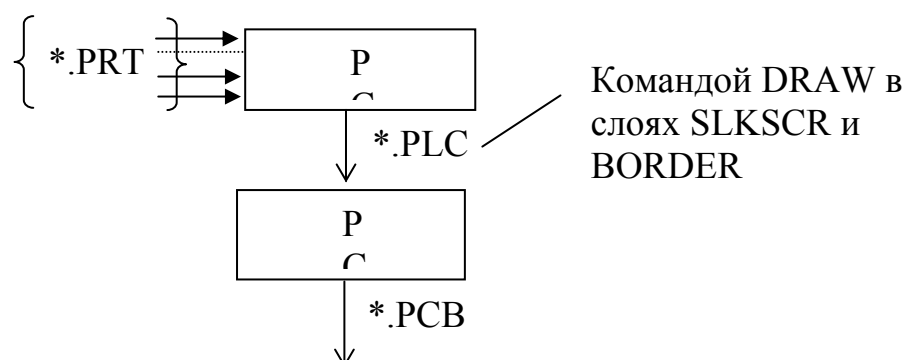
Для вызова с открытым меню нажать [F7]+имя программы+[Enter].

5.2. Способы проектирования и подготовки производства печатных плат

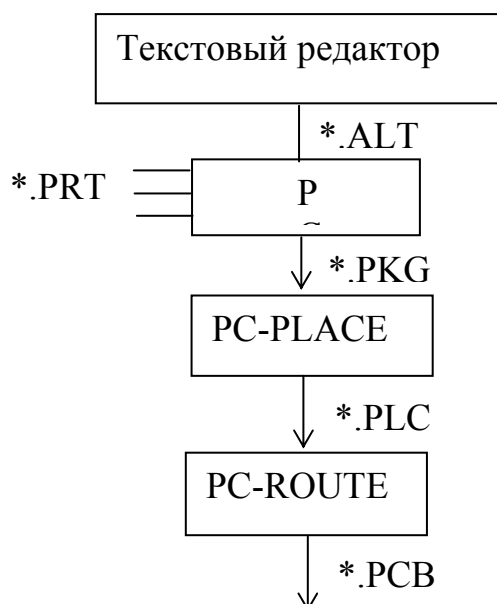
Существует три способа проектирования и подготовки производства печатных плат (ПП).

Первый способ. Предназначен для проектирования наиболее простых плат, для описания которых нет необходимости создавать схему электрическую принципиальную. Основная подготовка выполняется в PC-CARDS. Изображается печатная плата средствами PCCARDS, на которой в ручном режиме размещаются элементы посадочных мест из технологической библиотеки файлов *PRT/.

Списки цепей связности формируются непосредственным указанием на контактные выводы компонентов. Полученное описание может обрабатываться PC-ROUTE.



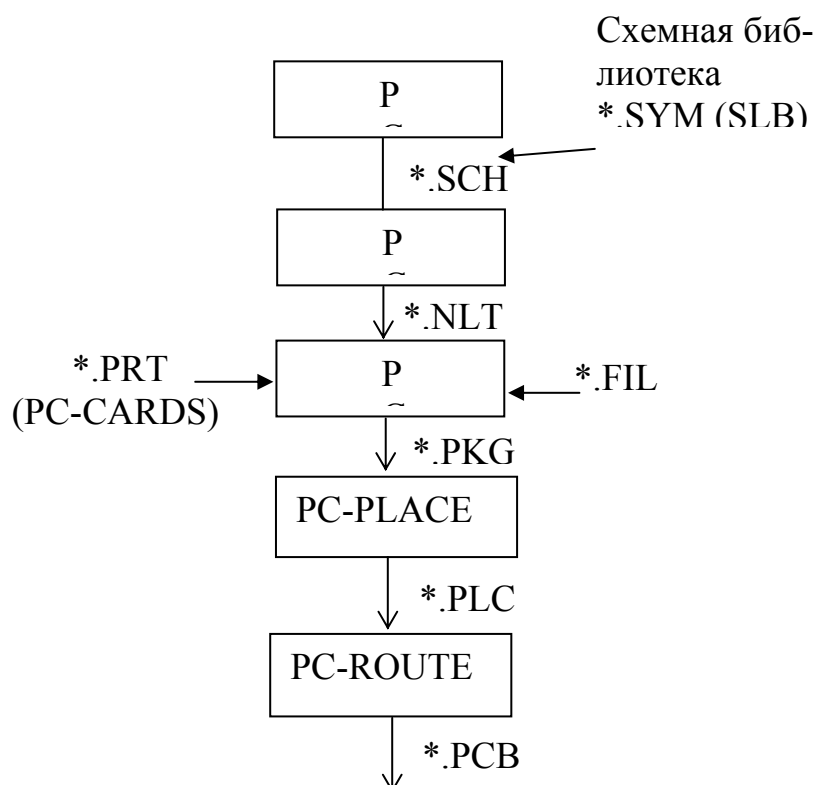
Второй способ. Второй способ предназначен для текстового описания цепей связности и компонентов. Этот способ целесообразно использовать для этапа подготовки производства печатной платы, когда схема электрическая принципиальная окончательно завершена и конструктор - пользователь системы PCAD, не может вносить в нее изменения. В этом случае средствами текстового редактора создается файл типа *.ALT, который обрабатывается программой PC-NLT, и полученный файл типа *.PKG может быть обработан программой PC-PLACE для размещения компонентов на печатной плате и затем программой автоматической трассировки PC-ROUTE.



*.ALT – файл списка цепей.

*.PRT – содержит библиотечные элементы посадочных мест на печатной плате (файлы *.PRT).

Третий способ. Третий способ заключается в создании графического описания электрической схемы средствами графического редактора PC-CAPS. Этот способ является наиболее полным из трех анализируемых способов и может включать в этапы проектирования такие фазы, как логическое моделирование схемы электрической принципиальной, иерархическое проектирование и автоматическую упаковку логических элементов и микросхем. В процессе проектирования выпускается схема принципиальная электрическая. Этот документ становится основой для проекта и все последующие этапы прохождения проекта, включая топологию печатной платы, могут быть автоматически сверены со схемой с диагностикой возможных ошибок.



Алгоритм работы с пакетом PCAD (третий способ).

Создание схемы электрической принципиальной - программа PC-CAPS, режим DETL. Результат – файл *.SCH.

Выделение списка цепей из схемы электрической принципиальной → PCNODES.

Создание библиотеки описаний элементов, необходимых данной электрической схеме:

- любым текстовым редактором создается файл перекрестных ссылок → *.FIL;

- программа PREPACK преобразует текстовый файл .FIL в двоичный файл с расширением .LIB.

Подготовка конструктива печатной платы, ручная расстановка корпусов PDK→PCCARDS, режим DETL. Результат – файл с расширением .PCB.

Упаковка данных об электрической принципиальной схеме и конструктива печатной платы. Выполняет программа PCPACK. Результат – файл с расширением .PKG и файл ошибок PCPACK.ERR, который можно смотреть в текстовом редакторе.

Оптимизация размещения корпусов РЭК по площади печатной платы – выполняет PCPLACE.

Трассировка печатной платы – PCROUTE. Результат работы – модифицированный файл .PCB (конструктив печатной платы + корпуса РЭК + печатные дорожки). Файл .REP – сведения о не разведенных связях и ошибках.

Преобразование файла .PCB в файл .PLT, пригодный для рисования печатной платы на принтере или плоттере выполняет программа PCCARDS, режим DETL\SYS\PLOT.

Рисование сформированного изображения печатной платы (.PLT) выполняет программа PC-PRINT или PC-PLOT.

Алгоритм проектирования радиоэлектронной аппаратуры (РЭА) с помощью пакета PCAD, включающий весь цикл работ, можно представить в виде следующей последовательности:

1. Создание или корректировка схемных (символьных) библиотечных элементов РЭА для принципиальных электрических схем (программа PC-CAPS).
2. Создание или корректировка конструкторско-технологической библиотеки (программа PC-CARDS).
3. Ввод принципиальной электрической схемы (программа PC-CAPS).
4. Выделение списка цепей из принципиальной электрической схемы (программа PC-NODES).
5. Создание библиотеки соответствия используемых компонентов (программа PREPACK).
6. Подготовка конструктива ПП (программа PC-CARDS).
7. Автоматическая упаковка логических элементов в микросхемы (программа PC-PACK).
8. Автоматическое или автоматизированное размещение компонентов РЭА по полю конструктива ПП (программа PC-PLACE).
9. Автоматическая трассировка соединений ПП (программа PC-ROUT).

10. Выпуск конструкторской документации и управляющих технологических программ для станков с числовым программным управлением.

В зависимости от специфики реализации конкретных проектов некоторые этапы работ могут быть исключены из общего алгоритма проектирования.

В качестве примера будем рассматривать этап разработки платы простейшего логического автомата, схема которого приведена в приложении А.

5.3 Проектирование печатной платы по ее топологии

Проектирование печатной платы по ее топологии наиболее подходит к данной логической схеме, т.к. она не содержит большого количества элементов и связей между ними. Разработка упрощается в виду использования в схеме элементов одного типа и серии. Вся схема электрическая принципиальная упаковывается в 4-х корпусах - по корпусу на логический элемент каждого типа. Поэтому для разработки печатной платы нам понадобится лишь 4 микросхемы (таблица 2).

