

В. Неррертер

РАСЧЕТ
ЭЛЕКТРИЧЕСКИХ
ЦЕПЕЙ
НА ПЕРСОНАЛЬНОЙ
ЭВМ

РАСЧЕТ ЭЛЕКТРИЧЕСКИХ ЦЕПЕЙ НА ПЕРСОНАЛЬНОЙ ЭВМ



В. Неррертер

РАСЧЕТ ЭЛЕКТРИЧЕСКИХ ЦЕПЕЙ НА ПЕРСОНАЛЬНОЙ ЭВМ

Перевод с немецкого С.П. БУНДЕЛЕВА

Под редакцией канд. техн. наук А.Н. ЛЕДОВСКОГО



МОСКВА ЭНЕРГОАТОМИЗДАТ 1991

ББК 31.21

Н 54

УДК 621.3.049:681.31—181.4

Рецензенты В. А. Алешечкин, Г. Х. Геворкян, Л. М. Саликов

NERRETER W.
BERECHNUNG ELEKTRISCHER SCHALTUNGEN
MIT DEM PERSONAL COMPUTER

Carl Hanser Verlag. München, 1987

Нерретер В.

Н 54 Расчет электрических цепей на персональной ЭВМ: Пер. с нем. — М.: Энергоатомиздат, 1991. — 220 с.: ил.

ISBN 5-283-02490-3

Изложены методы расчета электрических линейных и нелинейных цепей на персональной ЭВМ. Для цепей постоянного тока показаны приемы решения систем линейных уравнений, для цепей переменного тока — алгебраических систем уравнений с комплексными коэффициентами. Приведен анализ динамических процессов при различных воздействиях. Все программы представлены на языках БЕЙСИК и Турбо-ПАСКАЛЬ.

Для инженерно-технических работников, студентов вузов электротехнических специальностей.

Н 2404090000-270 216-90
051(01)-91

ББК 31.21

ISBN 5-283-02490-3 (рус.)

© Carl Hanser Verlag
München, 1987

ISBN 3-446-14167-7 (нем.)

© Перевод на русский язык,
Энергоатомиздат, 1991

Предисловие

Если бы было время!.. Если бы было время, то взял бы гл. 4, разобрал все примеры, ввел программу «Эйлер-тест» и другие и вволю «набаловался» с ними, ведь так интересно посмотреть, как меняется переходный процесс в зависимости от параметров, в зависимости от начальных условий, как сходятся результаты при разном числе шагов и разной ширине шага. Потом «пристроил» бы графику (хотя все это можно получить и аналитически) и стал бы наблюдать на экране дисплея за этими процессами. График на экране обладает особой притягательной силой, все становится наглядным, зримым, а процесс — более обзримым, его воспринимаешь как нечто цельное.

Предлагаемая книга хороша своей доступностью, понятностью, теми деталями, которые почему-то многие авторы упускают. Можно, однако, надеяться, что из 50 тысяч приобретших эту книгу хотя бы сто будут с ней работать. Может быть, они опробуют в деле эти программы, попытаются их развить, дополнить, усовершенствовать, видоизменить, приспособить для решения своих задач. Только надежда на такое продолжение — а это необходимое условие развития культуры — может окупить затраты материальных и духовных сил при выпуске книжной продукции вообще.

А теперь несколько практических замечаний.

В отечественных ПЭВМ знак операции возведения в степень \wedge , используемый в данной книге, может быть иным (например, \uparrow или \uparrow). Читателю это следует уточнить и внести изменения. То же относится к знаку символьной (строковой) переменной $\$$: у нас чаще используется «солнышко» или «черепашка». Если же в версии БЕЙСИКа, которая реализована на ПЭВМ читателя, нет строковых переменных вообще, то потребуются несколько

большая переделка. В частности, это касается замены ввода строковой переменной на ввод цифр.

Например, строки 4520—4528 (см. п. 4.5.1) можно записать так:

```
4520 INPUT «Тип ветви: 1—R, 2—G, 3—L, 4—C»;FR
4522 IF FR=1 THEN 4532
4524 IF FR=2 THEN 4538
4526 IF FR=3 THEN 4566
4528 IF FR=4 THEN 4572
```

Автор книги использует строковые переменные только в таких ситуациях. Их относительно легко найти (строки 2520—2530 в п. 2.5.2, 3526—3540 в п. 3.3.1, 142—146 в п. 5.1.6 и т. д.).

Во многих версиях языка БЕЙСИК не допускается описание массивов с указанием их размерности в виде переменной DIM X(K), которое используется в книге. Значит, пользователю надо вместо K подставить определенное число (скажем 10 или 100). Это требует известной осторожности при вводе и запуске программ.

В ряде версий языка БЕЙСИК необходимо, чтобы операторы DIM находились в начале программы. Следует и этот вопрос выяснить и переставить строки, если нужно.

Пожелаю читателям терпения и удачи при работе с этой книгой.

Г. Х. Геворкян

Предисловие к русскому изданию

С появлением персональных ЭВМ (ПЭВМ) как вычислительных средств на рабочем месте инженера эффективность его работы по сравнению с 70-ми годами значительно возросла. В то время как использование логарифмических линеек и простых калькуляторов позволяет выполнять лишь несколько операций с плавающей запятой в секунду, ПЭВМ выполняет от 10^4 до 10^5 подобных операций. В равной мере дело касается и оперативной памяти: простой калькулятор может запомнить только несколько чисел с плавающей запятой, а ПЭВМ запоминает от 10^4 до 10^5 таких чисел.

Несмотря на такие характеристики производительность ПЭВМ все-таки ниже производительности стандартных ЭВМ с аналогичными параметрами. Поэтому необходимо отдельно разрабатывать программы, ориентированные для эксплуатации на ПЭВМ и требующие небольшой памяти и малого операционного времени.

Это касается и расчета электрических схем. В книге рассматриваются исключительно вопросы анализа электрических цепей и приводятся программы, разработанные для небольших ПЭВМ. Значительное внимание уделяется универсальности использования программ: например, характеристики нелинейных элементов схемы могут вводиться поточно. Это позволяет рассчитывать не только диоды и транзисторы, но и другие нелинейные элементы. Однако в книге представлены не только программы для ПЭВМ. Особое внимание уделяется описанию электротехнических и математических основ методов и алгоритмов. Показано, как из относительно простых программ для линейных схем постоянного тока «возникают» программы для анализа переходных процессов в нелинейных цепях и как различные типы нелинейных це-

пей могут быть с помощью дискретных токовых моделей сведены к цепям постоянного тока.

Программы разработаны на двух языках программирования. В гл. 1—6 для каждого типа цепей приведены программы на БЕЙСИКе. При составлении этих программ были использованы некоторые операторы так называемого мини-БЕЙСИКа. Поэтому созданные программы применяются на различных персональных и даже на некоторых бытовых ЭВМ.

С помощью программы, написанной на языке программирования ПАСКАЛЬ, можно рассчитать цепи любых типов. Поскольку эта программа перед эксплуатацией компилируется, время, необходимое для ее выполнения, оказывается существенно меньше, чем для программ на языке БЕЙСИК.

Я рад тому, что теперь книга издается и на русском языке, и благодарю издательство за ее заботливую обработку.

Лемго, август 1988

Вольфганг Неррертер

Предисловие к немецкому изданию

С появлением ПЭВМ на рабочем столе инженера оказались вычислительные средства, решительно превосшедшие по своему качеству все, что было в его распоряжении до конца 70-х годов. Если с помощью логарифмической линейки или карманного калькулятора можно было выполнять только несколько операций в секунду над числами с плавающей запятой, то ПЭВМ обеспечили скорость вычислений до 10^5 операций в секунду. Примерно в той же мере увеличилась и оперативная память: простой микрокалькулятор мог «запомнить» лишь несколько чисел с плавающей запятой, ПЭВМ — до 10^5 таких чисел.

В то же время производительность ПЭВМ на порядок ниже производительности обычной ЭВМ. Поэтому имеет смысл специально для ПЭВМ разработать программы, требующие относительно небольшой памяти и малого времени вычисления. Это, в частности, относится и к программам расчета электрических схем.

В предлагаемой книге, посвященной анализу электрических цепей, описаны программы, предназначенные для работы на ПЭВМ с относительно невысоким быстродействием и малым объемом оперативной памяти. При этом большое значение придается универсальности использования программ: например, за счет поточечного ввода характеристик нелинейных элементов могут рассчитываться не только транзисторы и диоды, но и другие нелинейные элементы электрических цепей.

Особое внимание в книге уделено описанию электротехнических и математических принципов, лежащих в основе различных методов расчета электрических цепей. В книге дается последовательный переход от относительно простых программ для расчета линейных электрических цепей постоянного тока к сложным программам рас-

чета переходных процессов в нелинейных электрических цепях. Показано, как с помощью дискретной модели токовой схемы можно различные типы электрических цепей свести к цепи постоянного тока.

Приведенные в книге программы были разработаны для ПЭВМ типа COMMODORE 10. Эта ПЭВМ имеет оперативную память 256 Кбайт, два дисководов для гибких дисков (по 360 Кбайт) и монокромный дисплей (80 символов в строке). Она совместима с ПЭВМ типа IBM XT. Для нормального функционирования программ достаточно одного дисководов. Специальной графической платы не требуется.

Книга рассчитана на читателя, имеющего предварительные знания языков программирования БЕЙСИК и желательно ПАСКАЛЬ.

Для программ на БЕЙСИКе используются немногие наиболее употребительные директивы так называемого мини-БЕЙСИКа. Это означает, что разработанные программы пригодны для различных ПЭВМ и даже для некоторых типов бытовых ЭВМ.

Программы реализованы в операционной системе MS—DOS.

Я признателен всем коллегам, которые своими замечаниями и своей помощью содействовали мне в этой работе. Кроме того, я благодарен издательству Carl Hanser за совместную работу, основанную на взаимном доверии.

Лемго, январь 1987

Вольфганг Нерретер

Глава 1

ПРИМЕНЕНИЕ ПЕРСОНАЛЬНЫХ ЭВМ ДЛЯ РАСЧЕТА ЭЛЕКТРИЧЕСКИХ ЦЕПЕЙ

Современные ЭВМ содержат множество интегральных схем, электрические элементы которых — резисторы, конденсаторы, транзисторы — размещаются на одном кристалле. Такая интегральная схема (ИС) не может быть построена и полностью протестирована в лаборатории, как в случае проектирования на дискретных элементах. Во-первых, потому, что дискретные элементы ввиду различных технологических процессов имеют иные, чем у аналогичных интегральных элементов, характеристики, во-вторых, в дискретных структурах не существенны те эффекты, которые для ИС возникают при размещении отдельных элементов на общем основании. Кроме того, случается, что измерительное оборудование (головки щупов, проводники) не обеспечивает требуемой точности измерений. Поэтому проектирование ИС ведется автоматизированными методами. В конце 60-х годов были разработаны программы SPICE (NAP) и SCEPTRE (BOS) для расчета электрических схем, реализованные на крупных ЭВМ, которые позволили значительно ускорить проектирование ИС.

1.1. Персональные ЭВМ

В связи с ростом количества задач, решаемых на ЭВМ, необходимо разгружать крупные ЭВМ от простых программных заданий (например, редактирования программных текстов). Для такого рода задач используются ПЭВМ, находящиеся на рабочих местах отдельных пользователей, поэтому их (ПЭВМ) называют также автоматизированными рабочими местами.

Однако широкое применение ПЭВМ никоим образом не объясняется только необходимостью разгрузки стандартной ЭВМ. Так, с помощью ПЭВМ на небольших предприятиях осуществляется бухгалтерский учет, составление счетов и накладных, обработка текстов и оптимизация организационных процессов. Существуют и другие возможности использования ПЭВМ, например при управлении станками на производстве и при обработке результатов измерений.

По составу периферийного оборудования и быстродействию ПЭВМ значительно отличаются друг от друга. В качестве наиболее простого варианта может быть рассмотрена 8-разрядная ПЭВМ, например APPLEII, с объемом оперативной памяти 64 Кбайт и тактовой частотой 1 МГц.

Несмотря на постоянное совершенствование 8-разрядных ЭВМ в последние годы их вытесняют 16-разрядные машины, например ПЭВМ COMMODEORE 10, на которой были разработаны программы на языке ПАСКАЛЬ для этой книги. С помощью этой же ПЭВМ выполнена и обработка текста книги.

Обычно в состав 16-разрядной ЭВМ входят центральный блок с процессором Intel 8088 и тактовой частотой 4,77 МГц, устройство оперативной памяти с объемом 256 Кбайт, а также дисковод для гибкого диска, на котором может быть размещено 360 Кбайт информации.

Существуют ПЭВМ и с более высокой производительностью. Обычно они снабжаются аббревиатурой АТ (Advanced technology — улучшенная технология). Дальнейшее развитие ПЭВМ идет в направлении увеличения оперативной памяти и быстродействия.

1.2. Языки программирования

1.2.1. Язык программирования БЕЙСИК. Почти каждая ПЭВМ рассчитана на использование языка БЕЙСИК. Поэтому программы, приведенные в книге, написаны на этом языке.

БЕЙСИК обладает рядом преимуществ перед другими языками программирования. Во-первых, он прост. Кроме того, для ПЭВМ он выполнен как интерпретатор. Это упрощает отладку программ. Однако интерпретатор обуславливает более медленную обработку программ по сравнению с программами, предварительно откомпили-

рованными (переведенными) в машинный код. Поэтому многие ПЭВМ снабжаются компилятором БЕЙСИКа, с помощью которого можно создавать более быстродействующие программы в машинных кодах.

Значительные трудности возникают у пользователей из-за того, что производители ПЭВМ используют различные версии языка БЕЙСИК. Для того чтобы программы на БЕЙСИКе могли работать на ПЭВМ различных типов, необходимо применять так называемый мини-БЕЙСИК, соответствующий американскому стандарту ANSI X3.60—1978. В табл. 1.1 представлены встречающиеся в книге директивы языка БЕЙСИК. Необязательная директива LET для присвоения значений не используется.

Как уже упоминалось, программы сначала были разработаны на языке БЕЙСИК для ПЭВМ APPLE. Так же, как и в некоторых других версиях БЕЙСИКа, в этом случае значимыми для имен переменных являются лишь первые два символа; переменные WERT и WENDERPUNKT, например, интерпретатором не различаются. Поэтому во всех программах на БЕЙСИКе любые имена переменных выбирались состоящими из одного или двух алфавитно-цифровых символов.

В зависимости от типа переменных они могут принимать значения, лежащие в стандартных для 8-разрядных ЭВМ диапазонах: для INTEGER — от -32767 до $+32767$; для REAL — от -10^{38} до $+10^{38}$. Наименьшее значение вещественного числа (с плавающей запятой) составляет примерно 10^{-38} .

Серьезнейшей проблемой для БЕЙСИК-интерпретатора ПЭВМ является точность вычислений. Продемонстрируем это на простом примере, в котором мы тысячу раз прибавим константу 0,1 к переменной A, начальное значение которой равно 0:

```
100 A=0
110 FOR I=1 TO 1000
120 A=A+0.1
130 NEXT
140 PRINT «A=»; A
150 END
```

Некоторые бытовые ЭВМ выдают после решения этого примера точный результат, равный 100, что на несколько десятичных разрядов точнее любого результата, полученного на ПЭВМ.

Таблица 1.1. Используемые директивы языка БЕЙСИК

Директива	Комментарий
ABS	Определение абсолютного значения числового выражения
ATN	Функция арктангенс
COS	Функция косинус
DATA	Определение строк данных в программе
DEFDBL	Определение переменных с двойной точностью
DEF FN	Определение функции
DEFINT	Определение целочисленной переменной
DIM	Описание массива переменных
END	Конец
EXP	Экспоненциальная функция
FOR TO STEP	Программный цикл: определение начального, конечного значений и шага
GOSUB	Вызов подпрограммы
GOTO	Программный переход на заданную строку
IF THEN	Проверка условия
INPUT	Задание значения величины вводом с клавиатуры
INT	Определение наибольшего целого числа, которое меньше, чем значение числового выражения
LOG	Натуральный логарифм
NEXT	Изменение переменной цикла
ON GOTO	Переход на заданную строку программы в зависимости от аргумента ON
PRINT	Вывод
READ	Чтение данных из строки DATA
REM	Комментарий
RETURN	Завершение подпрограммы, возврат управления в основную программу
SGN	Определение знака числового выражения
SIN	Функция синус
SQR	Функция квадратный корень
TAN	Функция тангенс

Например, при использовании языка БЕЙСИК в ПЭВМ APPLE A = 99,9999963, а при использовании БЕЙСИКА GW в ПЭВМ COMMODORE PC 10 с обычной точностью A = 99,99904 и с двойной точностью A = =99,99999999999978.

Язык БЕЙСИК GW для COMMODORE PC 10 разработан фирмой Microsoft и совместим с другими версиями БЕЙСИКа этой фирмы (например, с языком

БЕЙСИК IBM PC). В БЕЙСИКе GW переменные могут запоминаться и обрабатываться с обычной или двойной точностью. При обычной точности вещественное (с плавающей запятой) число запоминается с семью десятичными разрядами. При двойной точности вещественное число запоминается с 17 десятичными разрядами, из которых при выводе на индикацию имеют смысл лишь первые 16 разрядов. Двойная точность может быть выбрана для констант путем размещения знака * перед числом, а для глобальной переменной — с помощью директивы DEFDBL. В этом случае директива DEFDBL A объявляет переменными двойной точности каждую переменную, чье имя начинается с буквы A.

1.2.2. Структура программ на языке БЕЙСИК. Состав программ на языке БЕЙСИК приведен в приложении П1, причем программы оптимизированы как с точки зрения необходимой памяти, так и с точки зрения времени вычислений. Из-за недостатка времени не удалось выполнить тестирование большого числа примеров. Можно лишь утверждать, что описанные примеры были рассчитаны правильно.

Любая программа на БЕЙСИКе начинается с директивы PRINT в строке 100; при обращении к этой директиве распечатывается идентификатор программы. В качестве номеров строк используются исключительно четные числа — таким образом обеспечивается возможность разного рода расширений программы. Основная программа может занимать строки с 100-й по 798-ю. Строки с 800-й по 998-ю с 1100-й по 1998-ю предусмотрены для директив DATA, если входные величины не вводятся в ПЭВМ с помощью клавиатуры. Программы всегда заканчиваются оператором 9999 END, который в тексте программ опущен.

Программы структурируются с помощью подпрограмм. Для обозначения номеров строк подпрограммы используются четырехразрядные числа, первая цифра которых согласуется с номером главы, где эта подпрограмма описана. Список подпрограмм приведен в приложении П2.

Каждая подпрограмма (UPR) обозначается номером строки своей первой директивы. Описание и текст подпрограммы UPR приводятся только один раз. В строках 1000—1010 каждой программы на БЕЙСИКе описано, какие подпрограммы вызываются в основной програм-

ме. Так, строка

1000 REM : UPR 2000, 2100

указывает на то, что при вызове основной программы в оперативной памяти ПЭВМ должны быть подпрограммы 2000 и 2100.

Поскольку некоторые подпрограммы могут быть использованы в нескольких основных программах, необходимо вести учет имен переменных. Каждой переменной во всех программах на БЕЙСИКе соответствует одно и то же имя. Эти имена собраны в приложении ПЗ.

Ввиду отсутствия блочной структуры в программах, написанных на языке БЕЙСИК, нет разделения между глобальными и локальными переменными: переменные основной программы являются одновременно переменными подпрограммы и наоборот. Однако различить переменные основной программы и подпрограммы можно довольно просто: имена переменных, которые используются исключительно в подпрограмме, состоят из одной буквы и одной цифры. Имена переменных, применяемые также и в основной программе, состоят из одной или двух букв.

Поскольку при выполнении программ в каждый конкретный момент времени вызывается только одна подпрограмма, переменные для подпрограмм можно выбирать произвольно, не опасаясь перепутать имена переменных в основной программе и подпрограммах. При этом в описании каждой подпрограммы дается пояснение каждого имени переменной.

1.2.3. Язык программирования ПАСКАЛЬ. Язык ПАСКАЛЬ является по существу дальнейшим развитием известного примерно с 1959 г. языка программирования АЛГОЛ. Оба языка до недавнего времени использовались исключительно в вычислительных центрах институтов и нескольких проектных лабораторий (на производстве чаще используются языки ФОРТРАН и КОБОЛ). Для имеющихся ПЭВМ не подходит ни один из этих языков программирования: во-первых, потому, что для редактора текста, компилятора, редактора связей и системной библиотеки необходима память порядка нескольких сотен килобайт, а во-вторых, потому, что редактирование текстов, компиляция, редактирование связей, а также загрузка и запуск программы в машинных кодах являются трудоемкими и длительными процедурами.

Положение изменилось в 1983 г., когда американская фирма Borland разработала язык Турбо-ПАСКАЛЬ. Весь Турбо-ПАСКАЛЬ занимает только 40 Кбайт оперативной памяти. Язык включает в себя редактор текста, работа с которых аналогична работе с широко известным текстовым редактором WORDSTAR.

Тексты данной книги были подготовлены с помощью редактора Турбо. Компилятор Турбо-ПАСКАЛЯ обрабатывает исходный текст с очень высокой скоростью. Кроме того, этот компилятор наиболее экономно, по сравнению с другими, использует память ПЭВМ. Ниже перечислены преимущества Турбо-ПАСКАЛЯ перед БЕЙСИКом:

этот язык значительно «богаче» БЕЙСИКа, например по типам переменных или операторам цикла, что существенно облегчает работу программиста;

исходные тексты, т. е. собственно программы на языке ПАСКАЛЬ, ввиду их блочной структуры в общем случае более наглядны, чем программы на языке БЕЙСИК, для которых наглядность теряется из-за множества операторов переходов;

в то время как интерпретатор БЕЙСИКа управляет памятью лишь в 64 Кбайт, Турбо-ПАСКАЛЬ может использовать всю оперативную память;

скомпилированная программа на Турбо-ПАСКАЛЕ выполняется значительно быстрее, чем программа, обрабатываемая интерпретатором БЕЙСИКа. В версии Турбо-87 возможно еще большее ускорение, если ПЭВМ располагает встроенным сопроцессором.

Турбо-ПАСКАЛЬ обеспечивает высокое быстродействие, что установлено опытным путем (с помощью программы, соответствующей одной из программ на языке БЕЙСИК):

```
PROGRAM test; VAR a: REAL; i: INTEGER;  
BEGIN  
A:=0;  
FOR I:=1 TO 1000 DO a:=a+0.1;  
writeln (I, 'A=', a)  
END.
```

При этом тестировался результат работы программы. Первая строка — результат работы программы на Турбо, вторая — на Турбо-87:

A=9.9999999980E+01

A=9.99999999999986E+001

Диапазон чисел с плавающей запятой лежит в пределах от -10^{38} до $+10^{38}$, причем если язык TURBO «работает» с 11 десятичными разрядами, то язык TURBO-87 — с 15 десятичными разрядами. Диапазон чисел с плавающей запятой для TURBO-87 составляет от -10^{308} до $+10^{308}$.

1.3. Типы цепей

С помощью ПЭВМ нельзя спроектировать схему высокого уровня интеграции. Однако существует множество расчетных задач, для которых целесообразно использовать ПЭВМ. Все зависит от объема ее оперативной памяти и быстродействия.

Возможность расчета электрических схем на ПЭВМ определяется лишь наличием соответствующего программного обеспечения. Методы расчета, используемые в стандартных ЭВМ, можно реализовать на ПЭВМ с учетом ограничений, обусловленных следующими причинами:

- отсутствием для многих ПЭВМ компиляторов с распространенных языков программирования (например, ФОРТРАНА). Эта проблема решается достаточно просто, если на ПЭВМ устанавливается уже скомпилированная программа. Например, в настоящее время существует версия программы SPICE для ПЭВМ класса АТ;

- отсутствием в большинстве случаев в составе программного обеспечения ПЭВМ библиотечных программ для часто используемых методов расчета (например, решения систем линейных уравнений), широко распространенных в математическом обеспечении стандартных ЭВМ;

- низкой скоростью вычислений, что требует разработки соответствующих вычислительных методов;

- малым объемом оперативной памяти ПЭВМ.

Для небольших ПЭВМ разработаны программы на языке БЕЙСИК, позволяющие отдельно рассчитывать каждый тип электрической схемы. Для промышленных ПЭВМ дополнительно разработана программа NWA (анализ цепей) на языке Турбо-ПАСКАЛЬ.

В дальнейшем речь пойдет исключительно о таком анализе электрических схем и цепей, который предполагает, что при заданных параметрах элементов и структуре цепи определяются токи и напряжения в ветвях цепи. Мы различаем несколько типов цепей.

Линейная цепь постоянного тока состоит из активных сопротивлений, идеальных источников питания с постоянными во времени параметрами и управляемых источников питания. Этот тип цепи имеет малое практическое значение, однако принципиально важен для расчета цепей других типов. Линейные цепи подробно рассмотрены в гл. 2. Там же приведена программа GLEICH расчета линейных цепей постоянного тока, которая занимает около 5,2 Кбайт оперативной памяти. Дополнительно следует предусмотреть память для записи переменных.

Линейная цепь синусоидального тока состоит из основных двухполюсников, идеальных (независимых от нагрузки) источников переменного тока с постоянной частотой и управляемых источников переменного тока с постоянным вещественным коэффициентом управления. В гл. 3 для расчета линейных цепей синусоидального тока приведена программа SINUS, требующая 8,1 Кбайт оперативной памяти.

Нелинейная цепь постоянного тока дополнительно содержит элемент линейной цепи постоянного тока — двухполюсник с нелинейной вольт-амперной характеристикой. Для расчета такой цепи по программе NEWTON необходимо 10,3 Кбайт памяти.

Упомянутые цепи являются статическими, и поэтому эффективные (действующие) значения напряжений и токов в их ветвях постоянны во времени. Для динамических цепей, напротив, характерно изменение структуры цепи в течение переходного процесса относительно момента времени $t=0$, а напряжения и токи в ветвях вычисляются для нескольких моментов времени $t>0$.

Линейная динамическая цепь состоит из двухполюсников, управляемых и идеальных источников, исходные параметры которых в общем случае по-разному зависят от времени. Программа DYNA-NL, разработанная для расчета таких цепей, занимает 8,8 Кбайт памяти.

Нелинейная динамическая цепь содержит дополнительно к элементам линейной динамической цепи двухполюсники с нелинейными характеристиками, которые могут быть индуктивностями или емкостями. В гл. 6 описана программа DYNA-NL для расчета этих цепей. В программах NEWTON, DYNA-NL и NWA характеристики нелинейных двухполюсников задаются поточечно. Таким образом могут быть заданы характеристики не только диодов и транзисторов, но и любых других нели-

нейных двухполюсников, например неоновых ламп, терморезисторов с положительным температурным коэффициентом, варисторов и т. д.

Применение описанных в книге программ не ограничивается только областью электротехники. Так, процессы нагрева тел могут быть рассмотрены по аналогии с электрическими цепями (табл. 1.2).

Т а б л и ц а 1.2. Аналогия физических величин

В теплотехнике	В электротехнике
Количество тепла Q	Количество заряда Q
Тепловой поток dQ/dt	Ток $i = dQ/dt$
Теплопроводность α	Проводимость γ
Термическое сопротивление $1/(\lambda A)$	Сопротивление $R = 1/(\gamma A)$
Разность температур ΔT	Напряжение $U = \Delta \phi$
Теплоемкость $C = dQ/dT$	Емкость $C = dQ/dU$

Г л а в а 2

ЛИНЕЙНЫЕ ЦЕПИ ПОСТОЯННОГО ТОКА

2.1. Линейные двухполюсные цепи

Линейными двухполюсными цепями принято называть цепи, содержащие только такие двухполюсники, вольт-амперная характеристика которых изображается прямой линией. Для постоянного тока — это активное сопротивление $R = U/I = \text{const}$ или проводимость $G = I/U = \text{const}$, идеальный источник напряжения $U_q = \text{const}$ и идеальный источник тока $I_q = \text{const}$ [1]. В дальнейшем подобные двухполюсники мы будем называть линейными двухполюсниками (рис. 2.1), а цепи, содержащие такие двухполюсники, — линейными цепями.

2.1.1. Анализ электрических цепей. Для анализа электрической цепи задается ее структура и определяются параметры всех элементов. Требуется же найти токи и напряжения во всех ветвях цепи, которые в дальнейшем будем именовать параметрами ветвей.

В основе любых методов анализа электрических цепей лежат уравнения двухполюсников и закон узловых потенциалов (первый закон Кирхгофа) и контурных токов (второй закон Кирхгофа).

Непосредственное решение этих уравнений возможно только для относительно простых схем. При расчете сложных схем применяются специальные методы, например метод наложений или преобразования соединений звезда—треугольник. Кроме того, на основании личного

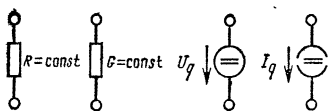


Рис. 2.1. Линейные двухполюсники

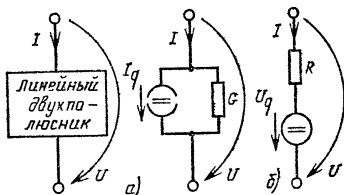


Рис. 2.2. Ветвь линейной цепи и замещающие двухполюсники:
а — линейный источник тока; б — линейный источник напряжения

опыта исследователь должен принять решение, какой из известных методов анализа целесообразно использовать в конкретном случае. С точки зрения машинной реализации подобный подход не приемлем.

2.1.2. Метод узловых потенциалов. Для автоматизированного анализа цепей были разработаны различные методы: узловых потенциалов, контурных токов, множества сечений и т. п. Однако в каждом конкретном случае целесообразно заранее выбрать подходящий метод.

В дальнейшем будем рассматривать в качестве единственного метода анализа цепей метод узловых потенциалов [31]. Решающее преимущество этого метода состоит в том, что он практически не требует предварительной подготовки. Кроме того, метод узловых потенциалов достаточно просто программировать.

Предусматривается замена каждой ветви схемы линейным источником тока, который представляет собой параллельно включенные идеальный источник тока и проводимость $G = \text{const}$ (рис. 2.2). В этом случае уравнение ветви выглядит следующим образом:

$$I = GU + I_q \quad (2.1-1)$$

Если в ветви присутствует только проводимость, то $I_q = 0$, а активное сопротивление $R = \text{const}$ можно преобразовать в проводимость $G = 1/R$. Если ветвь содержит идеальный источник тока, то $G = 0$.

Линейный источник напряжения может быть преобразован в линейный источник тока следующим образом:

$$I = -\frac{U_q}{R}, \quad (2.1-2)$$

$$G = \frac{1}{R}. \quad (2.1-3)$$

Преобразование идеального источника напряжения, у которого $R=0$, в линейный источник напряжения невозможно. Для начала опишем применение метода узловых потенциалов для расчета цепи, не содержащей идеальных источников напряжения. Затем, в § 2.2, покажем, каким образом учитываются идеальные источники напряжения.

В дальнейшем, как и на рис. 2.2, будем пользоваться принятой системой стрелочных указателей, причем для каждого двухполюсника направления стрелок должны совпадать с направлением тока и полярностью напряжения. В этом случае при $P=UI>0$ двухполюсник пассивен, т.е. потребляет электрическую энергию (нагрузка), а при $P=UI<0$ — активен, т.е. генерирует электрическую энергию.

Направление тока и полярность напряжения в ветви, обозначенные стрелками, позволяют описать схему в виде графа: говорят, что ветвь начинается в узле x и оканчивается в узле y (рис. 2.3.). Это имеет значение при

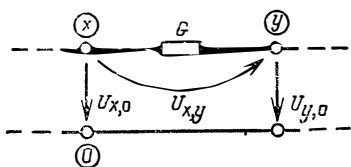


Рис. 2.3. Расчетная схема напряжения в ветви на основе узловых потенциалов

вводе номеров ветвей в процессе выполнения вычислений. Сначала всегда задается номер узла, в котором ветвь начинается, а затем номер узла, в котором ветвь заканчивается.

Ток и напряжение (I_q и U_q соответственно) считаются положительными, если их направления совпадают с направлениями тока и напряжения в ветви. В противном случае параметры источника считаются отрицательными.

При анализе электрических цепей методом узловых потенциалов сначала выбирается базовый узел, потенциал которого $\varphi_0=0$. Этому узлу присваивается номер 0, а остальные узлы нумеруются по порядку. Каждое напряжение может быть описано как разность потенциалов. При этом узловое напряжение между узлом x и базовым узлом определяется в виде

$$U_{x,0} = \varphi_x - \varphi_0. \quad (2.1-4)$$

где φ_x — потенциал узла x .

При $\varphi_0=0$ узловое напряжение

$$U_{x,0} = \varphi_x. \quad (2.1-5)$$

Число уравнений n для цепи без идеальных источников напряжения и базового узла равно числу узлов k_n . С помощью узлового напряжения можно рассчитать напряжение в каждой ветви. Например, напряжение между точками x и y на рис. 2.3

$$U_{x,y} = U_{x,0} - U_{y,0}. \quad (2.1-6)$$

В дальнейшем для определения узловых напряжений, напряжений и токов в ветвях будем использовать обозначения параметров цепи.

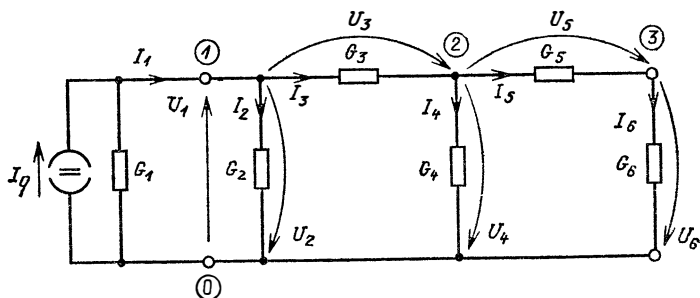


Рис. 2.4. Цепь постоянного тока с шестью ветвями и тремя узлами

Пример 2.1. Методом узловых потенциалов составим систему уравнений для цепи, изображенной на рис. 2.4. Уравнения для узлов 1—3 выглядят следующим образом:

$$I_2 + I_3 - I_1 = 0,$$

$$I_4 + I_5 - I_3 = 0,$$

$$I_6 + I_5 = 0.$$

Заменим ток в каждой ветви, используя для этого уравнение (2.1-1), тогда

$$G_2 U_2 + G_3 U_3 - G_1 U_1 - I_q = 0,$$

$$G_4 U_4 + G_5 U_5 - G_3 U_3 = 0,$$

$$G_6 U_6 - G_5 U_5 = 0.$$

Напряжение в каждой ветви можно выразить через узловые напряжения:

$$U_1 = -U_{1,0}; \quad U_2 = U_{1,0};$$

$$U_3 = U_{1,0} - U_{2,0}; \quad U_4 = U_{2,0};$$

$$U_5 = U_{2,0} - U_{3,0}; \quad U_6 = U_{3,0}.$$

Подставляя полученные выражения в узловые уравнения и объединяя последние, получаем

$$(G_1 + G_2 + G_3) U_{1,0} - G_3 U_{2,0} = I_q,$$

$$-G_3 U_{1,0} + (G_3 + G_4 + G_5) U_{2,0} - G_5 U_{3,0} = 0,$$

$$-G_5 U_{2,0} + (G_5 + G_6) U_{3,0} = 0.$$

Запишем теперь систему уравнений в матричной форме:

	1	2	3			
1	$G_1 + G_2 + G_3$	$-G_3$		\cdot	$\begin{bmatrix} U_{1,0} \\ U_{2,0} \\ U_{3,0} \end{bmatrix}$	$\begin{bmatrix} I_q \\ 0 \\ 0 \end{bmatrix}$
2	$-G_3$	$G_3 + G_4 + G_5$	$-G_5$			
3		$-G_5$	$G_5 + G_6$			

Матрицу, содержащую проводимости, назовем матрицей узловых проводимостей. Для двухполюсной цепи она симметрична. Структура матрицы подчиняется определенным правилам [12]: в каждом элементе главной диагонали матрицы стоит сумма проводимостей, которые одним своим полюсом соединены с соответствующим узлом. Во всех остальных элементах размещается отрицательная сумма тех проводимостей, которые расположены между узлами. Номера проводимостей в матрице определяются соответствующими номерами строк и столбцов. Матрица с одним столбцом токов источников строится также согласно строгим правилам: ток I_q источника получает отрицательный знак для того узла, из которого он «выходит», и положительный для узла, в который он «входит».

Вследствие простой структуры матрица может быть рассчитана по программе, приведенной в § 2.5.

2.2. Расширенный анализ узлов

Приведенная математическая модель расчета электрических цепей методом узловых потенциалов обладает следующими недостатками: идеальное короткое замыкание ($R=0$) между двумя узлами недопустимо, так как соответствующая проводимость ветви $G \rightarrow \infty$; использование идеального источника напряжения в качестве единственного элемента ветви не допускается.

Предложенную математическую модель можно расширить таким образом, что в любой ветви становятся допустимыми идеальные источники напряжения. Иногда удастся рассчитать и идеальное короткое замыкание, если его рассматривать как включение идеального источника напряжения с $U_q=0$. Такая модель получила название метода расширенного анализа узлов [26], [37].

2.2.1. Зависимые и независимые узлы. Если идеальный источник напряжения является единственным элементом ветви, то напряжение ветви при анализе цепи считается известным. В этом случае для расчета электрической цепи нет необходимости вычислять узловое напряжение, оно может быть определено из напряжения другого узла.

Преимуществом метода расширенного анализа узлов является то, что каждый идеальный источник описывается меньшим количеством уравнений и, следовательно, для его расчета требуется меньше машинного времени. Каждый узел, в котором начинается ветвь с идеальным источником напряжения, называется зависимым узлом, а все остальные узлы — независимыми.