Таблица 2

Ло логический элемент	Ко количество	Ко корпусов
К 1 К155 ЛН1	4 4	1 1
К 1 К155 ЛЕ1	3 3	1 1
К 1 К155 ЛА 3	2 2	1 1
К 1 К155 ЛЛ 1	2 2	1

Размещение будем производить в ручном режиме, исходя из критерия минимальной суммы длин связей между элементами схемы. Все действия по подготовке печатной платы проводятся в графическом редакторе PC - CARDS.

Наберите в командной строке pccards и нажмите клавишу Enter или, если Вы находитесь в интегрированной оболочке PC - SHELL, последовательно выберите в предложенном меню пункты с заглавными буквами Р и L.

Подготовьте графический редактор к работе. Для этого после загрузки программы PC - CARDS выберите пункт Configure PC-CARDS (Задание конфигурации редактора PC-CARDS) главного меню. Выбор того или иного пункта меню осуществляется нажатием клавиш «Стрелка вверх», «Стрелка вниз» или клавишей Space bar. Выбор режима осуществляется после нажатия клавиши Enter.

В конфигурации редактора следует указать имя файла - проекта, который будет автоматически загружаться при входе в редактор, имя биб-

лиотечного файла или нескольких библиотечных файлов, информация которых используется в данном графическом редакторе. Эту информацию необходимо внести в строку Library filename (Имя библиотеки).

В нашей схеме понадобятся файлы, содержащие графическую информацию о корпусах микросхем серии K155 и технологическая информация о разъеме. Файлы с технологической информацией имеют расширение prt и обычно содержатся в каталоге PRT или plb - в каталоге PLB. Файлы с расширением plb представляют собой существенно сжатые и распределенные по типам файлы *.prt. Эти файлы формируются и обслуживаются в программе - утилите PC - LIB.

Возможные варианты задания библиотеки:

*.PRT - заданы все файлы PRT, находящиеся в текущем директории C:\PCAD\PLB\155.plb или *.plb, находящаяся в директории C:\PCAD\PLB.

После установки соответствующих параметров нажмите клавишу Esc. Система выдаст запрос о сохранении текущей конфигурации. Ответьте YES и нажмите клавишу Enter. Конфигурация программы будет сохранена в файле pccards.cfg, задаваемым по умолчанию.

Для запуска режима редактирования редактора выберите пункт Edit database в главном меню.

Выбор той или иной команды в меню редактора осуществляется подведением выделенной строки к требуемой команде и нажатием клавиши Space bar или левой кнопки мыши.

В редакторе установите шаг координатной сетки 50 : 50. Для включения/выключения масштабной сетки используйте функциональную клавишу F7.

С помощью команды VLYR установите слои:

SLKSCR - обозначения габаритных размеров платы и компонент схемы;

BRDOUT - обозначения области трассировки;

DEVICE - обозначения имен компонент схемы;

PIN - обозначения контактных площадок элементов схемы;

COMP - 1 - го слоя трассировки;

SOLDER - 2 - го слоя трассировки;

ATTR - видимых атрибутов;

\$CONT - неразведенных связей;

Вышеперечисленные слои необходимо установить в положение ABL, причем, слой BRDOUT сделать активным, т.е. установить его в состояние ABL A. Остальные слои должны быть выключены (Off). Выберите Quit или нажмите правую кнопку мыши.

Для формирования образа печатной платы убедитесь, что активным стал слой SOLDER, затем командой DRAW/LINE или DRAW/RECT вычертите в окне редактора прямоугольник или многоугольник, характеризующий габаритные размеры печатной платы.

Для формирования области трассировки установите слой BRDOUT и командой DRAW/LINE или DRAW/RECT вычертите прямоугольник или многоугольник, внутри которого будет разрешена трассировка. Кроме того, в образе печатной платы можно указать границы областей, запретных для прокладки трасс печатных проводников. Области запрета обозначаются замкнутыми ломанными линиями нулевой толщины ($W=0$), выполненными в определенных слоях рисунка печатной платы. Выбор того или иного активного слоя осуществляется подведением указателя к имени слоя, расположенному в левом нижнем углу экрана и нажатием левой кнопки мыши. Нажимайте кнопку мыши, пока не установите требуемый слой.

Для ввода корпусов элементов на печатную плату воспользуйтесь командой ENTR/COMP. При этом система в нижней строке экрана попросит ввести требуемый элемент из технологической библиотеки, указанной в конфигурации редактора, или нажать клавишу F1 для выбора нужного элемента из списка компонент, высвечиваемом в правой части экрана вместо командного меню.

Если список компонент полностью не помещается на экране, то его можно пролистать, используя опции PgUp (вверх) или PgDn (вниз). Выбор того или иного элемента списка производится аналогично выбору команды меню редактора. Выбор элемента файла

PLB осуществляется после открытия последнего. После чего можно выбрать требуемый PRT- файл.

При выборе элемента списка система выдает запрос:

New layers may be added. Continue? Yes No

(Добавлять новый элемент? Да Нет)

Выберите Yes и нажмите клавишу Space bar или левую кнопку мыши.

Далее система спросит:

Converting comp to the English Unit. Continue? Yes No

(Конвертировать элемент в дюймовую систему? Да Нет)

Выберите Yes и нажмите клавишу Space bar или левую кнопку мыши.

После чего, система переходит в режим динамического отслеживания расположения элемента, т.е. пользователю предоставляется возможность «отбуксировать» элемент в определенное для него на печатной плате место. Элемент как бы оказывается привязанным к указателю мыши и при ее перемещении сдвигается по экрану в соответствующем направлении. Чтобы зафиксировать элемент нажмите клавишу Space bar или левую кнопку мыши.

Таким образом, для размещения корпусов микросхем на печатной плате необходимо проделать следующие действия:

- 1) активизируйте команду ENTR/COMP;
- 2) на запрос системы нажмите клавишу F1;

3) в появившемся списке выберите элемент 155LN1 и нажмите левую кнопку мыши;

4) поместите его с левой стороны печатной платы и нажмите левую кнопку мыши;

5) нажмите правую кнопку мыши для возвращения в список элементов;

6) выполните пункты 3 - 5 для размещения остальных корпусов микросхем данной схемы;

7) в списке найдите файл adapters.plb и выберите в нем файл r2hn-15.prt, который содержит технологическое описание разъема, разместите его в верхней части платы так как это показано в приложении Б.

Для выхода из списка активизируйте опцию Exit List (выход из списка) или нажмите правую кнопку мыши.

Не отчаивайтесь, если Вы неудачно разместили элементы на печатной плате. Чтобы исправить положение воспользуйтесь командами редактирования, предоставляемые редактором:

CLYR - изменить слой объекта;

MOVE - переместить элемент в требуемое положение;

ROT - поворачивает объект на 90 град. против часовой стрелки при нажатии на левую кнопку мыши, нажатие правой кнопки мыши завершает действие команды;

DEL - удалить элемент;

COPY - скопировать выбранный элемент.

Каждая из команд позволяет работать с несколькими объектами одновременно. Для этого необходимо воспользоваться опцией WIN, которая становится доступной при активизации команд редактирования. Чтобы выбрать элементы в окно система запрашивает координаты диаметрально противоположных углов рамки. Или с помощью указателя мыши в режиме динамического отслеживания определите координаты левого верхнего и правого нижнего угла окна. Причем, элемент выбирается, даже если он полностью не лежит в области ограниченной рамкой. Схема электрическая принципиальная имеет 5 сигнальных выводов и 2 шины: питания и «земля», которые система добавляет автоматически для элементов, имеющих соответствующие контакты.

Обратите внимание на то, чтобы все компоненты печатной платы размещались внутри области, ограниченной разрешенным для трассировки многоугольника. Если окажется, что отдельные компоненты или их части будут расположены за ее пределами, передвиньте их внутрь области, используя команду MOVE.

После размещения элементов на печатной плате можно приступить к назначению связей между ними. Для этого в экранном меню активизируйте команду ENTR/RATN. Система попросит ввести имя первого контакта, который будет являться начальной точкой линии связи. Выбор

контакта, выделенного синим цветом, производится подведением к нему указателя и нажатием левой кнопки мыши. Для ввода конечной точки линии связи подведите указатель к другому контакту и нажмите левую кнопку мыши. Чтобы не ошибиться в выборе контакта микросхемы знайте, что на контакт с номером 1 указывает метка в виде точки или треугольника на корпусе элемента (рис. 16). Остальные контакты нумеруются против часовой стрелки относительно 1-го контакта.

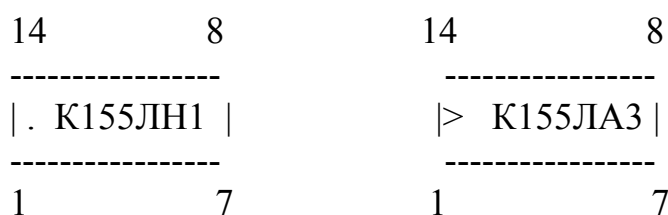


Рисунок 16

Используемые микросхемы серии K155 имеют следующие спецификации:

Контакт	Назначение
7	« земля »
14	+ 5V

Для задания связей необходимо знать какой контакт микросхемы соответствует какому вентилю этой микросхемы. Микросхема K155ЛН1 имеет 6 вентилях на корпус, остальные микросхемы, используемые в схеме имеют по 4 вентиля на корпус. Информация о соответствии контактов вентилю приведена в любом справочнике по интегральным микросхемам.