Для решения системы уравнений вычисляются узловые напряжения независимых узлов. Затем определяют напряжения зависимых узлов. На рис. 2.5 в качестве примера показана ветвь между зависимым a и независимым u узлами. Узловое напряжение $U_{a,0}$ зависимого узла вычисляется с помощью узлового напряжения $U_{u,0}$ независимого узла, т. е.

$$U_{a,0} = U_{u,0} + U_q.$$

Деление узлов на зависимые и независимые осуществляется по определенным правилам:

а) узловые напряжения зависимых узлов должны однозначно определяться узловыми напряжениями независимых узлов. Поэтому из каждого зависимого узла

может начинаться только одна ветвь с идеальным источником напряжения. Таким образом, недопустимо, чтобы два идеальных источника напряжения начинались из одного и того же узла (рис. 2.6, а). Обойти это затруднение можно, изменив полярность источников напряжения

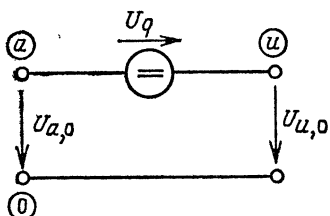


Рис. 2.5. Цепь с идеальным источником напряжения

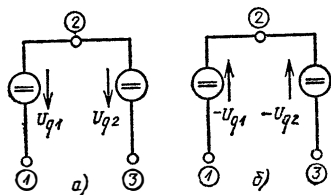


Рис. 2.6. Недопустимое расположение идеальных источников напряжения (а) и допустимая схема замещения (б)

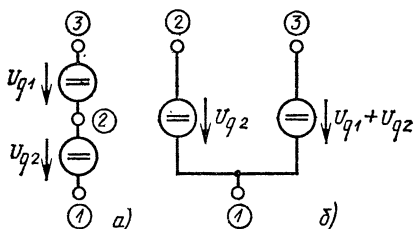


Рис. 2.7. Недопустимое последовательное соединение идеальных источников напряжения (а) и допустимая схема замещения (б)

Рис. 2.8. Идеальный источник напряжения, подключенный к базовому узлу. Недопустимое подключение (а) и допустимая схема замещения (б)

ния. На рис. 2.6, б узлы 1 и 3 являются зависимыми, а узел 2 — независим;

б) узел не должен быть одновременно зависимым и независимым. Подобное возможно, например, при последовательном соединении двух идеальных источников напряжения (рис. 2.7, а). В этом случае узлы 2 и 3 объявляются зависимыми узлами (рис. 2.7, б);

в) поскольку потенциал базового узла известен, этот узел можно считать независимым. Ветвь с идеальным источником напряжения не может начинаться в базовом

узле. В противном случае необходимо изменить полярность источников напряжения (рис. 2.8).

В общем случае недопустимое расположение источников может быть преобразовано в допустимое. Если же в некоторых специальных случаях этого сделать не удастся, значит, речь идет о ветвях, не имеющих физического смысла. Не имеет физического смысла, например, параллельное включение двух различных источников напряжения, поскольку при этом не выполняется первое правило. В ряде случаев может не иметь физического смысла и расположение идеальных источников тока.

Схема, в которой один узел связан с другими узлами исключительно через идеальный источник тока, имеет физический смысл только тогда, когда для этого узла выполняется закон узловых потенциалов.

Другие ограничения относительно структуры цепей определяются тем, что идеальный источник напряжения не может функционировать в условиях короткого замыкания, а идеальный источник тока — в условиях холостого хода [12].

2.2.2. Перемещение идеальных источников напряжения. С помощью уравнений ветвей для идеальных источников напряжения из совокупности узловых уравнений можно исключить уравнения зависимых узлов за счет перемещения каждого идеального источника в другие ветви, соединенные с зависимыми узлами. К каждому зависимому узлу подключается один источник напряжения (см. п. 2.2.1). Ветви, содержащие только активное сопротивление, называются нагрузочными ветвями.

Идеальный источник напряжения может быть перемещен во все соединенные с зависимым узлом нагрузочные ветви. Узловые напряжения независимых узлов и токи в ветвях от этого не изменятся. Изменение узлового напряжения в зависимом узле не имеет значения, так как этот узел после перемещения объединяется с соответствующим независимым узлом.

На рис. 2.9 показана схема электрической цепи, в которой источник с собственным напряжением U_{q4} перемещается в ветви 1 и 2. При этом исчезает зависимый узел 1.

После подобного перемещения идеальный источник напряжения с помощью проводимости ветви, подключенной к зависимому узлу, преобразуется в соответствующую

щий линейный источник тока. Аналогично можно поступать и в случае, когда ветвь содержит проводимость $G=0$.

Перемещая идеальные источники напряжения, получают цепь, систему уравнений которой с симметричной матрицей коэффициентов можно составить по аналогии

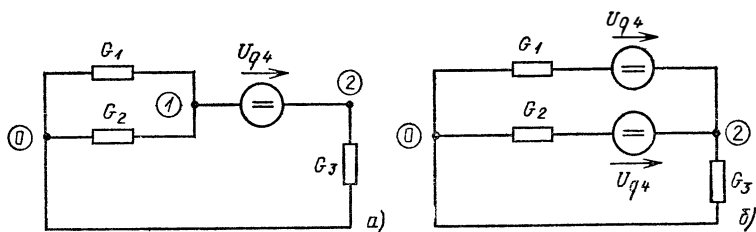


Рис. 2.9. Цепь до (а) и после (б) переноса идеального источника напряжения с собственным напряжением U_{q4}

с приведенным в § 2.1 описанием. Перемещение источников осуществляется с помощью специальной программы. При этом необходимо выполнить следующие действия: сначала определить, в какие ветви требуется переместить идеальный источник напряжения; затем в каждой ветви рассчитать соответствующий линейный источник тока, параметры которого добавляются к остальным параметрам ветви; после вычисления узлового напряжения независимых узлов вычислить узловые напряжения зависимых узлов.

2.2.3. Массив узлов. Количество уравнений, описывающих электрическую цепь, равно числу независимых узлов.

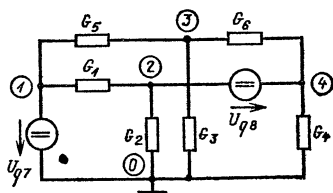
Расчет цепи, содержащей идеальные источники напряжения, выполняется гораздо быстрее, если число уравнений будет меньше, чем отношение числа узлов к числу идеальных источников напряжения.

Номер строки системы уравнений в общем случае не согласуется с номером узла цепи. Поэтому взаимосвязь между номерами строк и узлов в рассматриваемой ниже программе анализа цепей описывается с помощью массива $KF(KN)$, число элементов которого равно числу узлов цепи. Этот массив будем называть массивом узлов. Каждый элемент $KF(I)$ массива узлов содержит номер уравнения, которое соответствует узлу I . Рассмотрим

это утверждение на примере электрической цепи, представленной на рис. 2.10.

Узлы 1 и 2 цепи являются зависимыми, а узлы 3 и 4 — независимыми. Узловые уравнения для этих независимых узлов образуют систему с симметричной матрицей коэффициентов. Таким образом определяются элементы $KF(3)=1$ и $KF(4)=2$ массива узлов для схемы на рис. 2.10. Вследствие перемещения идеальных источников напряжения узлы 0 и 1, а также 2 и 4 объединя-

Рис. 2.10. Цепь с двумя идеальными источниками напряжения



ются. Поэтому элемент $KF(2)$ содержит тот же номер уравнения, что и элемент $KF(4)$, т.е. $KF(2)=2$.

Элемент $KF(1)$ содержит номер базового узла, т.е. $KF(1)=0$.

В программе, связывающей номера строк и узлов цепи, необходимо для базового узла предусмотреть элемент массива $KF(0)=0$.

Таким образом, массив KF для расчета схемы на рис. 2.10 содержит следующие номера уравнений: $KF(0)=0$; $KF(1)=0$; $KF(2)=2$; $KF(3)=1$; $KF(4)=2$. Содержимое массива узлов формируется с помощью подпрограммы UPR 2100. Элементы массива нумеруются в порядке их возрастания, однако в том случае, когда номер первого уравнения совпадает с номером $KA(K)$, нумерация прерывается (строки с 2106-й по 2120-ю). После этого каждый элемент массива, являющийся зависимым узлом содержит номер строки соответствующего независимого узла (с 2124-й по 2132-ю).

Подпрограмма UPR 2100

Описание идентификаторов:

а) используемые переменные вызывающей программы:

IS — количество идеальных источников напряжения;

KA(IS) — номер зависимого узла;

KU(IS) — номер независимого узла;

б) используемые имена переменных:

I, J, K — переменные циклов;
 в) результат:
 KF(KN) — массив узлов.
 2100 REM : заполнение массива узлов KF
 2102 REM
 2104 J=0
 2106 FOR I=1 TO KN
 2108 IF IS=0 THEN 2116
 2110 FOR K=1 TO 15
 2112 IF KA(K)=1 THEN 2120
 2114 NEXT K
 2116 J=J+1
 2118 KF(I)=J
 2120 NEXT I
 2122 IF IS=0 THEN 2134
 2124 FOR I=1 TO 15
 2126 J=KA(I)
 2128 K=KU(I)
 2130 KF(J)=KF(K)
 2132 NEXT
 2134 RETURN •

2.3. Цепи с линейным четырехполюсником

2.3.1. Цепи с пассивным четырехполюсником. Для линейного четырехполюсника (рис. 2.11) взаимосвязь между параметрами цепи U_1, I_1 на входе и U_2, I_2 на выходе

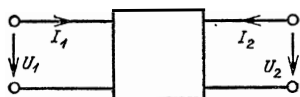


Рис. 2.11. Токи и напряжения четырехполюсника

с использованием проводимостей описывается следующим образом:

$$\left. \begin{aligned} I_1 &= Y_{11} U_1 + Y_{12} U_2, \\ I_2 &= Y_{21} U_1 + Y_{22} U_2. \end{aligned} \right\} \quad (2.3-1)$$

С помощью сопротивлений связь между параметрами цепи на входе и выходе четырехполюсника записывается в виде

$$\left. \begin{aligned} U_1 &= Z_{11} I_1 + Z_{12} I_2, \\ U_2 &= Z_{21} I_1 + Z_{22} I_2. \end{aligned} \right\} \quad (2.3-2)$$

Если линейный пассивный четырехполюсник содержит исключительно активные сопротивления или прово-

димости, то он является симметрично передающим элементом для любой нагрузки Z_1 , подключенной к входу четырехполюсника, и Z_2 , подключенной к его выходу: рабочий коэффициент передачи в обоих направлениях одинаков [12]. При этом Z_1 или Z_2 могут быть как внутренними сопротивлениями источника, так и сопротивлениями нагрузки. Симметрично передающий четырехполюсник полностью описывается с помощью всего лишь трех параметров.

Для проводимостей условие симметричности передачи выглядит следующим образом:

$$Y_{12} = Y_{21}, \quad (2.3-3)$$

а для сопротивлений

$$Z_{12} = Z_{21}. \quad (2.3-4)$$

В случае симметрично передающего четырехполюсника всегда может быть задана схема замещения из линейных двухполюсников. Если уравнения четырехполюсника приведены с учетом проводимостей, то в качестве схемы замещения целесообразно использовать П-образную схему (рис. 2.12). Проводимости этой схемы можно рас-

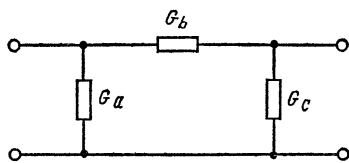


Рис. 2.12. П-образная схема замещения

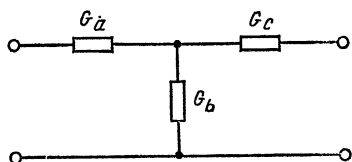


Рис. 2.13. Т-образная схема замещения

считать, используя параметры Y , следующим образом:

$$\left. \begin{aligned} G_a &= Y_{11} + Y_{12}, \\ G_b &= -Y_{12}, \\ G_c &= Y_{12} + Y_{22}. \end{aligned} \right\} \quad (2.3-5)$$

Если уравнения четырехполюсника заданы с помощью сопротивлений, то в качестве схемы замещения целесообразно использовать Т-образную схему. Проводимости такой схемы (рис. 2.13) можно рассчитать через

параметры Z :

$$\left. \begin{aligned} G_a &= 1/(Z_{11} - Z_{12}), \\ G_b &= 1/Z_{12}, \\ G_r &= 1/(Z_{22} - Z_{12}). \end{aligned} \right\} \quad (2.3-6)$$

2.3.2. Управляемые источники. Для расчета активно-го четырехполюсника исходят в принципе из тех же предположений, что и для пассивного четырехполюсника, т. е. заменяют его схемой замещения из линейных двухполюсников. Таким образом, любая схема с четырехполюсниками представляется и рассчитывается как схема из двухполюсников.

Каждый линейный активный четырехполюсник может быть заменен линейным управляемым источником тока или напряжения. В качестве управляющего параметра используется ток или напряжение ветви, причем управляемая ветвь является либо линейным источником тока (рис. 2.14), либо линейным источником напряжения (рис. 2.15).

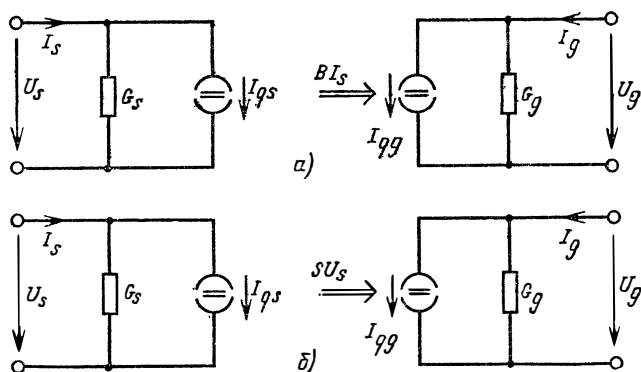


Рис. 2.14. Схема замещения управляемых источников тока:

a — источник тока, управляемый током; *б* — источник тока, управляемый напряжением

В зависимости от вида источника и управляющего параметра различают четыре основных типа управляемых источников. На рис. 2.14 и 2.15 управляющая ветвь обозначена индексом s , а управляемая — индексом g . В сокращениях, принятых в литературе, перед буквой Q ,

обозначающей источник, ставят сначала букву, соответствующую управляющему параметру, а затем букву, соответствующую виду источника (табл. 2.1).

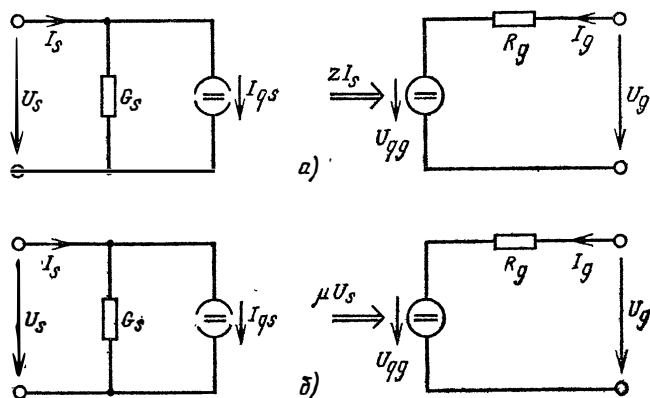


Рис. 2.15. Схема замещения управляемых источников напряжения:
а — источник напряжения, управляемый током; б — источник напряжения, управляемый напряжением

Т а б л и ц а 2.1. Управляемые источники

Вид источника	Параметры источника	Коэффициент управления
Источник (генератор) тока, управляемый током (IIQ)	$I_{qg} = B I_s$	$B = 1$
Источник (генератор) тока, управляемый напряжением (UIQ)	$I_{qg} = S U_s$	$S = 1 \text{ См}$
Источник (генератор) напряжения, управляемый током (IUQ)	$U_{qg} = Z I_s$	$Z = 1 \text{ Ом}$
Источник (генератор) напряжения, управляемый напряжением (UUQ)	$U_{qg} = \mu U_s$	$\mu = 1$

Собственный параметр линейно управляемого источника благодаря постоянному коэффициенту управления зависит от управляющего параметра линейно. Поскольку в дальнейшем будут рассматриваться исключительно управляемые источники напряжения с $R_g > 0$, каждый управляемый источник напряжения перед анализом це-

пи может быть преобразован в управляемый источник тока.

Источник напряжения, управляемый током (рис. 2.15, а), имеет смысл преобразовывать в источник тока, управляемый током, проводимость которого (рис. 2.14, а) определяется из выражения

$$G_g = 1/R_g. \quad (2.3-7)$$

Согласно уравнению (2.1-2) $I_{qg} = -U_{qg}/R_g$. С учетом равенства $U_{qg} = ZI_s$ и уравнения (2.3-7) получаем

$$I_{qg} = -ZG_g I_s. \quad (2.3-8)$$

Источник напряжения, управляемый током, можно заменить источником тока, управляемым током, с проводимостью $G_g = 1/R_g$ и коэффициентом управления $B = -ZG_g$. Этот коэффициент в общем случае называют коэффициентом усиления по току B .

Источник напряжения, управляемый напряжением (рис. 2.15, б), имеет смысл преобразовать в источник тока, управляемый напряжением, для которого проводимость определяется из выражения (2.3-7). Согласно уравнению (2.1-2) с учетом равенства $U_{qg} = \mu U_s$ и уравнения (2.3-7) получаем

$$I_{qg} = -\mu G_g U_s. \quad (2.3-9)$$

Источник напряжения, управляемый напряжением, заменяется источником тока, управляемым напряжением, с проводимостью $G_g = 1/R_g$ и коэффициентом управления $S = -\mu G_g$.

Пример 2.2. Операционный усилитель на рис. 2.16 может быть представлен источником напряжения, управ-

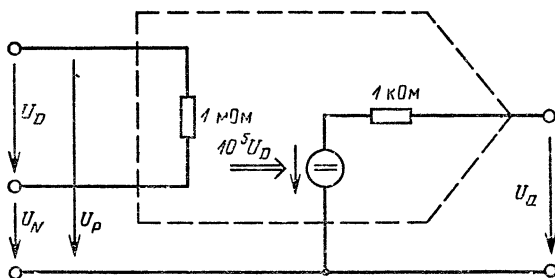


Рис. 2.16. Операционный усилитель, представленный источником напряжения, управляемого напряжением [55]

ляемым по напряжению, коэффициент управления которого μ равен коэффициенту усиления u . Преобразуем этот источник в источник тока, управляемый напряжением (рис. 2.14, б).

Из уравнения (2.3-7) $G=1$ мСм. С учетом уравнения (2.3-9) получаем

$$S = -10^5 \cdot 10^{-3} \text{ См} = -100 \text{ См}.$$

В управляющей ветви с напряжением

$$U_s = U_D = U_p - U_n$$

находится только проводимость $G_s = 1$ мкСм, причем $I_{qs} = 0$ (рис. 2.17). Управляющая ветвь с напряжением

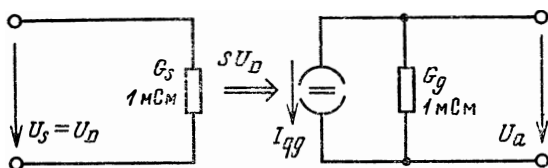


Рис. 2.17. Схема замещения операционного усилителя на рис. 2.16

$U_g = U_a$ состоит из линейного источника тока с собственным значением $I_{qg} = -100 \text{ См} \cdot U_s$ и проводимости $G_g = 1$ мСм.

2.3.3. Составление системы уравнений. Если цепь содержит один управляемый источник, то параметры управляемой ветви зависят от параметров управляющей ветви. Для однозначного решения задачи схемного анализа предположим, что управляющая ветвь не содержит ни одного управляемого источника: ток источника I_{qs} в управляющей ветви, таким образом, должен быть неуправляемым.

В системе уравнений цепи с одним управляемым источником тока в матрице токов, т. е. в правой части системы уравнений, стоит неизвестная величина $I_{qg} = B I_s$ или $I_{qg} = S U_s$ (см. рис. 2.14). В этой форме система уравнений не может быть решена. Однако для источника тока, управляемого напряжением, напряжение U_s управляющей ветви выражается как разность двух узловых напряжений. Таким образом, произведение $S U_s$ может быть перенесено в левую часть системы уравнений. Рассмотрим это на примере.

Пример 2.3. Составим систему уравнений для анализа цепи, показанной на рис. 2.18. Если ток источника I_{qg} задан, то можно записать

$$\begin{bmatrix} G_1 + G_2 & -G_2 & \\ -G_2 & G_2 + G_3 & -G_3 \\ & -G_3 & G_3 + G_4 \end{bmatrix} \cdot \begin{bmatrix} U_{1,0} \\ U_{2,0} \\ U_{3,0} \end{bmatrix} = \begin{bmatrix} -I_{q1} \\ -SU_2 \\ SU_2 \end{bmatrix}.$$

Теперь заменим напряжение ветви U_2 узловым напряжением $U_2 = U_{1,0} - U_{2,0}$, а выражение $SU_2 = SU_{1,0} - SU_{2,0}$

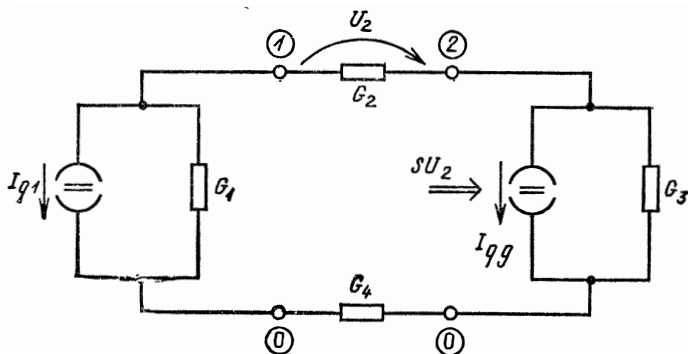


Рис. 2.18. Цепь с источником тока, управляемым напряжением

перенесем в левую часть системы уравнений и в результате получим

$$\begin{bmatrix} G_1 + G_2 & -G_2 & \\ S - G_2 & G_2 + G_3 - S & -G_3 \\ -S & S - G_3 & G_3 + G_4 \end{bmatrix} \cdot \begin{bmatrix} U_{1,0} \\ U_{2,0} \\ U_{3,0} \end{bmatrix} = \begin{bmatrix} -I_{q1} \\ 0 \\ 0 \end{bmatrix}.$$

При такой записи после подстановки численных значений параметров схемы можно найти решение системы уравнений.

Для источника (генератора) тока, управляемого током (рис. 2.14, а), поступают аналогичным образом, так как ток в управляющей ветви

$$I_s = G_s U_s + I_{qs} \quad (2.3-10)$$

всегда выражается через напряжение U_s . Ток источника

$$I_{qs} = BI_s = BG_s U_s + BI_{qs} \quad (2.3-11)$$

в качестве единственного неизвестного параметра содер-

жит напряжение управляющей ветви U_s , которое описывается как разность двух узловых напряжений и переносится в левую часть системы уравнений.

Пример 2.4. Биполярный транзистор в активной области можно приближенно заменить линейным источником тока, управляемым током, и его схема замещения на рис. 2.19 показана

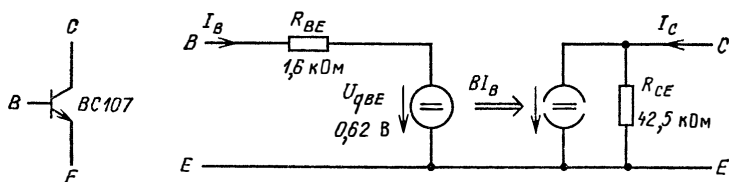


Рис. 2.19. Транзистор BC 107, представленный источником тока, управляемого током, и его схема замещения

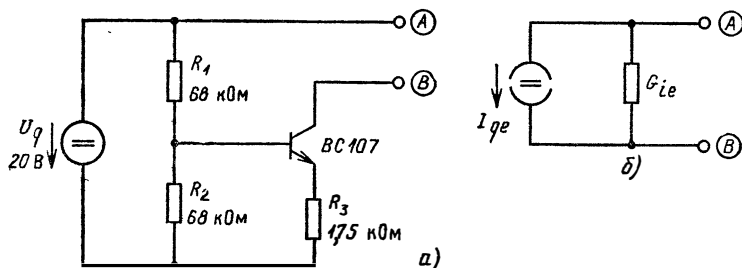


Рис. 2.20. Схема стабилизированного источника тока (а) и его схема замещения (б)

схема замещения транзистора BC107 при токе коллектора 5 мА. На основе этого транзистора строится простая схема, принцип действия которой напоминает источник постоянного тока. На рис. 2.20 приведены схема источника постоянного тока при $I_C = 5$ мА и его схема замещения. Составим систему уравнений расчета схемы на рис. 2.20 для случая, когда клеммы А и В замкнуты накоротко. Транзистор на рис. 2.20 замещается линейной схемой согласно рис. 2.19. Линейный источник напряжения (U_{qBE} , R_{BE}) преобразуем в линейный источник тока с параметрами

$$I_{q4} = -U_{qBE}/R_{BE}; \quad G_4 = 1/R_{BE}.$$

Идеальный источник напряжения с напряжением U_q перенесем в нагрузочные ветви, которые соединены с независимыми узлами (рис. 2.21).

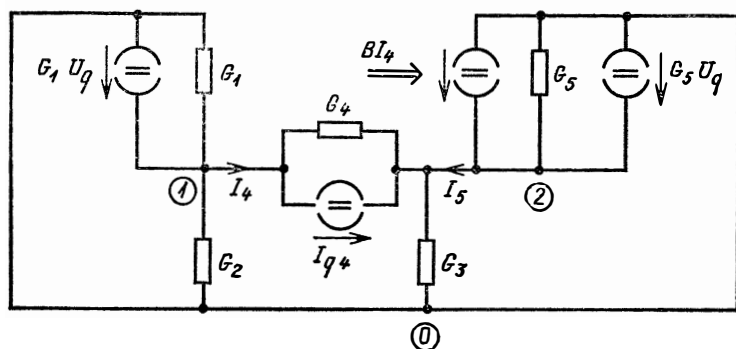


Рис. 2.21. Линейная схема замещения стабилизированного источника тока с короткозамкнутыми узлами А и В (см. рис. 2.20)

Для проводимости $G_5 = G_{CE}$ приведем систему уравнений с учетом того, что элемент BI_4 является источником тока:

$$\left| \begin{array}{cc} G_1 + G_2 + G_4 & -G_4 \\ -G_4 & G_3 + G_4 + G_5 \end{array} \right| \cdot \begin{bmatrix} U_{1,0} \\ U_{2,0} \end{bmatrix} = \begin{bmatrix} G_1 U_q - I_{q4} \\ I_{q4} + BI_4 + G_5 U_q \end{bmatrix}.$$

Теперь заменим ток ветви I_4 с помощью узлового напряжения и получим

$$BI_4 = BG_4(U_{1,0} - U_{2,0}) + BI_{q4}.$$

Сокращенно запишем: $B_1 = 1 + B$ и соответственно

$$\left| \begin{array}{cc} G_1 + G_2 + G_4 & -G_4 \\ -B_1 G_4 & G_3 + B_1 G_4 + G_5 \end{array} \right| \cdot \begin{bmatrix} U_{1,0} \\ U_{2,0} \end{bmatrix} = \begin{bmatrix} G_1 U_q - I_{q4} \\ B_1 I_{q4} + G_5 U_q \end{bmatrix}.$$

После задания численных значений элементов схемы можно найти решение системы.

2.4. Решение системы линейных уравнений

2.4.1. Алгоритм Гаусса. Для решения системы линейных уравнений существует несколько методов [31]. Алгоритм Гаусса называют также методом исключений, по-

тому что неизвестные X_1, X_2, \dots, X_{n-1} поочередно исключаются до тех пор, пока не останется одно уравнение с неизвестной X_n . После решения этого уравнения путем обратной подстановки определяются неизвестные $X_{n-1}, X_{n-2}, \dots, X_1$. Матрица с коэффициентами a_{jk} называется матрицей коэффициентов:

$$\begin{array}{|c|c|c|c|c|} \hline a_{11} & a_{12} & a_{13} & \dots & a_{1n} \\ \hline a_{21} & a_{22} & a_{23} & \dots & a_{2n} \\ \hline a_{31} & a_{32} & a_{33} & \dots & a_{3n} \\ \hline \dots & & & & \dots \\ \hline a_{n1} & a_{n2} & a_{n3} & \dots & a_{nn} \\ \hline \end{array} \begin{array}{|c|} \hline X_1 \\ \hline X_2 \\ \hline X_3 \\ \hline \dots \\ \hline X_n \\ \hline \end{array} = \begin{array}{|c|} \hline b_1 \\ \hline b_2 \\ \hline b_3 \\ \hline \dots \\ \hline b_n \\ \hline \end{array}. \quad (2.4-1)$$

В уравнении (2.4-1) однострочковая матрица с неизвестными X_j является вектором решения. Для исключения переменной X_1 во второй строке системы уравнений в алгоритме Гаусса первая строка умножается на коэффициент

$$r_{21} = a_{21}/a_{11} \quad (2.4-2)$$

и после преобразований выглядит следующим образом:

$$a_{21} X_1 + r_{21} a_{12} X + \dots + r_{21} a_{1n} X_n = r_{21} b_1.$$

Полученное уравнение вычитается из второй строки:

$$(a_{22} - r_{21} a_{12}) X_2 + \dots + (a_{2n} - r_{21} a_{1n}) X_n = b_2 - r_{21} b_1.$$

Соответственно умноженная на коэффициент

$$r_{31} = a_{31}/a_{11} \quad (2.4-3)$$

первая строка вычитается из третьей строки системы. В результате и третья строка не содержит теперь переменной X_1 . Если все остальные уравнения обработать аналогичным образом, то неизвестная X_1 исключается также и из строк со второй по n -ю систему. Этот процесс называют редукцией, а систему уравнений с $n-1$ неизвестной — редуцированной системой.

Обозначим количество исключенных переменных верхним индексом элементов матрицы коэффициентов правой части, т. е.

$$a_{22}^{(1)} = a_{22} - r_{21} a_{12},$$

$$a_{2n}^{(1)} = a_{2n} - r_{21} a_{1n},$$

$$b_2^{(1)} = b_2 - r_{21} b_1.$$

Теперь из редуцированной системы уравнений исключим следующую переменную. Процесс повторяется до тех пор, пока не останется одно уравнение с одной неизвестной.

Для экономии памяти ЭВМ каждое редуцированное уравнение записывается in place, т. е. в ту ячейку памяти, где хранились элементы матрицы системы уравнений перед выполнением каждой редукции.

Таким образом, после $n-1$ редукций получаем следующую систему уравнений:

$$\begin{array}{|c|c|c|c|c|} \hline a_{11} & a_{12} & a_{13} & \dots & a_{1n} \\ \hline & a_{23}^{(1)} & a_{23}^{(1)} & \dots & a_{2n}^{(1)} \\ \hline & & a_{33}^{(2)} & \dots & a_{3n}^{(2)} \\ \hline & & & & \dots \\ \hline & & & & a_{nn}^{(n-1)} \\ \hline \end{array} \cdot \begin{array}{|c|} \hline X_1 \\ \hline X_2 \\ \hline X_3 \\ \hline \dots \\ \hline X_n \\ \hline \end{array} = \begin{array}{|c|} \hline b_1 \\ \hline b_2^{(1)} \\ \hline b_3^{(2)} \\ \hline \dots \\ \hline b_n^{(n-1)} \\ \hline \end{array} \quad (2.4-4)$$

Матрица, содержащая коэффициенты a_{jk} на разных шагах редуцирования системы, называется треугольной, поскольку элементы расположены только по главной диагонали и над ней. В действительности элементы есть и под главной диагональю, однако в дальнейших расчетах они не используются. Разумеется, хранение in place разрушает как исходную матрицу коэффициентов, так и изначально существующую матрицу правой части системы.

После $n-1$ редукций системы можно вычислить вектор решения. Расчет начинают в n -й строке системы (2.4-4) и вычисляют

$$X_n = b_n^{(n-1)} / a_{nn}^{(n-1)}. \quad (2.4-5)$$

Затем x_n умножают на коэффициент в строке $n-1$ столбца n матрицы коэффициентов и полученное произведение вычитают из соответствующего элемента матрицы правой части системы. После деления результата предыдущих действий на соответствующий элемент главной диагонали матрицы коэффициентов определяют x_{n-1} .

2.4.2. Упорядочение системы уравнений. При редуцировании системы уравнений одна строка каждый раз ис-

пользуется для того, чтобы сократить соответствующую неизвестную. Эту строку называют строкой упорядочения, а коэффициенты неизвестных, которые должны быть сокращены, — элементами упорядочения. Например, для описанного в п. 2.4.1 исключения X_1 коэффициент a_{11} является элементом упорядочения.

Для вычисления коэффициента редуцирования r_{ji} делят на элемент упорядочения a_{ii} (см. выше), поэтому он не может быть равен нулю.

В качестве элемента упорядочения имеет смысл выбирать наибольший по модулю элемент строки или столбца матрицы коэффициентов. В результате замен в столбце или строке этот элемент должен оказаться на главной диагонали матрицы. Весь процесс называют упорядочением системы уравнений.

Если метод узловых потенциалов используется для двухполюсников, не содержащих управляемых источников, то наибольший элемент строки каждый раз попадает на главную диагональ матрицы узловых проводимостей, являющейся одновременно матрицей коэффициентов системы линейных уравнений. Кроме того, каждый элемент на главной диагонали больше нуля. Такую матрицу называют положительно определенной матрицей.

При каждой редукции положительно определенной матрицы вновь получаем положительно определенную матрицу, поэтому упорядочение системы линейных уравнений цепи двухполюсников, не содержащей управляемых источников, не требуется. Предполагается, что рассчитываемая цепь имеет физический смысл. Не имеющей физического смысла является, например, цепь, к узлу которой подключены только идеальные источники тока и ни одной проводимости (см. п. 2.2.1). В этом случае в главной диагонали «стоял» бы нуль. Если метод узловых потенциалов используется для цепи, содержащей управляемые источники, то матрица узловых потенциалов и соответственно матрица коэффициентов системы линейных уравнений не для каждого случая определены положительно, т. е. нельзя гарантировать, что наибольший по модулю член каждой строки матрицы коэффициентов стоит на главной диагонали. Кроме того, в особом случае из-за управляемых источников на главной диагонали может оказаться элемент, близкий или равный нулю. В примере 2.4 это предполагается для $S = G_2 + G_3$.

При редукции системы линейных уравнений значения каждого элемента главной диагонали должны быть больше некоторого ϵ (обычно $\epsilon = 10 \div 12$). Добиться того, чтобы ни один из элементов главной диагонали матрицы коэффициентов не был равен нулю, можно и не выполняя упорядочение всей матрицы коэффициентов, что требует некоторого усложнения программ и увеличения времени расчета. Значительно проще заранее обнаружить и исключить из рассмотрения описанный выше особый случай.

Для этого применяется только перестановка строк матрицы, а вектор решения остается неизменным.

В подпрограмме UPR 2300 каждый элемент на главной диагонали матрицы коэффициентов сравнивается с предельным значением ϵ , которое на БЭЙСИКе обозначено EP (строка 2306). Если он меньше ϵ , то обычно в столбце, в котором стоит этот элемент, отыскивают наибольший по модулю элемент и соответствующие строки меняют местами.

Однако в результате элемент, по модулю меньший ϵ , может вновь оказаться на главной диагонали. Рассмотрим этот случай на простом примере. Имеем матрицу

7	-4	-5
-1	6	-2
0	3	0

Найдем для нее элемент главной диагонали, который по модулю меньше ϵ . Таким элементом является 0 в строке 3. В этом столбце наибольший по модулю элемент находится в строке 1. Если же переставить строки 1 и 3, то элемент, равный нулю, вновь окажется на главной диагонали. Поэтому в подпрограмме UPR 2300 будем использовать более эффективную «стратегию» поиска; вычислим произведение обоих элементов, которые после перестановки двух строк оказались бы на главной диагонали, и значение произведения элементов AI (строка 2314).

Из n возможных произведений определим максимальное R1 и соответствующий ему номер строки J1. Переставим эту строку J1 и строку J (строки с 2326-й по 2340-ю).

В нашем примере будем переставлять строку с номером I=3. Если переставить строки J=1 и I=3, то элементы

$A(I, J) = 0$ и $A(J, I) = -5$ попадут на главную диагональ и произведение элементов будет $A1 = 0$. Перестановка строк $J=2$ и $I=3$ дает $A1=6$. Максимальным значением $A1$ является $R1=6$. Таким образом, переставлены должны быть строки $J1=2$ и $I=3$. В случае, если не найдено ни одной строки, в которой целесообразна перестановка, на печать выдается сообщение «Решение невозможно» и выполнение подпрограммы завершается.

Подпрограмма UPR 2300

а) необходимые переменные вызывающей программы:

$A(N, N)$ — матрица коэффициентов;

EP — предельное значение;

N — количество уравнений;

$X(N)$ — правая часть;

б) используемые имена переменных:

$A1$ — значение произведения (вспомогательная величина);

I, J — переменные цикла;

$J1$ — номер строки;

$R1$ — наибольшее значение произведения;

$X1$ — вспомогательная величина;

в) результат:

$A(N, N)$ — матрица коэффициентов;

$X(N)$ — правая часть.

2300 REM : Упорядочение системы уравнений

2302 REM

2304 FOR I=1 TO N

2306 IF ABS(A(I, I)) > EP THEN 2342

2308 J1=0

2310 R1=0

2312 FOR J=1 TO N

2314 A1=ABS(A(I, J) * A(J, I))

2316 IF A1 < R1 THEN 2322

2318 R1=A1

2320 J1=J

2322 NEXT

2324 IF R1 < EP THEN 2050

2326 X1=X(J1)

2328 X(J1)=X(I)

2330 X(I)=X1

2332 FOR J=1 TO N

2334 A1=A(J1, J)

2336 A(J1, J)=A(I, J)

2338 A(I, J)=A1

2340 NEXT J

2342 NEXT I

2344 RETURN

2.4.3. Система уравнений с несимметричной матрицей коэффициентов. Для решения системы линейных уравнений схемы, которая содержит управляемые источники ток или напряжения, составляется подпрограмма

UPR 200. Перед ее выполнением необходимо представить матрицу коэффициентов в виде массива $A(N, N)$, а матрицу правой части системы — в виде массива $X(N)$.