Для проведения связей руководствуйтесь следующим:

1) активизируйте команду ENTR/RATN;
 2) защелкните указатель мыши на 1-м контакте микросхемы K155ЛН1;

3) защелкните указатель на 5-м контакте микросхемы K155ЛЕ1;
 4) переместите указатель на ближайший контакт разъема и нажмите левую кнопку мыши;

5) нажмите правую кнопку мыши для завершения ввода связи. Не пугайтесь если связь, которую Вы только что провели вдруг исчезла с экрана. Ее можно вновь увидеть активизировав опцию R, расположенную в нижней строке экрана, т.е. сделать букву R зеленой (R выключена, если выделена красным цветом);

б) используя приложение А, проведите остальные связи. Шины питания и « земли » система формирует автоматически. Необходимо лишь соединить их с выбранными контактами разъема.

Система PCAD обладает свойством, по которому при перемещении соединенных элементов связи между ними не разрываются, т.е. при необходимости можно свободно перемещать элементы по печатной плате, не заботясь о нарушении связей между ними.

После перемещения воспользуйтесь командой REDR - обновления экрана.

Для записи схемы на диск активизируйте команду FILE/SAVE. В приглашении редактора введите имя файла схемы (auto1.plc) и нажмите клавишу Enter. Файл будет помещен в каталог, имя которого указано в конфигурации редактора.

Выход из редактора осуществляется с помощью команды SYS/QUIT, а из программы - путем выбора пункта Exit PC-CARDS главного меню.

Следующим шагом разработки печатной платы является процесс трассировки, т.е. разводки проводников по заданным правилам. Для этих целей в САПР PCAD используется программа PC - ROUTE. Программа PC - ROUTE предназначена для выполнения автоматического процесса трассировки двусторонних и многослойных печатных плат. Особенностью программы являются широкие возможности в предварительной настройке параметров и правил трассировки на конкретный проектируемый узел.

Входными данными для программы PC - ROUTE является файл проекта, в котором задано размещение элементов внутри конструктива печатной платы и заданы связи между элементами. Таким файлом является файл auto1.plc.

Для работы автотрассировщика необходим также файл, содержащий сведения о параметрах и правилах трассировки. Этот файл называют файлом стратегии трассировки. Файлы стратегии имеют расширение STL.

Выходными данными программы PC - ROUTE являются следующие файлы:

- файл базы данных, дополненный графической информацией о проложенных трассах печатных проводников, имеющий расширение PCB;
- файл сообщений, содержащий протокол трассировки, имеющий расширение REP.

В процессе работы программы создаются также два рабочих файла *.RTS, *.RTE. В результате работы программы не все связи могут оказаться разведенными. Ручная допроводка, а также корректировка отдельных связей (редактирование топологии печатной платы) может быть осуществлена пользователем средствами программы PC - CARDS. PC - ROUTE помечает связи, оставшиеся неразведенными. Информация о них содержится в следующих файлах:

- графическая - в слое \$CONT файла проекта *.PCB;

- текстовая - в файле сообщений *.REP.

Для запуска программы в командной строке введите `pcroute` и нажмите клавишу `Enter` или, если Вы находитесь в оболочке PC - SHELL, в меню последовательно выберите пункты с заглавными буквами P и R.

После запуска программы PC - ROUTE на экране дисплея появится основное меню программы. Данное меню содержит три поля. Верхнее поле определяет:

Enter database name: - имя проекта.

(Database reference name) - ссылочное имя проекта, используется.

для регистрации проекта в каталоге программы PC - ROUTE.

С помощью клавиши HOME перейдите в это поле и введите имя файла `auto1.plc`.

Select routing strategy - выберите стратегию трассировки. Просмотр имен файлов стратегии трассировки осуществляется нажатием клавиши `Space bar`.

Файл стратегии трассировки содержит информацию о количестве слоев, алгоритме трассировки, порядке трассировки и другие технологические параметры. Очень важно, чтобы в файле стратегии трассировки были указаны те же самые единицы измерения, которые использовались при построении образа печатной платы (MILS - дюймовая система; MM - метрическая система). Для задания единиц измерения в графическом редакторе PC - CARDS активизируйте команду SCMP/UNIT.

Если не заданы значения хотя бы одной контактной площадки, то трассировщик завершает работу с соответствующим сообщением об ошибке.

Среднее поле позволяет выбрать следующие варианты дальнейшей работы:

EDIT ROUTING STRATEGY - редактирование стратегии трассировки;

ROUTE - запуск автотрассировщика;

EXIT PC-ROUTE - возврат в DOS.

В 1-м разделе задайте следующие параметры трассировки:

ROUTING GRID X : 50 Y : 50 - шаг сетки по осям X, Y.

NUMBER OF ROUTING LAYERS: 2 - количество слоев печатной платы; ROUTE TYPE: STEINER - MINVIA - тип алгоритма трассировки. Алгоритм STEINER - MINVIA позволяет минимизировать количество переходных отверстий.

ROUTE ORDER: LONG - SHORT - задание порядка трассировки связей в зависимости от их длин. Переключение между вариантами производится с помощью клавиши `Space bar`. При выборе варианта LONG - SHORT связи трассируются в порядке убывания их длин, а при варианте SHORT - LONG - в порядке возрастания их длин. Для варианта SHORT - LONG число не разведенных связей меньше, но ручная допроводка со-

пряжена с большими трудностями, чем при использовании варианта LONG - SHORT.

PERFORM VIA MINIMIZATION: YES - выполнять процедуру минимизации числа переходных отверстий.

ORIENTATION PREDOMINANT COMPONENT: VERTICAL - задание основной ориентации микросхем (вертикальная). Данный параметр позволяет повысить качество и процент трассировки.

PIN TYPE: 24 - тип контакта (штыревой).

Значение 0 зарезервировано для сквозных переходных отверстий, с 1 по 24 - для штыревых контактов, с 25 по 50 - для планарных контактов, с 51 по 100 - для межслойных контактов переходных отверстий и с 101 по 255 - для контактных площадок пользователя.

Чтобы выйти из раздела редактирования стратегии трассировки в основное меню PC -ROUTE, следует нажать клавишу Esc. Предлагаются следующие варианты дальнейшей работы с отредактированной стратегией:

Save under current name - сохранить отредактированный вариант стратегии под старым именем;

Save under new name - сохранить отредактированный вариант стратегии под новым именем;

Do not use or save - отредактированный вариант стратегии не использовать и не сохранять.

Выбор варианта производится с помощью клавиши Space bar. Выберите 2-й вариант и введите имя PLATA.

В пункте Select routing strategy главного меню выберите стратегию PLATA.CTL и нажмите клавишу Enter.

Нижнее поле определяет вызов служебных программ утилит трассировщика. Для перехода между полями используются клавиши HOME и END.

Для запуска автотрассировщика выберите пункт ROUTE главного меню.

Прежде чем запустить автотрассировщик в работу, проверьте и установите значения следующих параметров данного меню:

EXTRACT DATA - параметр, определяющий необходимость считывания из файла данных проекта. Параметр может принимать значения:

YES - если выполняется впервые выход на трассировку с данным проектом или были изменения проекта;

NO - если ранее уже была работа с проектом. Выбор значения параметра производится с помощью клавиши Space bar.

ROUTE - параметр, определяющий вариант трассировки: первый запуск, повторный запуск (рестарт), отказ от трассировки. Параметр может принимать значения:

NEW - если трассировку следует начать сначала;

RESTART - если был прерван процесс трассировки и требуется его возобновить;

NO - если необходимо произвести только запись результатов трассировки проекта без запуска автотрассировщика в файле *.PCB. Выбор значения параметра производится с помощью клавиши Space bar.

CREATE ROUTED DATABASE - параметр, определяющий необходимость создания выходного файла проекта (файл *.PCB) после окончания трассировки. Параметр может принимать значения:

YES - если по окончании трассировки необходимо сформировать выходной файл проекта;

NO - если необходимости в создании выходного файла проекта нет. Выбор значения параметра производится с помощью клавиши Space bar.

DATABASE NAME - параметр, определяющий имя выходного файла проекта.

Если Вас удовлетворяет предложенное системой имя выходного файла проекта (auto1.pcb), нажмите клавишу Enter. В противном случае ведите с клавиатуры новое имя.

Запуск автотрассировщика производится путем перемещения курсора на строку START с последующим нажатием клавиши Enter.

Вся последующая работа программы происходит в автоматическом режиме. По завершению трассировки система сообщит:

ROUTE COMPLETE (CR)

Для завершения работы нажмите любую клавишу.

Если произошло прерывание работы программы, система сообщит о причине ошибки в файле сообщений (auto1.rep).

При запуске автотрассировщика на экране высвечивается следующая информация:

DATABASE - задает имя проекта;

STRATEGY - показывает имя файла стратегии;

TOTAL LAYERS - общее число слоев трассировки;

NUMBER OF PASSES - число проходов трассировщика;

TOTAL SUBNETS - задает число подцепей проекта;

ELAPSED - требуемое время работы автотрассировщика;

ROUTED SUBS - число уже проложенных цепей;

% COMPLETED - процент проложенных трасс к общему числу цепей;

ROUTE LAYERS - текущая пара слоев, в которых прокладывается трасса;

ALGORITHM - алгоритм выполняемого в данный момент этапа;

CURRENT SUBNET - номер текущей подцепи, которая обрабатывается в данный момент;

KEYBOARD STATUS - определяет горячие клавиши.