В строках с 2006-й по 2028-ю матрица коэффициентов и правая часть $n-1$ раз редуцируются. Решение одного уравнения ($n=1$) не требует редукции. Поэтому коэффициенты системы уравнений остаются неизменными. С помощью оператора условного перехода в строке 2016 уменьшается время расчета.

В строках 2030—2046 выполняется обратная подстановка. Для этого каждый (редуцированный) элемент правой части заменяется соответствующим элементом вектора решений. После выполнения UPR 2000 вектор решений записывается в массив $X(N)$.

Подпрограмма UPR 2000

а) необходимые переменные вызывающей программы:

$A(N, N)$; EP; N; $X(N)$ — пояснения см. в приложении ПЗ;

б) используемые имена переменных:

D1 — текущий элемент упорядочения (вспомогательная величина);

I — шаг редуцирования;

J, K — номер строки/столбца;

R1 — текущий коэффициент редуцирования;

в) результат: $X(N)$ — вектор решений.

2000 REM: Решение системы линейных уравнений

2002 REM

2004 IF N=1 TO 2030

2006 FOR I=1 TO N-1

2008 D1=A(I, I)

2010 IF ABS(D1)<EP THEN 2050

2012 FOR J=I+1 TO N

2014 R1=A(J, I)/D1

2016 IF R1=0 THEN 2026

2018 FOR K=I+1 TO N

2020 A(J, K)=A(J, K)-R1 * A(I, K)

2022 NEXT K

2024 X(J)=X(J)-R1 * X(I)

2026 NEXT J

2028 NEXT I

2030 FOR J=N TO 1 STEP -1

2032 IF ABS(A(J, J))<EP THEN 2050

2034 D1=X(J)

2036 IF J=N THEN 2044

2038 FOR K=N TO J+1 STEP -1

2040 D1=D1-A(J, K) * X(K)

2042 NEXT K

2044 X(J)=D1/A(J, J)

2046 NEXT J

2048 RETURN

2050 PRINT

2.4.4. Система уравнений с симметричной матрицей коэффициентов. Если электрическая схема не содержит управляемых источников, матрица узловых проводимостей является симметричной. Кроме того, в каждой строке наибольший по модулю элемент находится на главной диагонали. Поэтому матрица коэффициентов системы линейных уравнений является симметричной и положительно определенной.

Вследствие положительной определенности упорядочение системы уравнений не требуется. Симметрия матрицы коэффициентов может быть использована в алгоритме Гаусса для уменьшения времени редукции системы уравнений [49].

Если матрица симметрична, то для каждого элемента в общем случае

$$a_{jk} = a_{kj}. \quad (2.4-6)$$

При каждой редукции системы уравнений симметричность сохраняется. Покажем это на примере коэффициентов a_{23} и a_{32} при исключении X_1 :

$$a_{23}^{(1)} = a_{23} - r_{21} a_{13} = a_{23} - \frac{a_{21}}{a_{11}} a_{13}; \quad (2.4-7)$$

$$a_{32}^{(1)} = a_{32} - r_{31} a_{12} = a_{32} - \frac{a_{31}}{a_{11}} a_{12}. \quad (2.4-8)$$

В случае симметрии матрицы коэффициентов $a_{12} = a_{21}$; $a_{13} = a_{31}$; $a_{23} = a_{32}$. Отсюда

$$a_{32}^{(1)} = a_{23}^{(1)}, \quad (2.4-9)$$

т. е. нет необходимости вычислять значение a_{32} . Таким образом, расчет и запоминание коэффициентов a_{jk} при $j > k$ не требуются.

Другими словами, преимущество алгоритма Гаусса при редукции системы уравнений с симметричной матрицей коэффициентов заключается в том, что можно сэкономить почти половину вычислительного времени, необходимого для редукционирования системы уравнений с несимметричной матрицей.

Подпрограмма UPR 2200 решения системы линейных уравнений разработана на основе подпрограммы UPR 2000 с учетом двух изменений:

текущий коэффициент редукции вычисляется с помо-

щью коэффициента $A(I, J)$, который находится при $J > I$ всегда выше главной диагонали;

начальное значение переменной цикла K равно не $I+1$, как в подпрограмме UPR 2000, а J . Вследствие этого вычисляются только коэффициенты, лежащие на главной диагонали и выше нее.

Редукция правой части в подпрограмме UPR 2200 выполняется точно так же, как и в подпрограмме UPR 2000. В подпрограмме UPR 2200 определение неизвестных методом обратной подстановки осуществляется с помощью подпрограммы UPR 2000 (строка 2232), поэтому последняя должна всегда находиться в оперативной памяти ЭВМ.

Подпрограмма UPR 2200 (см. UPR 2000)

```
2200 REM: Решение системы линейных уравнений
2202 REM: с симметричной матрицей коэффициентов
2204 REM
2206 IF N=1 THEN 2232
2208 FOR I=1 TO N-1
2210 D1=A(I, I)
2212 IF ABS(D1)<EP THEN 2050
2214 FOR J=I+1 TO N
2216 R1=A(I, J)/D1
2218 IF R1=0 THEN 2228
2220 FOR K=J TO N
2222 A(J, K)=A(J, K)-R1 * A(I, K)
2224 NEXT K
2226 X(J)=X(J)-R1 * X(I)
2228 NEXT J
2230 NEXT I
2232 GOSUB 2030
2234 RETURN
```

2.5. Программа GLEICH на языке БЕЙСИК

С помощью программы GLEICH выполняется анализ линейной цепи постоянного тока, содержащей управляемые источники тока. Управляемые источники напряжения должны быть перед вводом в ЭВМ преобразованы в управляемые источники тока (см. п. 2.3.2).

2.5.1. Глобальные переменные. Для запуска программы вводятся глобальные переменные, которыми являются: IZ — количество ветвей; KN — количество узлов; IS — количество идеальных источников напряжения.

Базовый узел всегда имеет номер нуль и не учитывается в KN . С помощью введенных глобальных переменных

вычисляются количество N уравнений системы и число IR нагрузочных ветвей.

Для некоторых цепей, имеющих физический смысл, невозможно составить уравнения узловых потенциалов (например, последовательный контур с одним идеальным источником напряжения и одной проводимостью). Расчет таких цепей в программах на языке БЕЙСИК потребовал бы большого числа дополнительных запросов, поэтому его выполнение предусматривается только в программах на языке ПАСКАЛЬ.

2.5.2. Ввод в ЭВМ структуры схемы. Для каждой ветви I эта операция оформляется в виде подпрограммы UPR 2500, что позволяет достаточно просто программировать изменения параметров смены. Если ветвь I является нагрузочной ветвью ($I \leq IR$), то вводятся номер узла $KV(I)$, в котором начинается ветвь, и номер узла $KZ(I)$, где она заканчивается.

Элементами нагрузочной ветви являются: проводимость, идеальный и управляемый источники тока, включенные параллельно друг другу. Однако чаще в одной ветви присутствует только один из этих элементов. Поэтому для каждого элемента ветви перед его численным значением вводят специальную букву-идентификатор.

Для проводимости это буква G . Если вместо проводимости требуется ввести в машину соответствующее сопротивление, то используется буква R . Расчет проводимости при заданном сопротивлении осуществляется в подпрограмме (строка 2538). Для идеального источника тока используется буква I , а в случае управляемого источника тока — буква S . Ввод элементов ветви завершается буквой N ; после этого вводятся элементы следующей ветви. Такая расширенная система ввода применяется и в программах, описанных в гл. 3—6.

Возможна также корректировка вводимых значений текущей ветви I до того момента, пока не введена буква-идентификатор N : если для одной ветви вводятся несколько различных значений, то запоминается в программе последнее из них.

Ветви схемы нумеруются таким образом, чтобы ветви с сопротивлениями имели меньшие номера ($1 \leq I \leq IR$). В соответствии с этим идеальные источники должны соответствовать самым «старшим» номерам ветви. Такая нумерация необходима для того, чтобы нагрузочные ветви и идеальные источники напряжения при расчете цепи

могли обрабатываться разными операторами программы без дополнительных запросов.

Каждый идеальный источник напряжения начинается в зависимом узле KA(IL) и заканчивается в независимом узле KU(IL). С этой целью идеальные источники напряжения последовательно нумеруются от $IL=1$ до $IL=IS$, хотя всегда задается $I=IL+IR$ (IR — количество нагрузочных ветвей).

Подпрограмма UPR 2500 (она вызывает UPR 2400)

а) необходимые переменные вызывающей программы:

I — номер ветви;

IR — количество нагрузочных ветвей;

б) используемые имена переменных:

IL — номер идеального источника напряжения;

R1 — активное сопротивление;

в) результаты:

GD(I) — проводимость нагрузочной ветви;

IG — наличие управляемых источников;

KA(IL) — зависимые узлы;

KU(IL) — независимые узлы;

KV(I) — начальные узлы нагрузочных ветвей;

KZ(I) — конечные узлы нагрузочных ветвей;

MZ(I) — управляющие ветви управляемых источников тока;

QT(IL) — собственные напряжения идеальных источников напряжения;

SD(I) — токи источников;

SF(I) — управляющие коэффициенты или усиление по току.

2500 REM: Ввод элементов цепи для ветви I

2502 REM

2504 INPUT «от узла» ; J

2506 INPUT «к узлу» ; K

2508 IF I>IR THEN 2554

2510 KU(I)=J

2512 KZ(I)=K

2514 GD(I)=0

2516 SD(I)=0

2518 MZ(I)=0

2520 INPUT «Элемент» ; FR\$

2522 IF FB\$ = «R» THEN 2536

2524 IF FR\$ = «G» THEN 2542

2526 IF FR\$ = «I» THEN 2546

2528 IF FR\$ = «S» THEN 2550

2530 IF FR\$ = «N» THEN 2562

2532 PRINT «Ввод не имеет смысла»

2534 GOTO 2520

2536 INPUT «Сопротивление»; R1

2538 GD(I)=1/R1

2540 GOTO 2520

2542 INPUT «Индуктивность»; GD(I)

2544 GOTO 2520

2546 INPUT «Ток источника»; SD(I)

```

2548 GOTO 2520
2550 GOSUB 2400
2552 GOTO 2520
2554 IL=I-IR
2556 KA(IL)=J
2558 KU(IL)=K
2560 INPUT «Напряжение источника» ; QT(IL)
2562 RETURN

```

Ввод в ЭВМ управляемого источника тока оформляется в виде подпрограммы, которая используется и в программах ввода других элементов на БЕЙСИКе. Сначала осуществляется запрос, предполагается ли управление током или напряжением. Тип управляющих параметров фиксируется в массиве $MZ(I)$ следующим образом. При управлении напряжением номера ветвей запоминаются в массиве $MZ(I)$ как положительные числа, а при управлении по току программа присваивает номерам знак минус. Пользователь ничего этого не замечает, он вводит только букву I или U управляющего параметра и номер ветви управления.

В случае, если в ветви I нет ни одного управляемого источника, элемент массива управляющих ветвей $MZ(I)=0$ (строка 2518).

Управляемый источник тока обозначается переменной IG, которой перед вводом элементов схемы в строке 152 программы GLEICH присваивается значение, равное нулю. Если схема не содержит управляемых источников, то $IG=0$. При вводе в ЭВМ параметров источников тока устанавливается $IG=1$ (строка 2404).

Подпрограмма UPR 2400

```

а) необходимые переменные вызывающей программы:
  I — номер ветви;
б) используемые имена переменных:
  IS — номер управляющей ветви;
  JV — знак переменной;
в) результаты:
  IG — наличие управляемых источников;
  MZ(I) — номер управляющей ветви;
  SF(I) — коэффициент управления или усиления по току.
2400 REM: Ввод управляющих источников тока
2402 REM
2404 IG=1
2406 PRINT «Управляемый источник тока»
2408 INPUT «Управл параметр (I/U)»; FR$
2410 IF FR$ = «I» THEN 2416
2412 IF FR$ = «U» THEN 2422
2414 GOTO 2408
2416 JU=-1

```

```

2418 PRINT «Усиление по току B=»;
2420 GOTO 2426
2422 JV=1
2424 PRINT «Козффициент управления =»;
2426 INPUT « », SF(I)
2428 INPUT «Управляющая встень»; JS
2430 IF JS>0 THEN 2436
2432 PRINT
2434 GOTO 2428
2436 MZ(I)=JU * JS
RETURN

```

В главной программе после ввода элементов схемы определяется массив узлов. Затем подготавливается перемещение идеальных источников напряжения с помощью подпрограммы UPR 2600. Для этого собственные напряжения идеальных источников, которые должны быть размещены в ветви I, запоминаются в элементе VR(I) массива VR(IR).

Сначала каждому элементу VR(I) присваивается значение, равное нулю, затем выполняется запрос, является ли узел KV(I), в котором ветвь I начинается, или узел KZ(I), в котором она заканчивается, зависимым узлом KA(IL) идеального источника напряжения.

Необходимо обращать внимание на направление ветви. Если начальный узел KV(I) является зависимым, то собственное напряжение перемещаемого источника суммируется с VR(I). Если конечный узел KZ(I) является зависимым, то собственное напряжение источника вычитается из VR(I).

Г. дпрограмма UPR 2600

а) необходимые переменные вызывающей программы:

IR, IS, KA(I); KU(IR); KZ(IR); QT(IS)... (см. приложение ПЗ);

б) используемые имена переменных:

I — номер ветви;

IL — номер идеального источника напряжения;

JV — знак переменной;

в) результат:

VR(IR) — напряжение источников, соответствующих нагрузочным ветвям.

```

2600 REM: Перемещение идеальных источников напряжения
2602 REM
2604 FOR I=1 TO IR
2606 VR(I)=0
2608 IF IS=0 THEN 2626
2610 FOR IL=1 TO IS
2612 JV=1
2614 IF KU(I)=KA(IL) THEN 2622
2616 IF KZ(I)=KA(IL) THEN 2620
2618 GOTO 2624

```

```

2620 JV=-1
2622 VR(I)=VR(I)+QT(IL)
2624 NEXT IL
2626 NEXT I
2628 RETURN

```

2.5.3. Составление и решение системы уравнений. Матрица коэффициентов $A(N,N)$ системы линейных уравнений (2.4-1) является матрицей узловых проводимостей, а ее правая часть — матрицей токов источников. Перед составлением системы уравнений каждому элементу массивов $A(N,N)$ и $X(N)$ присваивается значение, равное нулю.

В строках 2716—2774 подпрограммы UPR 2700 заполняются эти массивы. Элементы нагрузочных ветвей обрабатываются следующим образом. Каждая ветвь I начинается в узле J , а кончается в узле K и характеризуется проводимостью $GD(I)$ и током источника $S1$, который вычисляется в строке 2718 через напряжение расположенного в этой ветви идеального источника напряжения.

Номера JR и KR строк и столбцов уравнений считаются из массива узлов (строки 2724 и 2726). В случае, если один из этих номеров равен нулю, обработка параметров ветви может быть опущена. Чтобы сократить число необходимых для этого запросов, заполняются строка 0 и столбец 0 массива A .

Для каждой ветви схемы проводимость $GD(I)$ суммируется с элементом главной диагонали и вычитается из элемента вспомогательной диагонали (строки 2730—2736). Кроме того, ток источника $S1$ согласно сформулированным в п. 2.1.2 правилам вычитается из элемента $X(IR)$ и складывается с $X(KR)$ (строки 2770 и 2772).

Если в ветви I нет ни одного управляемого источника, то строки 2740—2768 пропускаются. При обработке управляемого источника сначала вычисляются проводимость $G1$ и ток источника $S1$. В случае источника тока, управляемого напряжением, это происходит в строках 2744 и 2746, а в случае источника тока, управляемого током, — в строках 2750 и 2752.

В строке 2740 определяется номер управляющей ветви MS . Ее собственное напряжение может задаваться в виде разности узловых потенциалов, если один из двух узлов, которые она соединяет, является зависимым узлом идеального источника. Тогда напряжение $VR(MS)$,

размещенного в управляющей ветви источника напряжения, необходимо учитывать в правой части системы уравнений.

Для источника тока, управляемого напряжением, ток $S1$ и проводимость $G1$, равная коэффициенту управления $SF(I)$, суммируются с двумя элементами матрицы проводимостей (строки 2762—2768).

Номера строк JR и KR этих матричных элементов относятся к номерам узлов, в которых ветвь с управляемыми источниками начинается или заканчивается. В примере 2.3 $JR=2$ и $KR=3$.

Номера столбцов соответствуют узлам JS или KS , в которых управляющая ветвь начинается или заканчивается (строки 2754 и 2756). Поскольку в схеме могут существовать идеальные источники напряжения, номера JS и KS определяются с помощью массива узлов (строки 2758 и 2760). В примере 2.3 $JS=1$ и $KS=2$. В этом примере показано, по какой схеме проводимость $G1$ должна суммироваться с элементами матрицы узловых проводимостей или вычитаться из них.

Для источника тока, управляемого током, проводимость $G1$ определяется с помощью проводимости $GD(MS)$ управляющей ветви. В примере 2.4 ($JR=0$; $KR=2$; $JS=1$; $KS=2$) — это проводимость G_4 .

К току ветви $S1$ прибавляется ток управляющей ветви $SD(MS) + GD(MS) * VR(MS)$; умноженный на $SF(I)$.

В этом выражении уже учтены идеальные источники напряжения, которые размещаются в управляющей ветви.

После составления системы уравнений в подпрограмме UPR 2700 вызывается подпрограмма для ее решения. Если в цепи отсутствуют управляемые источники тока, то система уравнений решается подпрограммой UPR 2200. Для цепей с управляемыми источниками тока перед решением системы уравнений осуществляется ее упорядочение с помощью подпрограммы UPR 2300.

Подпрограмма UPR 2700 (она вызывает подпрограммы UPR 2000, 2200 и 2300)

а) необходимые переменные вызывающей программы (подробности см. в приложении ПЗ):

$GD(IR)$; IG ; IR ; $KF(KN)$; $KV(IR)$; $KZ(IR)$; $MZ(IR)$; $SD(IR)$; $SF(IR)$; $VR(IR)$;

б) используемые имена переменных и массивов:

$G1$ — проводимость;

I	— номер ветви;
J, K	— номера узлов;
JR, KR, JS, KS	— номера строк или столбцов;
MS	— номера управляющих ветвей;
S1	— ток источника,

в) результат:
 $X(N)$ — вектор решения.

```

2700 REM: Составление и решение системы линейных урав
2701 REM: нений
2702 REM
2704 FOR I=1 TO N
2706 X(I)=0
2708 FOR J=1 TO N
2710 A(I, J)=0
2712 NEXT J
2714 NEXT I
2716 FOR I=1 TO IR
2718 S1=SO(I)+GD(I) * VR(I)
2720 J=KV(I)
2722 K=KZ(I)
2724 JR=KF(J)
2726 KR=KF(K)
2728 IF JR=KR THEN 2774
2730 A(JR, JR)=A(JR, JR)+GD(I)
2732 A(JR, KR)=A(JR, KR)-GD(I)
2734 A(KR, JR)=A(KR, JR)-GD(I)
2736 A(KR, KR)=A(KR, KR)+GD(I)
2738 IF MZ(I)=0 THEN 2770
2740 MS=ABS(MZ(I))
2742 IF MZ(I)<0 THEN 2750
2744 G1=SF(I)
2746 S1=S1+SF(I) * VR(MS)
2748 GOTO 2754
2750 G1=SF(I) * GD(MS)
2752 S1=S1+G1 * VR(MS)+SF(I) * SD(MS)
2754 JS=KV(MS)
2756 KS=KZ(MS)
2758 JS=KF(JS)
2760 KS=KF(KS)
2762 A(JR, JS)=A(JR, JS)+G1
2764 A(JR, KS)=A(JR, KS)-G1
2766 A(KR, JS)=A(KR, JS)-G1
2768 A(KR, KS)=A(KR, KS)+G1
2770 X(JR)=X(JR)-S1
2772 X(KR)=X(KR)+S1
2774 NEXT I
2776 EP=1E-12
2778 IF IG=0 THEN 2786
2780 GOSUB 2300
2782 GOSUB 2000
2784 GOTO 2788
2786 GOSUB 2200
2788 RETURN

```

2.5.4. Расчет параметров ветви. Посредством решения

системы линейных уравнений сначала вычисляются узловые напряжения. Для этого используется подпрограмма UPR 2800.