Клавиша Esc вызывает меню, клавиша «+» изменяет выводимую на экран информацию.

6 P-CAD 2000 [ACCEL EDA 15.0]

До недавнего времени система P-CAD версий 4.5,6,8 являлась наиболее популярной системой разработки ПП. Все эти версии программы были сделаны под операционную систему DOS. Система P-CAD для Windows выпущена в 1996 году и в связи со сменой владельца получила название ACCEL EDA версии 12. Новая версия ACCEL EDA 15.0 вышла в конце лета 1999 года и получила название P-CAD 2000. Эта система объединяет лучшие качества систем ACCEL EDA и P-CAD. Она включает в себя программы для графического ввода схем, упаковки схем на ПП, ручного размещения компонентов, ручной, интерактивной и/или автоматической трассировки проводников, контроля ошибок в схеме и ПП и выпуска документации.

В этой системе за счет перехода на 32-разрядную арифметику увеличена разрешающая способность редактора ПП и повышена точность на два порядка. Также новая версия P-CAD по сравнению с более ранними версиями имеет следующие преимущества:

- Выполнение стандартных команд и организация меню в стиле Windows упрощает работу с системой и ознакомление с ней.
- Выводы результатов проектирования на периферийные устройства (принтеры, плоттеры), имеющие драйверы для Windows.
- Применение векторных шрифтов позволяет наносить на схемы и ПП надписи по-русски.
- Возможность задания разных типов переходных отверстий при переходе проводников со слоя на слой.
- Возможна автоматическая трассировка одной и той же цепи сегментами разной ширины.
- Усовершенствованы алгоритмы автотрассировки проводников, включая автоматическую трассировку многослойных ПП, имеющих внутренние слои металлизации.

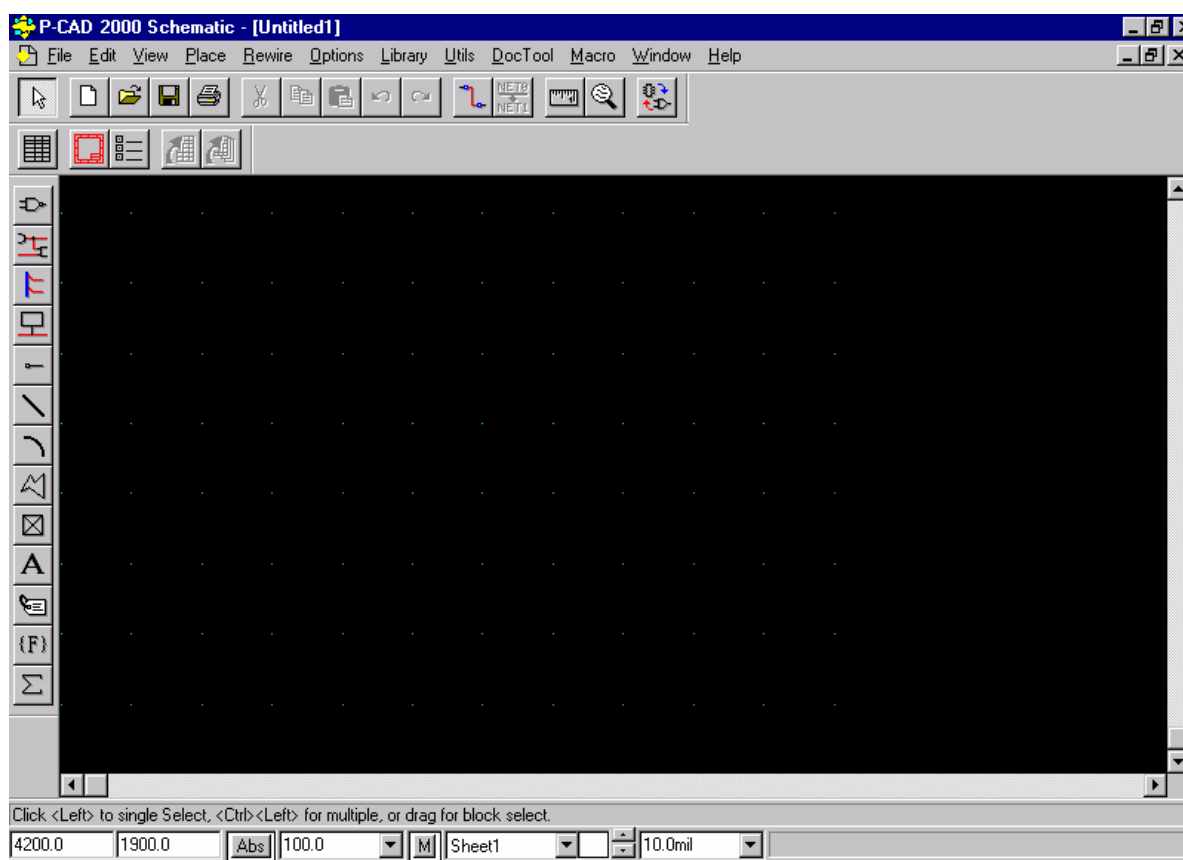
Поддержка форматов описания данных позволяет обмениваться информацией с такими системами как AutoCAD, OrCAD, Pspice и P-CAD более ранних версий.

Система **ACCEL EDA** состоит из четырех основных модулей **ACCEL Library Manager** - менеджер библиотек, **ACCEL Schematic** - редактор схем, **ACCEL PCB** - редактор ПП, **ACCEL Autorouters** - автотрассировщик, и ряда вспомогательных программ.

Установка пакета P-CAD 2000

Установка пакета производится запуском программы **Setup** из каталога **P-CAD\Setup**.

После запуска программы на экране появляется диалоговые окна, в которых нужно ответить на некоторые вопросы, касающиеся установки пакета. На первом экране программы надо нажать клавишу **Next**, на следующем **Yes**. На третьем экране надо ввести в первое поле Ваше имя, а во второе имя вашей организации, и в поле **Product Type** надо выбрать верхний элемент. На четвертом экране предлагается ввести серийные номера продуктов, надо ввести номер **X5YB-2SVT-X8VG-7CW8** во все три поля и нажать клавишу **Next**. В этом окне необходимо выбрать каталог куда вы хотите установить **P-CAD 2000**, по умолчанию это каталог **C:\Program Files\P-CAD 2000**. После выбора каталога необходимо нажать клавишу **Next**, и в следующем окне необходимо выбрать тип установки: **Typical** – установка по умолчанию (рекомендованная установка), **Custom** – установка с возможностью выбора элементов пакета P-CAD для установки (рекомендуется для опытных пользователей). Выбираем **Typical** и нажимаем **Next**. Далее он спрашивает имя папки которая будет размещаться в меню Программы, пишем **P-CAD 2000** и жмем **Next**. Далее снова **Next** и ждем когда программа установки закончит свою работу, это может занять несколько минут. После на появившемся окне нажимаем **Finish**, и установка пакета закончена, можно приступить к работе с ним.



Графический редактор схем ACCEL Schematic

Графический редактор схем **ACCEL Schematic** (далее **Schematic**) один из основных модулей системы **ACCEL EDA**.

1 Общие сведения о редакторе схем Schematic

Графический редактор ПП **Schematic** запускается путем выбора пункта **Schematic** в меню **PCAD 2000**, которое в свою очередь находится в меню «Программы».

После запуска редактора на экране появится окно графического редактора схем.

В верхней части окна располагается меню команд, активизируемых щелчком мыши или нажатии клавиши **ALT** в сочетании с подчеркнутой буквой в имени меню.



Последовательно выбираемые меню, команды и подкоманды в дальнейшем будут разделяться знаком «>».

Краткое описание пользовательского интерфейса

Пункт меню **File** используется для создания, открытия, сохранения и печати схем. А также можно генерировать отчет. Пункт меню **Edit** используется работы с буфером обмена, редактирования параметров активного элемента. Меню **View** – предназначено для изменения масштаба изображения, увеличение – уменьшение, расположение в центре и тд. В пункте меню **Place** можно выбирать для расположения на схеме различных элементов, таких как компоненты схем, цепи, шины, линии. **Rewire** - в нем находится единственный пункт меню **Manual** для ручного редактирования сегмента электрической цепи. **Options** - пункт меню, в котором устанавливаются различные параметры, такие параметры как настройка фильтров для выделения объектов внутри или вне выбранной области, настройка конфигурации, определение наборов сеток, установка абсолютной или реальной сеток, и многое другое. Пункт меню **Library** предназначен для работы с библиотеками а именно создание библиотек, назначение псевдонимов символам, копирование членов библиотек, открытие библиотек. **Utils** в этом пункте меню находятся команды вспомогательного назначения. Меню **Macro** - меню по работе с макросами, в которые можно записать часто повторяющиеся действия. В пункте меню **Window** находятся команды по работе с окнами: открытие нового окна, упорядочивание окон и иконок, переход к одному из открытых окон. Через пункт меню **Help** производится доступ к помощи справочной системе пакета.

Две панели инструментов дублируют наиболее часто употребляемые команды из пунктов меню. Горизонтальная панель содержит пиктограммы для системных команд, вертикальная панель для команд размещения элементов.

В центре, самая большая часть - рабочая область, сразу под ней строка сообщений, ниже строка состояния. В ней находятся следующие поля:

- Координаты X Y - текущие координаты курсора, в этих полях можно так же набрать координаты, в которые надо переместить курсор.
- Кнопка переключений сетки: **REL** - относительная сетка может иметь начало координат в любой точке рабочего области, **ABS** - абсолютная сетка имеет начало в левом нижнем углу рабочей области.
- Шаг сетки.
- Кнопка создания макрокоманд.
- Список листов схемы.
- Ширина линии.
- Строка состояния.

Настройка редактора

Прежде чем начинать создание принципиальной схемы следует сделать необходимые настройки конфигурации редактора принципиальных схем **Schematic**.

В меню команд **Options>Configure** можно выбрать размер листа – один из стандартных форматов A4, A3... так же можно самостоятельно задать размеры листа. В разделе **Title Sheets** выбирается основная и дополнительные надписи (рамка). Чертежи этих рамок создаются заранее с помощью редактора и заносятся в файлы с расширением **TTL**. Стандартные рамки занесены в каталог **TITLES** каталога **ACCEL**. Рамки чертежа становятся видны после выбора режима **Display Title Sheets**. Так же можно выбрать систему единиц **mil** – милы (1 мил – 0,001 дюйма), **mm** - миллиметры, **inch** – дюймы (1 дюйм – 25,4мм). Изменить систему единиц можно в любой момент работы со схемой. Так же можно установить и другие параметры.

В меню **Options>Grids** устанавливается шаг сетки. По команде **Add** добавляются новые значения.

В меню **Options>Display** задаются цвета различных объектов, стиль изображения шин и ряд других параметров.

Options>Text Style выбирается шрифт, его размеры и др. свойства текстов устанавливаемых по умолчанию.

Настройки конфигурации **ACCEL Schematic** заносятся в файл **SCH.INI**, и поэтому они сохраняются и при последующей работе с редак-

тором. Но параметры проекта (стили линий, стили шрифта и тд.) заносятся в файлы схемы.

Создание схемы электрической принципиальной

Перед началом нанесения элементов на схему необходимо подключить библиотеки с необходимыми элементами. Это делается по команде **Library>Setup** которая обеспечивает доступ к необходимым библиотекам. Клавишей **Add** производится добавление имен библиотек в список открытых.

Размещение компонентов на схеме начинается по команде **Place>Part**. После чего, щелкнув курсором мыши в любой точке экрана, открывается меню выбора компонента. В поле **Library** выбирают библиотеку, содержащую нужный элемент. Далее из списка компонентов выбирают элемент, при необходимости можно посмотреть его изображение элемента, нажав кнопку **Browse**. Также можно задать один из трех вариантов отображения элемента: **Normal** – нормальное, **DeMorgan** – обозначение логических функций, **IEEE** – в стандарте института инженеров по электронике и электротехнике. После выбора компонента и введения необходимых параметров нажимаем кнопку **OK** – курсор примет форму перекрестия. Непосредственное размещение символа элемента на схеме выполняется после щелчка курсором в любой точке рабочей области окна. До тех пор пока не будет отпущена левая кнопка мыши, символ можно перемещать по экрану, поворачивать, нажатием клавиши **R**, зеркально отображать нажатием – **F**. Повторный щелчок курсором размещает на схеме следующую копию выбранного компонента. В режиме **Edit>Select** щелчок курсором в любой точке внутри контура символа компонента производит его выбор с возможностью его перемещения, поворота и удаления символа в целом.

Следующим этапом создания схемы является размещение электрических цепей осуществляемое по команде **Place>Wire**. Появляется курсор в форме перекрестия, щелчком мыши обозначается начальная точка и точки излома проводников, нажатие правой кнопки мыши завершает ввод цепи. Ширина прокладываемой цепи устанавливается по команде **Options>Current Wire**. Неподсоединенные выводы компонентов и цепей помечаются квадратиками, которые гаснут после выполнения электрического соединения. Если выводы компонентов соприкасаются, то электрическое соединение, сохраняющееся при раздвижении, перемещении компонентов. Включение в цепь дополнительной точки излома для будущего редактирования выполняется по команде **Rewire>Manual**. Электрическое соединение пересекающихся цепей обозначается точкой, которая ставится автоматически на Т-образных соединениях. Чтобы поставить точку соеди-

нения пересекающихся цепей, нужно при вводе второй цепи щелкнуть курсором в точке пересечения и затем продолжить построение.

Размещение шины начинается по команде **Place>Bus**. После этого нажатием левой кнопки мы отмечаем начало и изломы шины, а нажатием правой кнопки мыши заканчиваем построение. Далее по команде **Place>Wire** размещаем цепи, которые должны быть присоединены к шине. Для присоединения цепей к шине необходимо при рисовании цепи один ее конец расположить на шине – это делается щелчком мыши в любом месте шины. При этом автоматически изображается излом под углом в 45° при правильном подсоединении к шине. Имена цепей образующих шину задаются по команде **Place>Port**, которая размещает на схеме специальный элемент – порт. В появившемся диалоге указывается имя цепи, способ отображения порта на схеме. И также имеется возможность установить автоматическое увеличение имени цепи на единицу, для ускорения именования цепей.

Для оформления листов схемы в **P-CAD** предусмотрены специальные команды оболочки. Существуют следующие способы оформления листов:

- Ручной – пользователь сам с помощью линий рисует рамку, размещает текст. Нарисовав рамку ее можно сохранить.
- Использование стандартных рамок поставляемых вместе в системе **P-CAD**. Они находятся в каталоге **Title**.

Загрузить рамки схемы можно следующими способами:

- По команде **File>Open** открывается файл с расширением **TTL**.
- По команде **Options>Configure** – на закладке **Title** выбирается нужный файл рамки, изображение рамки становится видно после установки флажка **Display Border**.

Заполнение полей рамки производится по команде **File>Design Info**. Нередко приходится создавать схемы, которые не размещаются на одном листе. Дополнительные листы схемы создаются по команде **Options>Sheets/Add**, задавая имена листов. Все листы схемы находятся в одном файле, что облегчает перенос схем с компьютера на компьютер. Для того чтобы **P-CAD** знал, что листы связаны, надо к разрыву цепи подсоединить порт с одним выводом, задать ему какое либо имя; и на другом листе к цепи надо подсоединить порт с таким же именем.

КОНТРОЛЬНАЯ РАБОТА № 1

Задача покрытия (ЗП)

Написать математическую модель ЗП, в соответствии с данными своего варианта.

Решить ЗП, применяя алгоритм Селютина.

Вычислить значение целевой функции.

Исходные данные

Вариант №1

Данные структурного анализа схемы:

=

t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8
2	12	9	13	8	5	3	21

Распределение типов элементарных схем по типам ячеек:

$t_j \backslash Y_i$	1	2	3	4	5	6	7	8	C_i
1					1	1		2	6
2		4							4
3	1			2			1		13
4				1	2		2		19
5			2					1	7
6	1				2			1	6

Где t_j – тип элементарной схемы;

Y_i – тип ячейки (модуля);

C_i – стоимость ячейки [в условных единицах].

Вариант №2

Данные структурного анализа схемы:

M=

t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8
21	3	5	8	13	9	12	2

Распределение типов элементарных схем по типам ячеек:

$t_j \backslash Y_i$	1	2	3	4	5	6	7	8	C_i
1						1	1	2	5

2		4						7
3	2		1	1			1	19
4				2	1		2	13
5			2				1	4
6	1				2		1	6

Где t_j – тип элементарной схемы;

Y_i – тип ячейки (модуля);

C_i – стоимость ячейки [в условных единицах].

Вариант №3

Данные структурного анализа схемы:

t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8
22	25	3	12	9	8	13	5

=

Распределение типов элементарных схем по типам ячеек:

$t_j \backslash Y_i$	1	2	3	4	5	6	7	8	C_i
1						1	1	2	18
2		4							19
3	2		1	1					61
4				2	1		2		5
5			2			3		1	7
6	1				2			1	8

Где t_j – тип элементарной схемы;

Y_i – тип ячейки (модуля);

C_i – стоимость ячейки [в условных единицах].

Вариант №4

Данные структурного анализа схемы:

t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8
17	-	19	25	18	16	31	4

M=

Распределение типов элементарных схем по типам ячеек:

$t_j \backslash Y_i$	1	2	3	4	5	6	7	8	C_i
1			4						10
2	2	1						1	9
3				1	2		2		7
4						3		1	15
5		2		2					17

6			1		1		1	1	18
---	--	--	---	--	---	--	---	---	----

Где t_j – тип элементарной схемы;

Y_i – тип ячейки (модуля);

C_i – стоимость ячейки [в условных единицах].

Вариант №5

Данные структурного анализа схемы:

$M=$	t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8
	17	21	18	13	19	22	25	4

Распределение типов элементарных схем по типам ячеек:

$t_j \backslash Y_i$	1	2	3	4	5	6	7	8	C_i
1			4						9
2	1	2					1		5
3				1				4	14
4				2		2			13
5			1				3		19
6	2				2				6

Где t_j – тип элементарной схемы;

Y_i – тип ячейки (модуля);

C_i – стоимость ячейки [в условных единицах].

Вариант №6

Данные структурного анализа схемы:

$M=$	t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8
	14	12	18	-	16	-	9	5

Распределение типов элементарных схем по типам ячеек:

$t_j \backslash Y_i$	1	2	3	4	5	6	7	8	C_i
1			4						9
2	1			2				1	6
3		2			2				17
4						3	1		14
5			1					3	15

6	1						3		8
---	---	--	--	--	--	--	---	--	---

Где t_j – тип элементарной схемы;

Y_i – тип ячейки (модуля);

C_i – стоимость ячейки [в условных единицах].