Узловые напряжения независимых узлов являются решением системы уравнений. Номер узла, соответствующий номеру строки системы уравнений, считывается из массива узлов. Зависимым узлам сначала присваивается произвольное значение: таким образом можно очень просто сформулировать операторы цикла (строки 2804—2810). В строках 2816—2824 рассчитываются узловые напряжения зависимых узлов согласно уравнению (2.2-1).

Подпрограмма UPR 2800

а) необходимые переменные и массивы в вызывающей программе (подробности см. в приложении ПЗ):

IS, KA(IS); KF(KN); KU(IS); QT(IS); X(N);

б) используемые имена переменных:

IL — номер идеального источника напряжения;

J, K — номера узлов;

в) результат:

UN(KN) — узловые напряжения.

2800 REM: Расчет узловых напряжений

2802 REM

2804 FOR J=1 TO KN

2806 K=KF(J)

2808 UN(J)=X(K)

2810 NEXT

2812 UN(0)=0

2814 IF IS=0 THEN 2826

2816 FOR IL=1 TO IS

2818 J=KA(IL)

2820 K=KU(IL)

2822 UN(J)=UN(K)+QT(IL)

2824 NEXT

2826 RETURN

Расчет напряжений и токов для всех нагрузочных ветвей оформляется в виде подпрограммы UPR 2900. Каждое напряжение ветви является разностью двух узловых напряжений (строка 2910). Соответствующий ток ветви SA(I) вычисляется согласно уравнению (2.1-1).

Определить ток ветви (строки 2904—2914) можно только тогда, когда в ней отсутствует управляемый источник тока. В противном случае собственный ток источника SA(I) необходимо сложить с током ветви I.

Для источника тока, управляемого напряжением, ток вычисляется с помощью напряжения ветви (строка 2924); для источника, управляемого током, — с помощью тока ветви (строка 2928).

Подпрограмма UPR 2900

- а) необходимые переменные и массивы вызывающей программы (подробности см. в приложении ПЗ):
GD(IR); IR; KF(KN); KV(IR); KZ(IR); MZ(IR), SD(IR); UN(KN);
- б) используемые имена переменных:
I — номер ветви;
J, K — номера узлов;
MS — номер управляющей ветви;
S1 — собственный ток источника тока;
- в) результаты:
SA(IR) — токи нагрузочных ветвей;
UA(IR) — напряжения нагрузочных ветвей
- ```
2900 REM. Расчет параметров ветвей
2902 REM
2904 FOR I=1 TO IR
2906 J=KV(U)
2908 K=KZ(I)
2910 UA(I); UN(J)—UN(K)
2912 SA(I)=UA(I) * GD(I)+SD(I)
2914 NEXT
2916 FOR I=1 TO IR
2918 MS=ABS(MZ(I))
2920 IF MS=0 THEN 2932
2922 IF MZ(I)<0 THEN 2928
2924 S1=SF(I) * UA(MS)
2926 GOTO 2930
2928 S1=SF(I) * SA(MS)
2930 SA(I)=SA(I)+S1
2932 NEXT
2934 RETURN
```

После вычисления параметров в основной программе GLEICH осуществляется их вывод на периферийное устройство (строки 200—208). Несмотря на то что вычисления выполнялись с двойной точностью, параметры ветви выводятся на печать с обычной точностью (строки 202 и 204).

Результаты вычислений распечатываются в соответствующих размерностях, т. е. напряжения — в вольтах, а токи — в амперах.

**2.5.5. Изменения структуры цепи.** В основной программе после расчета и распечатки параметров ветви предусмотрена возможность исправления структуры цепи. Если же задается  $ME=0$ , то работа программы заканчивается. За счет изменения параметров элементов удается рассчитывать замещающие источники для двухполюсников без подробного ввода данных. Для этого цепь анализируется дважды: сначала для нагрузки с проводимостью  $G_{va}$ , затем для нагрузки с проводимос-



тью  $G_{vb}$ . В первом случае через нагрузку течет ток ветви  $I_{va}$ . При напряжении ветви  $U_{va}$  уравнение узлов выглядит следующим образом:

$$I_{qe} + G_{ie} U_{va} + I_{va} = 0. \quad (2.5-1)$$

Во втором случае через нагрузку течет ток ветви  $I_{vb}$  и уравнение узлов записывается в виде

$$I_{qe} + G_{ie} U_{vb} + I_{vb} = 0, \quad (2.5-2)$$

На рис. 2.22 показаны расчетные схемы активной двухполюсной цепи.

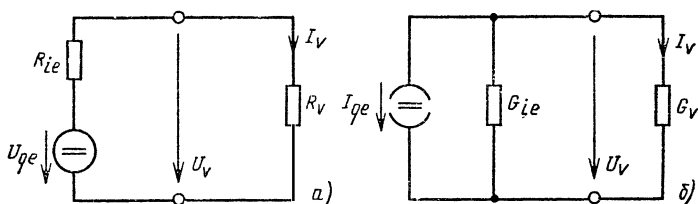


Рис. 2.22. К расчету замещающих источников напряжения (а) или тока (б) активной двухполюсной цепи

С учетом уравнений (2.5-1) и (2.5-2) проводимость

$$G_{ie} = - \frac{I_{va} - I_{vb}}{U_{va} - U_{vb}}, \quad (2.5-3)$$

а ток замещающего источника тока

$$I_{qe} = - G_{ie} U_{va} - I_{va}. \quad (2.5-4)$$

Значения  $U_{qe}$  и  $R_{ie}$  замещающего источника напряжения можно рассчитать с помощью уравнения (2.1-2) и (2.1-3).

**2.5.6. Основная программа.** Сегменты основной программы представляются в виде определенной последовательности операторов с соответствующими комментариями. Последней (не печатаемой) строкой основной программы является

9999 END.

При запуске программы GLEICH в оперативной памяти должны находиться заданные в строках 1000 и 1002 подпрограммы.

100 PRINT GLEICH

102 PRINT «Узловой анализ линейной цепи»

```

104 PRINT «постоянного тока»
106 PRINT «с управляемыми источниками тока»
108 PRINT
110 DEFINT I—K, M—N
112 DEFDBL A—D, G, L, P—Z
114 REM
116 REM: Ввод глобальных параметров
118 INPUT «Количество ветвей»; IZ
120 INPUT «Количество узлов (без базового)»; KN
122 INPUT «Количество идеальных источн напряж»; IS
124 PRINT
126 N=KN-IS
128 IF IZ<IS THEN 132
130 IF N>0 THEN 138
132 PRINT «Ввод не имеет смысла»
134 PRINT «или обработка невозможна»
136 GOTO 118
138 IR=IZ-IS
140 DIM UN(KN), KF(KN), A(N, N), X(N), KA(IS), KU(IS),
 QT(IS)
142 DIM SF(IR), MZ(IR), VR(IR)
144 DIM KV(IR), KZ(R), GD(IR), SD(IR), SA(IR), UA(IR)
146 REM
148 PRINT «Ввод структуры цепи»
150 REM
152 IG=0
154 FOR I=1 TO IZ
156 PRINT «Ветвь»; I
158 GOSUB 2500
160 NEXT I
162 REM
164 REM: Определение массива узлов
166 GOSUB 2100
168 REM
170 REM: Подготовка перемещения
172 REM: идеальных источников напряжения
174 GOSUB 2600
176 REM
178 REM: Составление и решение
180 REM: системы линейных уравнений
182 GOSUB 2700
184 REM
186 REM: Расчет узловых напряжений
188 GOSUB 2800
190 REM
192 REM: Вычисление и вывод параметров ветвей
194 REM
196 GOSUB 2900
198 PRINT
200 FOR I=1 TO IR
202 EU=UA(I)
204 EI=SA(I)
206 PRINT «U»; I; «=»; EU; «V»; I; I; «=»; EI; «A»
208 NEXT
210 REM

```

```

212 REM: Изменение параметров ветвей
214 REM
216 PRINT
218 PRINT «В скольких ветвях должны быть»
220 INPUT «изменены элементы»; ME
222 IF ME<1 THEN 9999
224 FOR ML=1 TO ME
226 INPUT «Ветвь»; I
228 GOSUB 2500
230 NEXT
232 GOTO 166
1000 REM: UPR 2000, 2100, 2200, 2300, 2400
1002 REM: UPR 2500, 2600, 2700, 2800, 2900

```

**2.5.7. Пример.** Рассчитаем схему, приведенную на рис. 2.20, а. Заменяем транзистор ВС 107 источником тока, управляемым током (рис. 2.19), с коэффициентом усиления  $B=280$ . Двухполюсная схема между клеммами 1 и 2 (рис. 2.23) нагружается сначала проводимос-

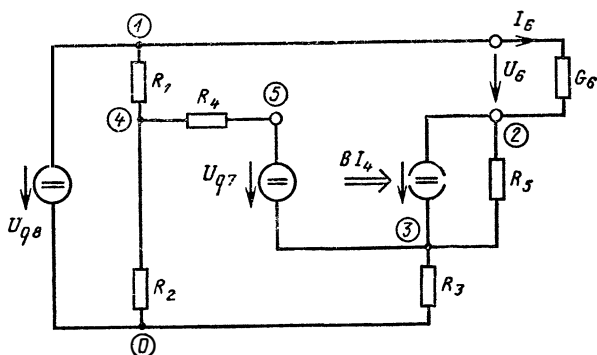


Рис. 2.23. Стабилизированный источник тока (см. рис. 2.20, а) с линейной схемой замещения транзистора ВС 107, изображенного на рис. 2.19

тью  $G_{6a}=1$  мСм, а затем проводимостью  $G_{6b}=2$  мСм.

Расчет электрической схемы, приведенной на рис. 2.10, по программе GLEICH

Узловой анализ линейной  
цепи постоянного тока с  
управляемыми источниками  
тока

Количество ветвей ? 8

Количество узлов без базового ? 5

Количество идеальных источников напряжения ? 2

Ввод структуры цепи

Ветвь 1

из узла ? 1

к узлу ? 4

Элемент ? R  
 Сопротивление ? 68000  
 Элемент ? N  
 Ветвь 2  
   из узла ? 4  
   к узлу ? 0  
 Элемент ? R  
 Сопротивление ? 68000  
 Элемент ? N  
 Ветвь ? 3  
   из узла ? 3  
   к узлу ? 0  
 Элемент ? R  
 Сопротивление ? 1750  
 Элемент ? N  
 Ветвь ? 4  
   из узла ? 4  
   к узлу ? 5  
 Элемент ? R  
 Сопротивление ? 1600  
 Элемент ? N  
 Ветвь 5  
   из узла ? 2  
   к узлу ? 3  
 Элемент ? R  
 Сопротивление ? 42500  
 Элемент ? S  
 Управляемый источник тока  
 Управляющий параметр (1/U)? I  
 Усиление по току B= ? 280  
 Управляющая ветвь ? 4  
 Элемент ? N  
 Ветвь 6  
   из узла ? 1  
   к узлу ? 2  
 Элемент ? G  
 Проводимость ? . 001  
 Элемент ? N  
 Ветвь 7  
   из узла ? 5  
   к узлу ? 3  
 Напряжение источника ? .62  
 Ветвь 8  
   из узла ? 1  
   к узлу ? 0  
 Напряжение источника ? 20  
 Результаты расчета (машинная распечатка):

|                     |                     |
|---------------------|---------------------|
| U 1=10.58818 В;     | I 1=1.557085E-04 А  |
| U 2=9.411819 В;     | I 2=1.384091E-04 А  |
| U 3=8.764141 В;     | I 3=5.0080081E-03 А |
| U 4=2.767908E-02 В; | I 4=1.729942E-05 А  |
| U 5=6.245079 В      | I 5=4.990781E-03 А  |
| U 6=4.990781 В;     | I 6=4.990781E-03 А  |

В скольких ветвях должны быть  
изменены элементы ? 1

Ветвь ? 6

из ула ? 1

к узлу ? 2

Элемент ? C

Проводимость ? .002

Элемент ? N

Результаты расчета:

U 1=10,58158 В;

U 2=9,418421 В;

U 3=8,771052 В;

U 4=2,736845E-2 В;

U 5=8,731486 В;

U 6=2,497462 В;

I 1=1.556115E-04 А

I 2=1.385062E-04 А

I 3=5.01203E-03 А

I 4=1.710528E-05 А

I 5=4.994925E-03 А

I 6=4.994925E-03 А

В скольких ветвях должны быть  
изменены элементы ? 0

Ок

Подставим результирующие значения  $U_{6a}=4,99078$  В;  
 $I_{6a}=4,99078$  мА и  $U_{6b}=2,49746$  В;  $I_{6b}=4,99493$  мА  
в уравнения (2.5-3) и (2.5-4) и вычислим параметры за-  
мещающих источников тока:  $I_e = -4,999$  мА;  $G_e =$   
 $=1,66$  мкСм.

## Глава 3

# ЛИНЕЙНЫЕ ЦЕПИ СИНУСОИДАЛЬНОГО НАПРЯЖЕНИЯ

## 3.1. Комплексные величины

### 3.1.1. Синусоидальное напряжение. Выражение

$$u = \hat{u} \cos(\omega t + \varphi_u) \quad (3.1-1)$$

полностью описывается с помощью трех величин:

1) амплитуды  $\hat{u}$ ; вместо нее может быть задано эф-  
фективное значение

$$U = \hat{u} / \sqrt{2}; \quad (3.1-2)$$

2) круговой частоты  $\omega$ , определяемой через частоту  
 $f$  следующим образом:

$$\omega = 2\pi f; \quad (3.1-3)$$

3) начального фазового сдвига

$$-180^\circ \leq \varphi_u \leq 180^\circ.$$

Если линейная схема содержит источники синусоидального напряжения с одинаковой частотой, то все токи и напряжения в установившемся режиме изменяются по синусоидальному закону с той же частотой. Такая цепь может быть описана с помощью символического метода. Для этого из эффективного значения и начального фазового сдвига образуют комплексную величину [12].

Для синусоидального напряжения с эффективным значением  $\underline{U}$  и начальным фазовым сдвигом  $\varphi_u$  этой комплексной величиной является напряжение

$$\underline{U} = U e^{j\varphi_u} = U \angle \varphi_u. \quad (3.1-4)$$

**3.1.2. Комплексная проводимость.** По аналогии с синусоидальным напряжением в комплексной форме можно записать и синусоидальный ток:

$$i = \hat{i} \cos(\omega t + \varphi_i); \quad (3.1-5)$$

$$\underline{I} = I e^{j\varphi_i} = I \angle \varphi_i. \quad (3.1-6)$$

При этом  $I = \hat{i} / \sqrt{2}$  является эффективным (действующим) значением синусоидального тока. Взаимосвязь между синусоидальным значением  $\underline{U}$  и током  $\underline{I}$  линейного двухполюсника описывается с помощью комплексного сопротивления

$$\underline{Z} = \frac{\underline{U}}{\underline{I}}. \quad (3.1-7)$$

В методе узловых потенциалов используется комплексная проводимость, обратная комплексному сопротивлению;

$$\underline{Y} = \frac{1}{\underline{Z}} = \frac{\underline{I}}{\underline{U}}. \quad (3.1-8)$$

**3.1.3. Базовые двухполюсники.** Линейная цепь синусоидального напряжения может содержать базовые двухполюсники: активные сопротивления  $R = \text{const}$ , индуктивность  $L = \text{const}$  и емкость  $C = \text{const}$ .

Комплексная проводимость активного сопротивления является вещественной величиной, т. е.

$$\underline{Y}_R = \frac{1}{R}. \quad (3.1-9)$$

Комплексная проводимость емкости представляет собой

мнимую величину:

$$\underline{Y}_C = j\omega C. \quad (3.1-10)$$

Для определения комплексной проводимости индуктивности

$$\underline{Y}_L = \frac{1}{j\omega L} \quad (3.1-11)$$

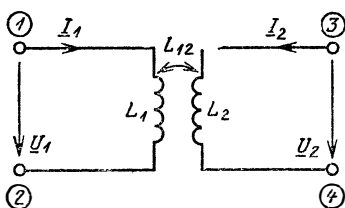
умножим числитель и знаменатель на  $j$  и, учитывая, что  $j^2 = -1$ , получим

$$\underline{Y}_L = -j \frac{1}{\omega L}. \quad (3.1-12)$$

Проводимость параллельно включенных базовых двухполюсников  $R$ ,  $L$ ,  $C$  равна сумме проводимостей:

$$\underline{Y} = \underline{Y}_R + \underline{Y}_L + \underline{Y}_C. \quad (3.1-13)$$

**3.1.4. Взаимная индуктивность.** Магнитная связь между двумя проводниками с индуктивностями  $L_1$  и  $L_2$  описывается взаимной индуктивностью  $L_{12} = L_{21}$  (рис. 3.1). В некоторых случаях ее можно учесть и с помощью коэффициента связи



$$k = \frac{L_{12}}{\sqrt{L_1 L_2}}. \quad (3.1-14)$$

Для трансформатора без потерь справедливы следующие соотношения [12]:

$$\left. \begin{aligned} \underline{U}_1 &= j\omega L_1 \underline{I}_1 + j\omega L_{12} \underline{I}_2; \\ \underline{U}_2 &= j\omega L_{12} \underline{I}_1 + j\omega L_2 \underline{I}_2. \end{aligned} \right\}$$

Рис. 3.1. Магнитная связь между двумя проводниками:

1—4 — зажимы

$$(3.1-15)$$

Заменим четырехполюсник, описанный с помощью уравнений (3.1-15), линейными двухполюсниками (см. п. 2.3.1). Это возможно лишь в случае соединения его входных и выходных зажимов (или клемм). Для схемы, изображенной на рис. 3.2, могут быть заданы элементы Т-образной схемы замещения, если подставить параметры  $\underline{Z}_{11} = j\omega L_1$ ;  $\underline{Z}_{12} = \underline{Z}_{21} = j\omega L_{12}$ ;  $\underline{Z}_{22} = j\omega L_2$  уравнения четырехполюсника (3.1-15) в соответствующие уравнениям (2.3-6) комплексные уравнения. Вместо комплексных проводимостей на рис. 3.3 заданы индуктивности.