Вариант №7

Данные структурного анализа схемы:

M=

t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8
6	51	5	4	13	21	6	15

Распределение типов элементарных схем по типам ячеек:

$t_j \backslash Y_i$	1	2	3	4	5	6	7	8	C_i
1			4						7
2	1			2					6
3		2			2				19
4						3	1		13
5			1				2		8
6	1							4	9

Где t_j – тип элементарной схемы;

Y_i – тип ячейки (модуля);

C_i – стоимость ячейки [в условных единицах].

Вариант №8

Данные структурного анализа схемы:

M=

t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8
19	18	9	-	11	-	51	6

Распределение типов элементарных схем по типам ячеек:

$t_j \backslash Y_i$	1	2	3	4	5	6	7	8	C_i
1		2	1						7
2	1			2					15
3					2		2		10
4						3		2	19
5	3		2						7
6				4					6

Где t_j – тип элементарной схемы;

Y_i – тип ячейки (модуля);

C_i – стоимость ячейки [в условных единицах].

Вариант №9

Данные структурного анализа схемы:

M=

t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8
-	-	41	15	6	8	19	11

Распределение типов элементарных схем по типам ячеек:

$t_j \backslash Y_i$	1	2	3	4	5	6	7	8	C_i
1		1	2						16
2	2			1					15
3					2		2		18
4						3		1	9
5	3		1						11
6				4					5

Где t_j – тип элементарной схемы;

Y_i – тип ячейки (модуля);

C_i – стоимость ячейки [в условных единицах].

Вариант №10

Данные структурного анализа схемы:

M=

t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8
17	13	-	6	18	19	4	0

Распределение типов элементарных схем по типам ячеек:

$t_j \backslash Y_i$	1	2	3	4	5	6	7	8	C_i
1		1	2						13
2	1			1					7
3					2		1		21
4						3		1	18
5	3		1						9
6				4					10

Где t_j – тип элементарной схемы;

Y_i – тип ячейки (модуля);

C_i – стоимость ячейки [в условных единицах].

Вариант №11

Данные структурного анализа схемы:

M=

t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8
4	22	13	-	7	14	10	9

Распределение типов элементарных схем по типам ячеек:

$t_j \backslash Y_i$	1	2	3	4	5	6	7	8	C_i
1	1	2							10
2			1	1					9
3					1		2		16
4						1		3	7
5	1		3						6
6							4		5

Где t_j – тип элементарной схемы;

Y_i – тип ячейки (модуля);

C_i – стоимость ячейки [в условных единицах].

Вариант №12

Данные структурного анализа схемы:

$M=$	t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8
	20	0	11	6	8	19	0	25

Распределение типов элементарных схем по типам ячеек:

$t_j \backslash Y_i$	1	2	3	4	5	6	7	8	C_i
1	1	2							9
2			1	1					10
3					1		2		15
4						1		3	6
5							4		5
6	1		3						7

Где t_j – тип элементарной схемы;

Y_i – тип ячейки (модуля);

C_i – стоимость ячейки [в условных единицах].

Вариант №13

Данные структурного анализа схемы:

$M=$	t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8
	4	13	8	12	11	21	9	12

Распределение типов элементарных схем по типам ячеек:

$t_j \backslash Y_i$	1	2	3	4	5	6	7	8	C_i
1	1	2							8
2			1	1					10
3					1		2		6
4						1		3	15

5							4		5
6	1		3						6

Где t_j – тип элементарной схемы;

Y_i – тип ячейки (модуля);

C_i – стоимость ячейки [в условных единицах].

Вариант №14

Данные структурного анализа схемы:

$M=$	t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8
	-	6	12	4	8	5	13	-

Распределение типов элементарных схем по типам ячеек:

$t_j \backslash Y_i$	1	2	3	4	5	6	7	8	C_i
1	1	2							13
2			1	1					5
3					1		2		7
4						1		3	9
5							4		6
6	1		3						19

Где t_j – тип элементарной схемы;

Y_i – тип ячейки (модуля);

C_i – стоимость ячейки [в условных единицах].

Вариант №15

Данные структурного анализа схемы:

$M=$	t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8
	13	9	-	7	18	11	-	4

Распределение типов элементарных схем по типам ячеек:

$t_j \backslash Y_i$	1	2	3	4	5	6	7	8	C_i
1	1	2							19
2			1	1					6
3					1		2		9
4						1		3	7
5							4		5
6	1		3						13

Где t_j – тип элементарной схемы;

Y_i – тип ячейки (модуля);

C_i – стоимость ячейки [в условных единицах].

Вариант №16

Данные структурного анализа схемы:

t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8
1	13	6	9	5	12	16	13

Распределение типов элементарных схем по типам ячеек:

$t_j \backslash Y_i$	1	2	3	4	5	6	7	8	C_i
1			4						32
2	2	1						1	24
3				2		1			9
4					4				18
5						1	2	1	13
6		4					1		11

Где t_j – тип элементарной схемы;

Y_i – тип ячейки (модуля);

C_i – стоимость ячейки [в условных единицах].

Вариант №17

Данные структурного анализа схемы:

t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8
6	14	8	-	16	21	22	2

Распределение типов элементарных схем по типам ячеек:

$t_j \backslash Y_i$	1	2	3	4	5	6	7	8	C_i
1			4						19
2	2	1							7
3				2		1			10
4					4				6
5						1	2	1	18
6		1				4			5

Где t_j – тип элементарной схемы;

Y_i – тип ячейки (модуля);

C_i – стоимость ячейки [в условных единицах].

Вариант №18

Данные структурного анализа схемы:

t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8
13	17	15	6	9	2	14	6

Распределение типов элементарных схем по типам ячеек:

$t_j \backslash Y_i$	1	2	3	4	5	6	7	8	C_i
1			4						9
2	2	1							20
3				2		1			6
4					4				7
5							1	1	14
6						4			5

Где t_j – тип элементарной схемы;

Y_i – тип ячейки (модуля);

C_i – стоимость ячейки [в условных единицах].

Вариант №19

Данные структурного анализа схемы:

$M=$	t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8
	6	14	2	9	6	15	17	13

Распределение типов элементарных схем по типам ячеек:

$t_j \backslash Y_i$	1	2	3	4	5	6	7	8	C_i
1		2		1					5
2			4						20
3	2			2					14
4		1			1		1	1	6
5					1	2		3	7
6					2		2		14

Где t_j – тип элементарной схемы;

Y_i – тип ячейки (модуля);

C_i – стоимость ячейки [в условных единицах].

Вариант №20

Данные структурного анализа схемы:

$M=$	t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8
	14	9	17	13	15	6	2	18

Распределение типов элементарных схем по типам ячеек:

$t_j \backslash Y_i$	1	2	3	4	5	6	7	8	C_i
1		2		1					7
2			4						6
3	2			2					20
4		1			1		1	1	25
5					1	2		3	5

6					2		2		14
---	--	--	--	--	---	--	---	--	----

Где t_j – тип элементарной схемы;

Y_i – тип ячейки (модуля);

C_i – стоимость ячейки [в условных единицах].

Вариант №21

Данные структурного анализа схемы:

$M=$	t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8
	25	5	0	18	19	22	32	17

Распределение типов элементарных схем по типам ячеек:

$t_j \backslash Y_i$	1	2	3	4	5	6	7	8	C_i
1	4								12
2		2				2			8
3			1	2	1				7
4							2	2	11
5					2		1	1	9
6	1	1	1			1			6

Где t_j – тип элементарной схемы;

Y_i – тип ячейки (модуля);

C_i – стоимость ячейки [в условных единицах].

Вариант №22

Данные структурного анализа схемы:

$M=$	t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8
	22	34	11	2	5	17	21	19

Распределение типов элементарных схем по типам ячеек:

$t_j \backslash Y_i$	1	2	3	4	5	6	7	8	C_i
1	2		1		1				5
2		1		1		2			6
3	1		1		1			1	7
4				1		1	1	1	4
5		4							5
6					2		2		7

Где t_j – тип элементарной схемы;

Y_i – тип ячейки (модуля);

C_i – стоимость ячейки [в условных единицах].

Вариант №23

Данные структурного анализа схемы:

M=	t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8
	21	19	22	15	14	32	54	48

Распределение типов элементарных схем по типам ячеек:

$t_j \backslash Y_i$	1	2	3	4	5	6	7	8	C_i
1	2			1	1				9
2							1	3	6
3						2	2		8
4		1	1	1	1				7
5	1					1		2	7
6		2	2						5

Где t_j – тип элементарной схемы;

Y_i – тип ячейки (модуля);

C_i – стоимость ячейки [в условных единицах].

Вариант №24

Данные структурного анализа схемы:

M=	t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8
	56	43	25	32	22	17	0	19

Распределение типов элементарных схем по типам ячеек:

$t_j \backslash Y_i$	1	2	3	4	5	6	7	8	C_i
1	1	1	1	1					3
2				1			1	2	4
3					2	1		1	5
4	2						2		2
5		1	1		1	1			4
6	3			1					6

Где t_j – тип элементарной схемы;

Y_i – тип ячейки (модуля);

C_i – стоимость ячейки [в условных единицах].

Вариант №25

Данные структурного анализа схемы:

M=	t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8
	43	32	21	20	15	18	33	27

Распределение типов элементарных схем по типам ячеек:

$t_j \backslash Y_i$	1	2	3	4	5	6	7	8	C_i
1						1		3	7
2	1		1		1		1		8
3				1		2		1	5
4	2	1		1					6
5			1				3		9
6		2			2				4

Где t_j – тип элементарной схемы;

Y_i – тип ячейки (модуля);

C_i – стоимость ячейки [в условных единицах].

Контрольная работа №2

Контрольная работа №2 состоит из двух заданий.

Задание №1

Решить задачу размещения. Применить для решения последовательно-итерационный алгоритм.

Исходные данные:

Дан граф схемы $G=(X,U)$, где $U=\{(\overline{x_i x_j})_k\}$, $\}$, где число “к” равно числу кратности ребер, связывающих вершины x_i, x_j .

Вариант №1

$U=\{(\overline{x_1 x_2})_2, (\overline{x_2 x_3})_1, (\overline{x_2 x_8})_1, (\overline{x_1 x_5})_1, (\overline{x_1 x_4})_1, (\overline{x_3 x_7})_3, (\overline{x_4 x_6})_1, (\overline{x_6 x_8})_1, (\overline{x_7 x_9})_3, (\overline{x_5 x_8})_1, (\overline{x_4 x_5})_2\}$

Вариант №2

$U=\{(\overline{x_1 x_2})_2, (\overline{x_2 x_3})_3, (\overline{x_2 x_8})_1, (\overline{x_1 x_5})_5, (\overline{x_1 x_4})_1, (\overline{x_3 x_7})_3, (\overline{x_4 x_6})_1, (\overline{x_6 x_8})_2, (\overline{x_7 x_9})_3, (\overline{x_5 x_8})_1, (\overline{x_4 x_5})_2\}$

Вариант №3

$U=\{(\overline{x_1 x_2})_4, (\overline{x_2 x_3})_3, (\overline{x_2 x_8})_6, (\overline{x_1 x_5})_2, (\overline{x_1 x_4})_1, (\overline{x_3 x_7})_3, (\overline{x_4 x_6})_1, (\overline{x_6 x_8})_1, (\overline{x_7 x_9})_3, (\overline{x_5 x_8})_1, (\overline{x_4 x_5})_2\}$

Вариант №4

$U=\{(\overline{x_1 x_2})_2, (\overline{x_2 x_3})_1, (\overline{x_2 x_8})_1, (\overline{x_1 x_5})_1, (\overline{x_1 x_4})_1, (\overline{x_3 x_7})_3, (\overline{x_4 x_6})_1, (\overline{x_6 x_8})_4, (\overline{x_7 x_9})_1, (\overline{x_5 x_8})_5, (\overline{x_4 x_5})_7\}$

Вариант №5

$U=\{(\overline{x_1 x_2})_2, (\overline{x_2 x_3})_1, (\overline{x_2 x_8})_7, (\overline{x_1 x_5})_1, (\overline{x_1 x_4})_5, (\overline{x_3 x_7})_3, (\overline{x_4 x_6})_1, (\overline{x_6 x_8})_1, (\overline{x_7 x_9})_3, (\overline{x_5 x_8})_1, (\overline{x_4 x_5})_2\}$

Вариант №6

$$U = \{ (\overline{x_1 x_2})_2, (\overline{x_2 x_3})_1, (\overline{x_2 x_8})_1, (\overline{x_1 x_5})_1, (\overline{x_1 x_4})_3, (\overline{x_3 x_7})_3, (\overline{x_4 x_6})_1, (\overline{x_6 x_8})_4, (\overline{x_7 x_9})_6, (\overline{x_5 x_8})_1, (\overline{x_4 x_5})_2 \}$$

Вариант №7

$$\mathbf{U} = \{ (\overline{x_1 x_2})_2, (\overline{x_2 x_3})_1, (\overline{x_2 x_8})_1, (\overline{x_1 x_5})_1, (\overline{x_1 x_4})_3, (\overline{x_3 x_7})_3, (\overline{x_4 x_6})_8, (\overline{x_6 x_8})_9, (\overline{x_7 x_9})_3, (\overline{x_5 x_8})_1, (\overline{x_4 x_5})_2 \}$$

Вариант №8

$$U = \{ (\overline{x_1 x_2})_2, (\overline{x_2 x_3})_1, (\overline{x_2 x_8})_1, (\overline{x_1 x_5})_1, (\overline{x_1 x_4})_1, (\overline{x_3 x_7})_3, (\overline{x_4 x_6})_3, (\overline{x_6 x_8})_5, (\overline{x_7 x_9})_3, (\overline{x_5 x_8})_7, (\overline{x_4 x_5})_2 \}$$

Вариант №9

$$\mathbf{U} = \{ (\overline{x_1 x_2})_2, (\overline{x_2 x_3})_1, (\overline{x_2 x_8})_1, (\overline{x_1 x_5})_4, (\overline{x_1 x_4})_6, (\overline{x_3 x_7})_8, (\overline{x_4 x_6})_1, (\overline{x_6 x_8})_1, (\overline{x_7 x_9})_3, (\overline{x_5 x_8})_1, (\overline{x_4 x_5})_2 \}$$

Вариант №10

$$U = \{ (\overline{x_1 x_2})_4, (\overline{x_2 x_3})_3, (\overline{x_2 x_8})_4, (\overline{x_1 x_5})_5, (\overline{x_1 x_4})_2, (\overline{x_3 x_7})_3, (\overline{x_4 x_6})_1, (\overline{x_6 x_8})_1, (\overline{x_7 x_9})_3, (\overline{x_5 x_8})_1, (\overline{x_4 x_5})_2 \}$$

Вариант №11

$$U = \{ \overline{(x_1 x_2)_2}, \overline{(x_2 x_3)_1}, \overline{(x_2 x_8)_1}, \overline{(x_1 x_5)_3}, \overline{(x_1 x_4)_3}, \overline{(x_3 x_7)_6}, \overline{(x_4 x_6)_7}, \overline{(x_6 x_8)_1}, \overline{(x_7 x_9)_3}, \overline{(x_5 x_8)_1}, \overline{(x_4 x_5)_2} \}$$

Вариант №12

$$U = \{ (\overline{x_1 x_2})_2, (\overline{x_2 x_3})_1, (\overline{x_2 x_8})_1, (\overline{x_1 x_5})_1, (\overline{x_1 x_4})_1, (\overline{x_3 x_7})_3, (\overline{x_4 x_6})_1, (\overline{x_6 x_8})_5, (\overline{x_7 x_9})_6, (\overline{x_5 x_8})_7, (\overline{x_4 x_5})_8 \}$$

Вариант №13

$$\mathbf{U} = \{ (\overline{x_1 x_2})_2, (\overline{x_2 x_3})_1, (\overline{x_2 x_8})_1, (\overline{x_1 x_5})_5, (\overline{x_1 x_4})_5, (\overline{x_3 x_7})_6, (\overline{x_4 x_6})_4, (\overline{x_6 x_8})_1, (\overline{x_7 x_9})_3, (\overline{x_5 x_8})_1, (\overline{x_4 x_5})_2 \}$$

Вариант №14

$$U = \{ (\overline{x_1 x_2})_2, (\overline{x_2 x_3})_1, (\overline{x_2 x_8})_1, (\overline{x_1 x_5})_1, (\overline{x_1 x_4})_3, (\overline{x_3 x_7})_5, (\overline{x_4 x_6})_7, (\overline{x_6 x_8})_1, (\overline{x_7 x_9})_3, (\overline{x_5 x_8})_1, (\overline{x_4 x_5})_2 \}$$

Вариант №15

$$\mathbf{U} = \{ (\overline{x_1 x_2})_2, (\overline{x_2 x_3})_1, (\overline{x_2 x_8})_1, (\overline{x_1 x_5})_1, (\overline{x_1 x_4})_1, (\overline{x_3 x_7})_3, (\overline{x_4 x_6})_1, (\overline{x_6 x_8})_0, (\overline{x_7 x_9})_3, (\overline{x_5 x_8})_0, (\overline{x_4 x_5})_2 \}$$

Вариант №16

$$U = \{ (\overline{x_1 x_2})_2, (\overline{x_2 x_3})_1, (\overline{x_3 x_8})_1, (\overline{x_1 x_4})_1, (\overline{x_1 x_5})_1, (\overline{x_5 x_7})_3, (\overline{x_4 x_6})_1, (\overline{x_6 x_7})_1, (\overline{x_3 x_5})_3, (\overline{x_5 x_9})_1, (\overline{x_4 x_9})_2, (\overline{x_3 x_6})_1, (\overline{x_4 x_8})_1 \}$$

Вариант №17

$$U = \{ (\overline{x_1 x_2})_2, (\overline{x_2 x_3})_4, (\overline{x_3 x_8})_5, (\overline{x_1 x_4})_8, (\overline{x_1 x_5})_5, (\overline{x_5 x_7})_3, (\overline{x_4 x_6})_1, (\overline{x_6 x_7})_1, (\overline{x_3 x_5})_3, (\overline{x_5 x_9})_1, (\overline{x_4 x_9})_2, (\overline{x_3 x_6})_6, (\overline{x_4 x_8})_1 \}$$