Рассчитаем теперь идеальный трансформатор с коэффициентом связи  $k=1$ . Если его входная и выходная клеммы соединены друг с другом, то  $L_1 - L_{12} = 0$  и  $L_2 - L_{12} = 0$ , т.е. трансформатор можно заменить схемой, представленной на рис. 3.4.

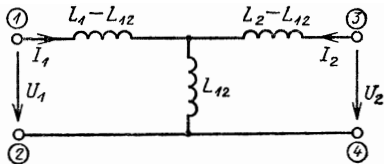
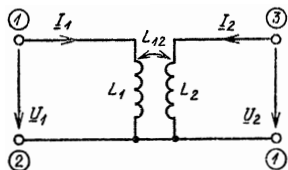


Рис. 3.2. Магнитная связь между двумя проводниками с электрическим соединением зажимов 2 и 4

Рис. 3.3. Т-образная схема замещения двух магнитно-связанных проводников согласно рис. 3.2:

1—4 — зажимы

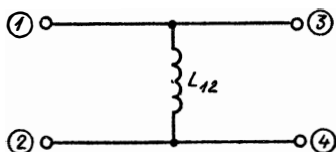


Рис. 3.4. Схема замещения идеального трансформатора ( $\ddot{u}=1$ ) с электрически связанными зажимами 1—4

Трансформатор, у которого электрической связи между клеммами входа и выхода (см. рис. 3.2) не существует, удастся рассчитать только в том случае, если не учитывать потери в его обмотках. Разделение клемм может быть описано с помощью идеального трансформатора (рис. 3.5, а) с передаточным отношением  $\ddot{u} = N_1/N_2$ , причем уравнения четырехполюсника заданы в форме проводимостей

$$\left. \begin{aligned} \underline{U}_1 &= \ddot{u} \underline{U}_2; \\ \underline{I}_1 &= -\frac{1}{\ddot{u}} \underline{I}_2. \end{aligned} \right\} \quad (3.1-16)$$

Схема замещения идеального трансформатора состоит из двух управляемых источников (рис. 3.5, б).

Управляемые источники могут быть преобразованы с помощью последовательно включенных сопротивлений в управляемый источник тока (см. п. 2.3.2). Согласно



описанному методу схемного анализа рассчитываются только линейные управляемые источники тока, коэффициент управления которых является вещественным и постоянным. Однако это справедливо только в том случае, если последовательно включенные сопротивления явля-

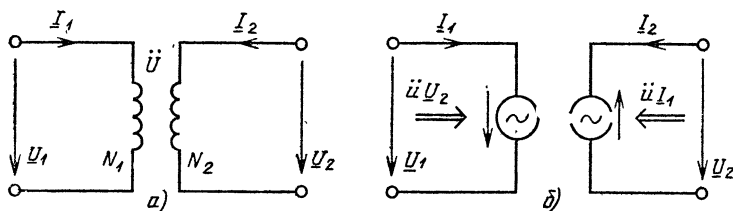


Рис. 3.5. Собственно схема (а) и схема замещения (б) идеального трансформатора

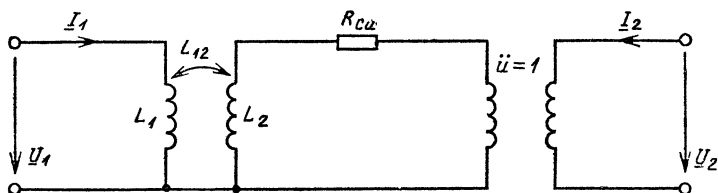


Рис. 3.6. Трансформатор с разделенными зажимами, имеющий потери

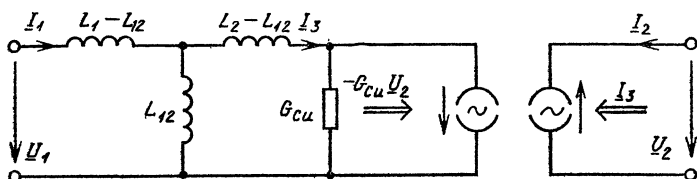


Рис. 3.7. Схема замещения трансформатора, приведенного на рис. 3.6

ются активными, иначе коэффициент управления будет комплексным и зависящим от частоты.

Если потери в обмотках трансформатора учитываются с помощью активного сопротивления, включенного последовательно с обмоткой (рис. 3.6), то схема замещения трансформатора для этого случая приведена на рис. 3.7.

## 3.2. Комплексные вычисления

**3.2.1. Преобразования координат.** Комплексная величина может быть записана в полярных или декартовых координатах. В первом случае (сокращенно *P*-форма) присутствует модуль и угол комплексной величины. Для комплексного напряжения  $U$  модулем является эффективное значение  $U$ , а углом — начальный фазовый сдвиг  $\varphi$  [см. уравнение (3.1-4)]. Угол называют также аргументом. Во втором случае (*R*-форма) существуют реальная и мнимая части комплексной величины. При проведении расчетов целесообразно воспользоваться *R*-формой, а результаты комплексных вычислений обычно выводятся из ЭВМ в *P*-форме.

В дальнейшем будем иметь дело с коэффициентными преобразованиями, причем используем общие обозначения  $r_p$  и  $\varphi$  для модуля и угла комплексной величины, а также  $a_r$  и  $b_i$  для вещественной и мнимой частей (рис. 3.8):

$$r_p \angle \varphi = r_p e^{j\varphi} = a_r + jb_i. \quad (3.2-1)$$

Для преобразования координат из *P*-формы в *R*-форму задаются модуль  $r_p$ , угол  $\varphi$  и определяются реальная и мнимая части комплексной величины. С учетом формулы Эйлера

$$r_p e^{j\varphi} = r_p (\cos \varphi + j \sin \varphi). \quad (3.2-2)$$

Поскольку левые части уравнений (3.2-1) и (3.2-2) равны между собой, запишем

$$a_r = r_p \cos \varphi; \quad (3.2-3)$$

$$b_i = r_p \sin \varphi. \quad (3.2-4)$$

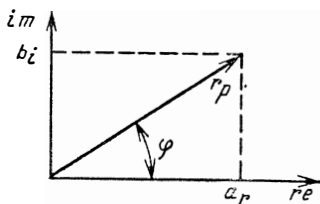


Рис. 3.8. Преобразование координат комплексного числа

Существует специальная подпрограмма UPR 3100 на БЕЙСИКе для преобразования координат.

В строке 3104 по закону гипотенуз Пифагора вычисляется модуль

$$r_p = \sqrt{a_r^2 + b_i^2}. \quad (3.2-5)$$

Угол  $-180^\circ \leq \varphi \leq 180^\circ$  измеряется в градусах. С помощью функции ATN можно определить только угол

$$\varphi_1 = \frac{180}{\pi} \operatorname{Arctan} \frac{b_i}{a_r}, \quad (3.2-6)$$

причем  $-90^\circ \leq \varphi_1 \leq 90^\circ$ . Поэтому перед вычислением угла  $\varphi$  потребуются несколько дополнительных рассуждений.

При делении  $b_i/a_r$  в программе необходимо исключить случай, когда  $a_r=0$ , потому что иначе операционная система сообщит о переполнении разрядной сетки. Для  $a_r=0$  угол  $\varphi=\varphi_2$  определяется непосредственно с помощью сигнум-функции

$$\varphi_2 = 90 \operatorname{sgn}(b_i). \quad (3.2-7)$$

Для  $a_r=0$  выполняется возврат в главную программу (строки 3104 и 3114). Угол  $\varphi$  совпадает с углом  $\varphi_1$  только для  $a_r>0$ . В этом случае оба угла меняются в диапазоне  $-90^\circ \leq \varphi \leq +90^\circ$ . Для  $a_r<0$  угол  $\varphi$  вычисляется с учетом корректирующего угла  $\varphi_3$ , т. е.

$$\varphi = \varphi_1 + \varphi_3. \quad (3.2-8)$$

Угол  $\varphi_3$  при  $b_i<0$  равняется  $-180^\circ$ , а при  $b_i \geq 0$  он равен  $180^\circ$  (различные значения  $\varphi_3$  см. ниже).

В подпрограмме UPR 3100 выполняется корректировка угла  $\varphi_1$  для  $a_r>0$  или  $a_r<0$  с помощью вспомогательной величины

$$S_1 = 1 - \operatorname{sgn}(a_r) \quad (3.2-9)$$

|                                                    | $a_r < 0$<br>$S_1 = 2$   | $a_r > 0$<br>$S_1 = 0$ |
|----------------------------------------------------|--------------------------|------------------------|
| $b_i > 0$<br>$\varphi_2 = 90^\circ$<br>$S_2 = 1$   | $\varphi_3 = 180^\circ$  | $\varphi_3 = 0$        |
| $b_i = 0$<br>$\varphi_2 = 0$<br>$S_2 = 1$          | $\varphi_3 = 180^\circ$  | $\varphi_3 = 0$        |
| $b_i < 0$<br>$\varphi_2 = -90^\circ$<br>$S_2 = -1$ | $\varphi_3 = -180^\circ$ | $\varphi_3 = 0$        |

Если  $a_r>0$ , то  $S_1=0$ , а если  $a_r<0$ , то  $S_1=2$ . Поскольку корректирующий угол  $\varphi_3$  содержит в качест-

ве коэффициента вспомогательную величину  $S_1$ , изменений угла  $\varphi_1$  определяются согласно уравнению (3.2-8) только для случая, когда вещественная часть  $a_r$  отрицательна.

Зависимость корректирующего угла  $\varphi_3$  от знака мнимой части  $b_i$  описывается с помощью коэффициента

$$S_2 = \operatorname{sgn}(\varphi_2 + 1). \quad (3.2-10)$$

Если  $b_i \geq 0$ , то  $S_2 = 1$ , а если  $b_i < 0$ , то  $S_2 = -1$ .

Таким образом, корректирующий угол  $\varphi_3$  вычисляется следующим образом (строка 3110):

$$\varphi_3 = 90 S_1 S_2. \quad (3.2-11)$$

В подпрограмме UPR 3100 углам  $\varphi$ ,  $\varphi_1$ ,  $\varphi_2$  и  $\varphi_3$  присваивается имя PH.

#### Подпрограмма UPR 3100

а) необходимые переменные в вызывающей программе:

AR — действительная часть комплексного числа;

BI — мнимая часть комплексного числа;

PI — число;

б) используемые имена переменных: нет;

в) результат:

PH — угол комплексного числа (в угловых градусах);

RP — модуль комплексного числа.

3100 REM : Преобразование координат R→P

3102 REM

3104 RP=SQR(AR \* AR+BI \* BI)

3106 PH=90 \* SGN(BI)

3108 IF AR=Q THEN 3114

3110 PH=90 \* (1—SGN(AR)) \* SGN(PH+1)

3112 PH=180 \* ATN(BI/AR)/PI+PH

3114 RETURN

**3.2.2. Основные типы вычислений.** С комплексными числами можно выполнять те же операции, что и с вещественными. В дальнейшем под основными типами вычислений будем понимать сложение, вычитание, умножение и деление. Для ПЭВМ основные типы вычислений целесообразно программировать в  $R$ -форме, поскольку преобразование координат требует слишком много вычислительного времени. Опишем процесс вычислений на примере комплексных чисел  $\underline{a} = a_r + ja_i$  и  $\underline{b} = b_r + jb_i$ . В результате получим  $\underline{c} = c_r + jc_i$ .

Сложение:  $\underline{c} = \underline{a} + \underline{b}$ . Подставим комплексные числа в  $R$ -форме:

$$c_r + jc_i = a_r + ja_i + b_r + jb_i. \quad (3.2-12)$$

Теперь разложим уравнение на вещественную  $c_r$  и мнимую  $c_i$  части:

$$c_r = a_r + b_r; \quad (3.2-13)$$

$$c_i = a_i + b_i. \quad (3.2-14)$$

Вычитание:  $\underline{c} = \underline{a} - \underline{b}$ . Так же, как и при сложении, выделим из уравнения

$$c_r + jc_i = a_r + ja_i - (b_r + jb_i) \quad (3.2-15)$$

соответственно вещественную и мнимую части:

$$c_r = a_r - b_r; \quad (3.2-16)$$

$$c_i = a_i - b_i. \quad (3.2-17)$$

Умножение:  $\underline{C} = \underline{a}\underline{b}$ . Подставим в выражение комплексные числа в  $R$ -форме

$$c_r + jc_i = (a_r + ja_i)(b_r + jb_i) \quad (3.2-18)$$

и перемножим их. Поскольку  $j^2 = -1$ , имеем

$$c_r + jc_i = a_r b_r + ja_r b_i + ja_i b_r - a_i b_i. \quad (3.2-19)$$

Это уравнение разложим вновь на вещественную и мнимую части:

$$c_r = a_r b_r - a_i b_i; \quad (3.2-20)$$

$$c_i = a_i b_r + a_r b_i. \quad (3.2-21)$$

Деление:  $\underline{c} = \underline{a}/\underline{b}$ . Воспользуемся сопряженным комплексным знаменателем  $b^* = (b_r - jb_i)$  и запишем

$$c = \frac{a b^*}{b b^*} = \frac{(a_r + ja_i)(b_r - jb_i)}{(b_r + jb_i)(b_r - jb_i)}. \quad (3.2-22)$$

Знаменатель выражения (3.2-22) теперь представляет собой действительную часть, а числитель — мнимую:

$$c_r + jc_i = \frac{a_r b_r + a_i b_i + j(a_i b_r - a_r b_i)}{b_r^2 + b_i^2}. \quad (3.2-23)$$

Выделим отдельно вещественную и мнимую части:

$$c_r = (a_r b_r + a_i b_i)/(b_r^2 + b_i^2); \quad (3.2-24)$$

$$c_i = (a_i b_r - a_r b_i)/(b_r^2 + b_i^2). \quad (3.2-25)$$

Недостаток языка БЕЙСИК заключается в том, что при выполнении подпрограммы невозможно согласовать передачу параметров: в одной программе могут много-

кратно использоваться только глобальные переменные. Если бы выполнение одной базовой вычислительной операции было оформлено в виде подпрограммы, то потребовалось бы перед ее вызовом предусмотреть переименование переменных. Поэтому проще описывать комплексные вычисления непосредственно с помощью текущих переменных в качестве фрагментов программы.

В программах же на языке ПАСКАЛЬ при вызове подпрограммы передача параметров возможна. Однако для комплексных вычислений, так же как и в программах на БЕЙСИКе, от этого приходится отказаться, чтобы сэкономить время вычислений.

**3.2.3. Решение системы линейных уравнений.** Комплексную систему уравнений в принципе можно решить в действительных числах, поскольку выражение

$$(A_r + jA_i)(x_r + jx_i) = b_r + jb_i \quad (3.2-26)$$

перемножается следующим образом:

$$A_r X_r - A_i x_i = b_r,$$

$$A_i X_r + A_r x_i = b_i.$$

Полученная система вещественных уравнений с помощью частичных матриц  $A_r$ ;  $A_i$ ;  $x_r$ ,  $x_i$ ,  $b_i$  может быть представлена в виде

$$\begin{bmatrix} A_r & -A_i \\ A_i & A_r \end{bmatrix} \cdot \begin{bmatrix} x_r \\ x_i \end{bmatrix} = \begin{bmatrix} b_r \\ b_i \end{bmatrix}. \quad (3.2-27)$$

При схемном анализе этот метод [22] не гарантирует от ошибок по двум причинам. Во-первых, для хранения в памяти ЭВМ вещественной матрицы коэффициентов необходим в 2 раза больший объем оперативной памяти по сравнению с тем, который требуется для хранения вещественной и мнимой частей комплексной матрицы коэффициентов. Кроме того, для решения системы (3.2-27) нужно больше времени, чем для решения (3.2-26).

Во-вторых, если двухполюсник не содержит ни одного сопротивления, то реальная часть комплексной проводимости равна нулю. В этом случае один элемент на главной диагонали матрицы коэффициентов в (3.2-27) равен нулю. Поэтому упорядочение матрицы необходимо также и для цепи, не содержащей ни одного управляемого источника.

Однако вследствие замены строк симметричная матрица коэффициентов теряет свою симметрию и система уравнений решается с большими затратами машинного времени (см. п. 2.4.3).

Таким образом, целесообразно решать систему линейных уравнений (3.2-26) в комплексных числах и для этого модифицировать подпрограммы UPR 2000, 2200 и 2300.

С помощью подпрограммы UPR 3000 может быть решена комплексная система уравнений цепи с управляемым источником тока. При этом  $IG=1$  (см. п. 2.5.2). Комплексную матрицу коэффициентов  $(A_r + jA_i)$  следует перед вызовом программы UPR 3000 разместить в массивах  $A(N, N)$  и  $B(N, N)$ . Правая часть  $(b_r + jb_i)$  выражения (3.2-27) предварительно размещается в массивах  $X(N)$  и  $Y(N)$ . Решение  $x_r + jx_i$  после вызова UPR 3000 содержится в массивах  $X(N)$  и  $Y(N)$ ; таким образом, «старая» информация стирается из памяти ЭВМ. При выполнении программы UPR 3000 матрица коэффициентов также стирается.

Величина  $D1$  или  $R1$  из подпрограммы UPR 2000 становится в UPR 3000 величиной комплексной ( $D1 + jD2$  или  $R1 + jR3$ ). Кроме того, согласно уравнению (3.2-23) необходимо вычислить знаменатель  $D3$  или  $R3$ .

При выяснении, является ли значение элемента на главной диагонали матрицы коэффициентов меньшим, чем  $\epsilon$ , корень квадратный не извлекается. Вместо этого сравниваются квадраты значений коэффициента и  $\epsilon$ . Выбирается такое значение предела  $\epsilon$ , чтобы  $\epsilon^2$  превышало наименьшее значение диагонального элемента (число с плавающей запятой) для используемой ПЭВМ (см. гл. 1).