Вариант №18

$$U = \{ (\overline{x_1 x_2})_2, (\overline{x_2 x_3})_1, (\overline{x_3 x_8})_5, (\overline{x_1 x_4})_7, (\overline{x_1 x_5})_1, (\overline{x_5 x_7})_3, (\overline{x_4 x_6})_1, (\overline{x_6 x_7})_1, (\overline{x_3 x_5})_3, (\overline{x_5 x_9})_1, \\ (\overline{x_4 x_9})_2, (\overline{x_3 x_6})_{12}, (\overline{x_4 x_8})_4 \}$$

Вариант №19

$$U = \left\{ \overline{(x_1 x_2)}_2, \overline{(x_2 x_3)}_1, \overline{(x_3 x_8)}_1, \overline{(x_1 x_4)}_2, \overline{(x_1 x_5)}_1, \overline{(x_5 x_7)}_3, \overline{(x_4 x_6)}_7, \overline{(x_6 x_7)}_5, \overline{(x_3 x_5)}_3, \overline{(x_5 x_9)}_3, \right. \\ \left. \overline{(x_4 x_9)}_2, \overline{(x_3 x_6)}_4, \overline{(x_4 x_8)}_5 \right\}$$

Вариант №20

$$U = \{ \overline{(x_1 x_2)_2}, \overline{(x_2 x_3)_6}, \overline{(x_3 x_8)_6}, \overline{(x_1 x_4)_1}, \overline{(x_1 x_5)_1}, \overline{(x_5 x_7)_3}, \overline{(x_4 x_6)_1}, \overline{(x_6 x_7)_1}, \overline{(x_3 x_5)_3}, \overline{(x_5 x_9)_1}, \overline{(x_4 x_9)_2}, \overline{(x_3 x_6)_5}, \overline{(x_4 x_8)_4} \}$$

Вариант №21

$$U = \{ (\overline{x_1 x_2})_2, (\overline{x_2 x_3})_5, (\overline{x_3 x_8})_2, (\overline{x_1 x_4})_3, (\overline{x_1 x_5})_4, (\overline{x_5 x_7})_3, (\overline{x_4 x_6})_1, (\overline{x_6 x_7})_3, (\overline{x_3 x_5})_3, (\overline{x_5 x_9})_4, (\overline{x_4 x_9})_2, (\overline{x_3 x_6})_4, (\overline{x_4 x_8})_5 \}$$

Вариант №22

$$U = \{ (\overline{x_1 x_2})_2, (\overline{x_2 x_3})_3, (\overline{x_3 x_8})_3, (\overline{x_1 x_4})_3, (\overline{x_1 x_5})_3, (\overline{x_5 x_7})_3, (\overline{x_4 x_6})_3, (\overline{x_6 x_7})_3, (\overline{x_3 x_5})_3, (\overline{x_5 x_9})_3, (\overline{x_4 x_9})_2, (\overline{x_3 x_6})_1, (\overline{x_4 x_8})_3 \}$$

Вариант №23

$$U = \{ (\overline{x_1 x_2})_2, (\overline{x_2 x_3})_1, (\overline{x_3 x_8})_5, (\overline{x_1 x_4})_1, (\overline{x_1 x_5})_1, (\overline{x_5 x_7})_3, (\overline{x_4 x_6})_1, (\overline{x_6 x_7})_1, (\overline{x_3 x_5})_3, (\overline{x_5 x_9})_1, (\overline{x_4 x_9})_2, (\overline{x_3 x_6})_8, (\overline{x_4 x_8})_8 \}$$

Вариант №24

$$U = \left\{ \overline{(x_1 x_2)_2}, \overline{(x_2 x_3)_5}, \overline{(x_3 x_8)_6}, \overline{(x_1 x_4)_4}, \overline{(x_1 x_5)_1}, \overline{(x_5 x_7)_3}, \overline{(x_4 x_6)_1}, \overline{(x_6 x_7)_7}, \overline{(x_3 x_5)_3}, \overline{(x_5 x_9)_1}, \right. \\ \left. \overline{(x_4 x_9)_2}, \overline{(x_3 x_6)_1}, \overline{(x_4 x_8)_1} \right\}$$

Вариант №25

$$U = \left\{ \overline{(x_1 x_2)}_2, \overline{(x_2 x_3)}_1, \overline{(x_3 x_8)}_1, \overline{(x_1 x_4)}_1, \overline{(x_1 x_5)}_7, \overline{(x_5 x_7)}_3, \overline{(x_4 x_6)}_1, \overline{(x_6 x_7)}_6, \overline{(x_3 x_5)}_3, \overline{(x_5 x_9)}_1, \right. \\ \left. \overline{(x_4 x_9)}_2, \overline{(x_3 x_6)}_8, \overline{(x_4 x_8)}_3 \right\}$$

Вариант №26

$$U \Rightarrow \left\{ \overline{(x_1 x_2)}_2, \overline{(x_2 x_3)}_4, \overline{(x_3 x_8)}_1, \overline{(x_1 x_4)}_3, \overline{(x_1 x_5)}_1, \overline{(x_5 x_7)}_3, \overline{(x_4 x_6)}_1, \overline{(x_6 x_7)}_1, \overline{(x_3 x_5)}_3, \overline{(x_5 x_9)}_1, \right. \\ \left. \overline{(x_4 x_9)}_2, \overline{(x_3 x_6)}_1, \overline{(x_4 x_8)}_5 \right\}$$

Вариант №27

$$U \Rightarrow \left\{ \overline{(x_1 x_2)}_2, \overline{(x_2 x_8)}_4, \overline{(x_3 x_8)}_1, \overline{(x_1 x_4)}_3, \overline{(x_1 x_5)}_1, \overline{(x_5 x_7)}_3, \overline{(x_4 x_6)}_1, \overline{(x_6 x_7)}_1, \overline{(x_3 x_7)}_3, \overline{(x_5 x_9)}_1, \right. \\ \left. \overline{(x_4 x_9)}_2, \overline{(x_3 x_6)}_1, \overline{(x_4 x_8)}_5, \overline{(x_4 x_9)}_5 \right\}$$

Вариант №28

$$U \Rightarrow \left\{ \overline{(x_1 x_2)}_2, \overline{(x_2 x_3)}_1, \overline{(x_3 x_8)}_1, \overline{(x_1 x_4)}_1, \overline{(x_1 x_5)}_7, \overline{(x_5 x_7)}_3, \overline{(x_4 x_6)}_1, \overline{(x_6 x_7)}_6, \overline{(x_3 x_5)}_3, \overline{(x_5 x_9)}_1, \right. \\ \left. \overline{(x_4 x_9)}_2, \overline{(x_5 x_6)}_8, \overline{(x_4 x_8)}_3 \right\}$$

Вариант №29

$$U \Rightarrow \left\{ \overline{(x_1 x_2)}_2, \overline{(x_2 x_3)}_1, \overline{(x_3 x_8)}_1, \overline{(x_1 x_4)}_1, \overline{(x_1 x_5)}_7, \overline{(x_5 x_7)}_3, \overline{(x_4 x_6)}_1, \overline{(x_6 x_7)}_6, \overline{(x_3 x_5)}_3, \overline{(x_5 x_9)}_1, \right. \\ \left. \overline{(x_4 x_9)}_2, \overline{(x_3 x_6)}_8, \overline{(x_4 x_8)}_3 \right\}$$

Вариант №30

$$U \Rightarrow \left\{ \overline{(x_1 x_2)}_2, \overline{(x_2 x_3)}_1, \overline{(x_2 x_8)}_1, \overline{(x_1 x_5)}_1, \overline{(x_1 x_4)}_1, \overline{(x_3 x_7)}_0, \overline{(x_5 x_6)}_3, \overline{(x_6 x_8)}_1, \overline{(x_7 x_9)}_3, \overline{(x_8 x_9)}_3, \right. \\ \left. \overline{(x_3 x_5)}_4 \right\}$$

Задание №2

Спроектировать печатную плату для схемы, приведенной в Приложении А.

Методические указания приведены на стр. 39 данного пособия.

СПИСОК ЛИТЕРАТУРЫ

Основная литература

1. Сучков Д.И. Проектирование печатных плат в САПР PCAD 4.5. – Обнинск: Макрос, 1992. – 475 с.
2. Курейчик В.М. Математическое обеспечение конструкторского и технологического проектирования с применением САПР: Учебник для высших учебных заведений. Допущено Министерством высшего и среднего специального образования СССР в качестве учебника для студентов вузов, обучающихся по специальности «Конструирование и технология радиоэлектронных средств». – М.: Радио и связь, 1990. – 352 с.
3. Корячко В.П., Курейчик В.М., Норенков И.П. Теоретические основы САПР. Допущено Министерством высшего и среднего специального образования СССР в качестве учебника для студентов вузов, обучающихся по специальности «Конструирование и производство электронно-вычислительной аппаратуры». – М.: Энергоиздат, 1987. – 383 с.
4. Уваров А. P_CAD 2000. ACCEL EDA. Конструирование печатных плат: Учебный курс. – СПб: Питер, 2001. – 320 с.

Дополнительная литература

1. Фидлер Дж.К., Найтингейд К. Машинное проектирование электронных схем. – М.: Высшая школа, 1985.
2. Разевиг В.Д. ACCEL EDA 15.0 (P_CAD 2000). Для студентов технических университетов. – М.: Солон, 2000. – 413 с..
3. Колесников А. EXCEL 2000, русифицированная версия. – Киев: Ириша, ВНУ, 1999. – с. 490.
4. Дьяконов В.П. Система MathCAD. – М.: Радио и связь, 1993.