#### Подпрограмма UPR 3000

- а) необходимые переменные вызывающей программы:
  - $A(N, N)$  — действительная часть  $A$ , матрицы коэффициентов;
  - $B(N, N)$  — мнимая часть  $A_i$  матрицы коэффициентов;
  - EP — предельное значение  $\epsilon$ ;
  - IG — наличие управляемых источников тока;
  - N — количество уравнений;
  - $X(N)$  — действительная составляющая правой части  $b_r$ ;
  - $Y(N)$  — мнимая составляющая правой части  $b_i$ ;
- б) используемые имена переменных:
  - E2 — квадрат предельного значения  $\epsilon$ ;
  - $D1-D3$  — вспомогательные величины,
  - I, J, K — переменные циклов;
  - $R1-R3$  — вспомогательные величины;

в) результат:

$X(N)$  — действительная часть вектора решения;  
 $Y(N)$  — мнимая часть вектора решения.

```
3000 REM: Решение системы линейных уравнений
3002 REM
3004 E2=EP * EP
3006 IF N=1 THEN 3042
3008 FOR I=1 TO N-1
3010 D1=A(I, I)
3012 D2=B(I, I)
3014 D3=D1 * D1+D2 * D2
3016 IF D3<E2 THEN 3074
3018 FOR J=1+1 TO N
3020 R1=(A(J, I) * D1+B(J, I) * D2)/D3
3022 R2=(B(J, I) * D1-A(J, I) * D2)/D3
3024 IF (ABS(R1)+ABS(R2))=0 THEN 3038
3026 FOR K=1+1 TO N
3028 A(J, K)=A(J, K)-R1 * A(I, K)+R2 * B(I, K)
3030 B(J, K)=B(J, K)-R1 * B(I, K)-R2 * A(I, K)
3032 NEXT K
3034 X(J)=X(J)-R1 * X(I)+R2 * Y(I)
3036 Y(J)=Y(J)-R1 * Y(I)-R2 * X(I)
3038 NEXT J
3040 NEXT I
3042 FOR J=N TO 1 STEP -1
3044 D1=X(J)
3046 D2=Y(J)
3048 IF J=N THEN 3058
3050 FOR K=N TO J+1 STEP -1
3052 D1=D1-A(J, K) * X(K)+B(J, K) * Y(K)
3054 D2=D2-B(J, K) * X(K)-A(J, K) * Y(K)
3056 NEXT K
3058 R1=A(J, J)
3060 R2=B(J, J)
3062 R3=R1 * R1+R2 * R2
3064 IF R3<E2 THEN 3074
3066 X(J)=(D1 * R1+D2 * R2)/R3
3068 Y(J)=(D2 * R1-D1 * R2)/R3
3070 NEXT J
3072 RETURN
3774 PRINT «Решение невозможно»
3076 GOTO 9999
3078 RETURN
```

В том случае, когда цепь содержит управляемый источник ( $IG=1$ ), необходимо удостовериться, что все элементы на главной диагонали матрицы коэффициентов отличны от нуля. Это выполняется с помощью подпрограммы UPR 3300, которая является версией подпрограммы UPR 2300. Если замена строк матрицы невозможна, то выдается сообщение «Решение невозможно».



## Подпрограмма UPR 3300

а) необходимые переменные вызывающей программы:

A(N, N) — матрица коэффициентов;  
 EP — предельное значение  $\varepsilon$ ;  
 N — количество уравнений;  
 X(N) — правая часть;

б) используемые имена переменных:

A1, B1, B2 — вспомогательные величины;  
 I, J — переменные циклов;  
 J1 — номер строки;  
 R1, X1, Y1 — вспомогательные величины;

в) результат:

A(N, N) — матрица коэффициентов;  
 X(N) — правая часть.

```

3300 REM: Упорядочение системы уравнений
3302 REM
3304 FOR I=1 TO N
3306 B1=SQR(A(I, I) ^ 2 + B(I, I) ^ 2)
3308 IF B1 > EP THEN B360
3310 J1=0
3312 R1=0
3314 FOR J=1 TO N
3316 B1=SQR(A(I, J) ^ 2 + B(I, J) ^ 2)
3318 B2=SQR(A(J, I) ^ 2 + B(J, I) ^ 2)
3320 A1=B1 * B2
3322 IF A1 < R1 THEN B328
3324 R1=A1
3326 J1=J
3328 NEXT J
3330 IF R1 < EP THEN B074
3332 X1=X(J1)
3334 Y1=Y(J1)
3336 X(J1)=X(I)
3338 Y(J1)=Y(I)
3340 X(I)=X1
3342 Y(I)=Y1
3344 FOR J=1 TO N
3346 A1=A(J, I, J)
3348 B1=B(J1, J)
3350 A(J1, J)=A(I, J)
3352 B(J1, J)=B(I, J)
3354 A(I, J)=A1
3356 B(I, J)=B1
3358 NEXT J
3360 NEXT I
3362 RETURN

```

Если цепь не содержит управляемых источников, матрица коэффициентов системы комплексных уравнений симметрична.

Эта симметрия используется при редукции системы уравнений для уменьшения времени вычислений (см. п. 2.4.4). При  $j > k$  коэффициенты  $a_{jk}$  не вычисляются и не

запоминаются. С помощью подпрограммы UPR 3200, которая является комплексной версией UPR 2200, осуществляется только редукция матрицы коэффициентов и матрицы правой части системы. Поскольку обратная подстановка выполняется так же, как и в подпрограмме UPR 3000, последняя вызывается со строки 3042. Поэтому UPR 3000 должна всегда присутствовать в оперативной памяти ЭВМ одновременно с UPR 3200.

#### Подпрограмма UPR 3200 (она вызывает UPR 3000)

```

а) —в) см. UPR 3000
3200 REM: Решение системы линейных уравнений
3202 REM: с симметричной матрицей коэффициентов
3204 REM
3206 E2=EP* EP
3208 IF N=1 THEN 3244
3210 FOR I=1 TO N-1
3212 D1=A(I, I)
3214 D2=B(I, I)
3216 D3=D1 * D1+D2 * D2
3218 IF D3<E2 THEN 3074
3220 FOR J=I+1 TO N
3222 R1=(A(I, J) * D1+B(I, J) * D2)/D3
3224 R2=(B(I, J) * D1-A(I, J) * D2)/D3
3226 IF (ABS(R1)+ABS(R2))=0 THEN 3240
3228 FOR K=J TO N
3230 A(J, K)=A(J, K)-R1 * A(I, K)+R2 * B(I, K)
3232 B(J, K)=B(J, K)-R1 * B(I, K)-R2 * A(I, K)
3234 NEXT K
3236 X(J)=X(J)-R1 * X(I)+R2 * Y(I)
3238 Y(J)=Y(J)-R1 * Y(I)-R2 * X(I)
3240 NEXT J
3242 NEXT I
3244 GOSUB 3042
3246 RETURN

```

### 3.3. Программа SINUS на языке БЕЙСИК

С помощью программы SINUS выполняется анализ линейной цепи, содержащей следующие элементы: активные проводимость или сопротивление; индуктивность  $L=\text{const}$ ;  $C=\text{const}$ ; управляемый источник тока с постоянным вещественным коэффициентом управления, идеальный источник тока  $I_q=\text{const}$ ; идеальный источник напряжения  $U_q=\text{const}$ .

Активное сопротивление или проводимость  $G=\text{const}$ , индуктивность  $L=\text{const}$  и емкость  $C=\text{const}$  называются базовыми двухполюсниками.

Идеальные источники обеспечивают независимо от нагрузки постоянные ток или напряжение.

Линейные управляемые источники напряжения требуется до ввода структуры цепи в ЭВМ преобразовать в управляемые источники тока. Однако подобное преобразование в случае идеальных источников напряжения невыполнимо. Поэтому такие источники рассчитываются по программе SINUS.

Взаимная индуктивность может быть описана с помощью схемы замещения, приведенной в п. 3.1.4, если предварительно определены ее элементы.

**3.3.1. Ввод структуры цепи.** После ввода глобальных переменных (см. п. 2.5.1) в ЭВМ вводится структура цепи. Каждая нагрузочная ветвь может состоять из параллельно включенных базовых двухполюсников, идеального источника синусоидального тока и управляемого источника тока (рис. 3.9).

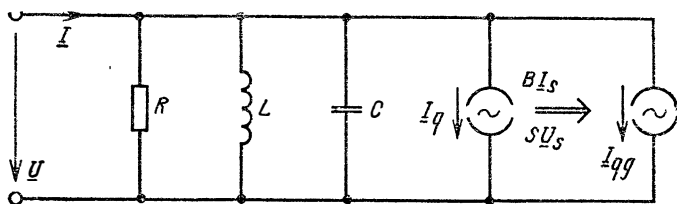


Рис. 3.9. Нагрузочная ветвь цепи синусоидального напряжения

Параллельное включение элементов позволяет непосредственно найти решение с помощью метода узловых потенциалов даже при наличии резонанса в одной из ветвей. Схема последовательного включения базовых двухполюсников на резонансной частоте имеет реактивное сопротивление  $X=0$ , которое должно быть замещено, как и в случае короткого замыкания, идеальным источником напряжения с  $\underline{U}_q=0$ .

Подобное замещение требует существенных программных затрат. Чтобы избежать их, в программе SINUS каждый элемент схемы определяется как ветвь. Ввод и запоминание структуры цепи выполняется подпрограммой UPR 3500 на базе UPR 2500 (см. п. 2.5.2). К предусмотренным в подпрограммах ключевым символам G, R, I, S и N добавляются еще C и L, характеризующие

ющие индуктивность и емкость. Необходимо вводить только такие элементы, численные значения которых отличаются от нуля, потому что все остальные элементы по умолчанию устанавливаются равными нулю (строки 3514—3524). При этом  $MZ(I) = 0$  означает, что в ветви I нет ни одного управляемого источника тока.

Собственный ток идеального источника тока и собственное напряжение идеального источника напряжения вводятся в ЭВМ в R-форме (строки 3566 и 3568, а также 3584 и 3586). Как и в программе GLEICH, ветви цепи необходимо предварительно пронумеровать таким образом, чтобы идеальный источник напряжения получил старший номер ветви, т. е.  $I > IR$ .

**Подпрограмма UPR 3500 (она вызывает UPR 2400)**

а) необходимые переменные в вызывающей программе:

I — номер ветви;

IR — количество нагрузочных ветвей;

б) используемые имена переменных:

IL — номер идеального источника напряжения;

R1 — сопротивление;

в) результат:

G(I); GD(I); IG; KA(IL); KU(IL); KV(I); KZ(I); MZ(I);  
QT(IL); QI(IL); SD(I); SI(I); SF(I)... см. приложение ПЗ.

3500 REM : Ввод элементов цепи для ветви I

3502 REM

3504 INPUT «из узла»; J

3506 INPUT «к узлу»; K

3508 IF I > IR THEN 3576

3510 KV(I) = J

3512 KZ(I) = K

3514 GD(I) = 0

3516 L(I) = 0

3518 C(I) = 0

3520 SD(I) = 0

3522 SI(I) = 0

3524 MZ(I) = 0

3526 INPUT «Элемент»; FR\$

3528 IF FR\$ = «R» THEN 3546

3530 IF FR\$ = «G» THEN 3552

3532 IF FR\$ = «L» THEN 3556

3534 IF FR\$ = «C» THEN 3560

3536 IF FR\$ = «I» THEN 3564

3538 IF FR\$ = «S» THEN 3572

3540 IF FR\$ = «N» THEN 3588

3542 PRINT «Ввод не имеет смысла»

3544 GOTO 3526

3546 INPUT «Сопротивление»; R1

3548 GD(I) = 1/R1

3550 GOTO 3526

3552 INPUT «Проводимость»; GD(I)

3554 GOTO 3526

```

3556 INPUT «Индуктивность»; L(I)
3558 GOTO 3526
3560 INPUT «Емкость»; C(I)
3562 GOTO 3526
3564 PRINT «Ток источника»
3566 INPUT «Действительная часть»; SD(I)
3568 INPUT «Мнимая часть»; SI(I)
3570 GOTO 3526
3572 GOSUB 2400
3574 GOTO 3526
3576 IL=I-IR
3578 KA(IL)=J
3580 KU(IL)=K
3582 PRINT «Напряжение источника»
3584 INPUT «Действительная часть»; QT(IL)
3586 INPUT «Мнимая часть»; QI(IL)
3588 RETURN

```

**3.3.2. Частота.** Для определения зависимости параметров ветви от частоты необходимо проанализировать цепь при нескольких конкретных значениях частоты. Частота  $F$  синусоидального источника изменяется с помощью оператора цикла (строка 202). Предварительно в ЭВМ вводятся: начальное значение частоты  $F_A$ , ее приращение  $FS$  и конечное значение частоты  $F_E$  (строки 196—200). Параметры частоты задаются с обычной точностью. Варьируя  $FS$  и  $F_E$ , можно рассчитать цепь при заданном значении  $F_A$ . Для каждого значения частоты выдаются только интересующие пользователя параметры ветви (строки 364—410). Их тип и количество определяются в подпрограмме UPR 3400.

#### Подпрограмма UPR 3400

а) необходимые переменные вызывающей программы:  
 IR — количество нагрузочных ветвей;  
 б) используемые имена переменных:  
 I — переменная цикла;  
 в) результат:

|        |                                           |
|--------|-------------------------------------------|
| MA     | — количество выводимых напряжений ветвей; |
| MB     | — количество выводимых токов ветвей;      |
| MI(IR) | — номера ветвей для выводимых токов;      |
| MU(IR) | — номера ветвей для выводимых напряжений. |

```

3400 REM : Определение выводимых величин
3402 REM
3404 FOR I=1 TO IR
3406 MI(I)=I
3408 MU(I)=I
3410 NEXT
3412 PRINT
3414 PRINT «Какие параметры ветви»
3416 PRINT «должны быть выведены?»
3418 INPUT «Количество напряжений ветвей»; MA

```

```

3420 IF MA<1 THEN 3434
3422 IF MA=IR THEN 3434
3424 PRINT «Номера ветвей»
3426 FOR I=1 TO MA
3428 INPUT MU(I)
3430 NEXT
3432 PRINT
3434 INPUT «Количество токов ветвей»; MB
3436 IF MB<1 THEN 3448
3438 IF MB=IR THEN 3448
3440 PRINT «Номера ветвей»
3442 FOR I=1 TO MB
3444 INPUT MI (I)
3446 NEXT
3448 RETURN

```

**3.3.3. Перемещение идеальных источников напряжения.** После ввода в ЭВМ параметров цепи осуществляется перемещение идеальных источников. Для этого разработана подпрограмма UPR 3600, являющаяся комплексной версией подпрограммы UPR 2600. В UPR 3600 как собственное напряжение идеального источника напряжения  $QT(IL) + jQI(IL)$ , так и напряжение источника, соответствующее нагрузочной ветви,  $VR(I) + jVI(I)$  являются комплексными величинами.

#### Подпрограмма UPR 3600

а) необходимые переменные вызывающей программы:  
 IR, IS; KA(IS); KV(IR); KZ(IR), QI(IS); QT(IS)  
 см. приложение ПЗ;

б) используемые имена переменных: см. подпрограмму UPR 2600;

в) результат:  
 $VI(IR)$  — мнимая часть;  
 $VR(IR)$  — действительная часть напряжения источника, соответствующего нагрузочной ветви.

```

3600 REM : Перемещение идеальных источников напряжения
3602 REM
3604 FOR I=1 TO IR
3606 VR(I)=0
3608 VI(I)=0
3610 IF IS=0 THEN 3630
3612 FOR IL=1 TO IS
3614 JV=1
3616 IF KV(I)=KA(IL) THEN 3624
3618 IF KZ(I)<KA(IL) THEN 3622
3620 GOTO 3628
3622 JV=-1
3624 VR(I)=VR(I)+JV * QT(IL)
3626 VI(I)=VI(I)+JV* QI(IL)
3628 NEXT IL
3630 NEXT I
3632 RETURN

```

**3.3.4. Составление и решение системы уравнений.** Система комплексных уравнений находится в строках 218—332 основной программы. Эта часть программы сформулирована на основе преобразования в комплексную форму подпрограммы UPR 2700. Комплексными величинами являются не только матрица коэффициентов  $A(N,N) + jB(N,N)$  и правая часть или вектор решения  $X(N) + jY(N)$ , но и проводимость  $GD(I) + jGI(I)$ , а также вспомогательные величины  $G1 + jG2$  и  $S1 + jS2$ . Если управляемых источников нет, система уравнений решается подпрограммой UPR 3200, в противном случае — подпрограммой UPR 3000.

```

100 PRINT «SINUS»
102 PRINT «Узловой анализ линейной»
104 PRINT «цепи переменного тока»
106 PRINT «с управляемыми источниками тока»
108 PRINT
110 DEFINT I—K, M—N
112 DEFDBL A—D, G, L, P—Z
114 REM
116 REM: Ввод глобальных параметров
118 INPUT «Количество ветвей»; IZ
120 INPUT «Количество узлов без базового»; KN
122 INPUT «Количество идеальных источн. напряж. »; IS
124 PRINT
126 N=KN-IS
128 IF IZ<IS THEN 132
130 IF N>0 THEN 138
132 PRINT «Ввод не имеет смысла»
134 PRINT «или обработка невозможна»
136 GOTO 118
138 IR=IZ-IS
140 REM
142 DIM UN(KN), UI(KN), KF(KN)
144 DIM A(N, N), B(N, N), X(N), Y(N)
146 DIM KA(IS), KU(IS), QT(IS), QI(IS)
148 DIM SF(IR), MZ(IR), VR(IR), VI(IR), C(IR), L(IR)
150 DIM KV(IR), KZ(IR), GD(IR), GI(IR), SD(IR), SI(IR)
152 DIM SA(IR), SB(IR), UA(IR), UB(IR), MI(IR), MU(IR)
154 REM
156 PRINT «Ввод структуры цепи»
158 REM
160 IG=0
162 FOR I=1 TO IZ
164 PRINT «Ветвь»; I
166 GOSUB 3500
168 NEXT
170 REM
172 REM: Определение массива узлов
174 GOSUB 2100
176 REM: Подготовка перемещения

```

```

178 REM: идеальных источников напряжения
180 GOSUB 8600
182 REM
184 REM : Определение выводимых величин
186 GOSUB 8400
188 REM
190 PI=3.14159265358979 #
192 PRINT
194 PRINT «Частота»
196 INPUT «Начальное значение»; FA
198 INPUT «Значение шага»; FS
200 INPUT «Конечное значение»; FE
202 REM
204 FOR F=FA TO FE STEP FS
206 W=2 * PI * F
208 PRINT
210 PRINT «f=»; F; «Hz»
212 REM
214 REM: Составление и решение
216 REM: системы линейных уравнений
218 FOR I=1 TO N
220 X(I)=0
222 Y(I)=0
224 FOR J=1 TO N
226 A(I, J)=0
228 B(I, J)=0
230 NEXT J
232 NEXT I
234 REM
236 FOR I=1 TO IR
238 GI(I)=W * C(I)
240 IF L(I)=0 THEN 244
242 GI(I)=GI(I)-1/(W * L(I))
244 S1=SD(I)+GD(I) * VR(I)-GI(I) * VI(I)
246 S2=SI(I)+GI(I) * VR(I)+GD(I) * VI(I)
248 J=KV(I)
250 K=KZ(I)
252 JR=KF(J)
254 KR=KF(K)
256 IF JR=KR THEN 332
258 A(JR, JR)=A(JR, JR)+GD(I)
260 B(JR, JR)=B(JR, JR)+GI(I)
262 A(JR, KR)=A(JR, KR)-GD(I)
264 B(JR, KR)=B(JR, KR)-GI(I)
266 A(KR, JR)=A(KR, JR)-GD(I)
268 B(KR, JR)=B(KR, JR)-GI(I)
270 A(KR, KR)=A(KR, KR)+GD(I)
272 B(KR, KR)=B(KR, KR)+GI(I)
274 REM
276 IF MZ(I)=0 THEN 324
278 MS=ABS(MZ(I))
280 IF MZ(I)<0 THEN 292
282 G1=SF(I)
284 G2=0
286 S1=S1+SF(I) * VR(MS)

```



```

288 S2=S2+SF(I)* VI(MS)
290 GOTO 300
292 G1=SF(I) * GD(MS)
294 G2=SF(I) * GI(MS)
296 S1=S1+G1* VR(MS)-G2* VI(MS)+SF(I) * SD(MS)
298 S2=S2+G2 * VR(MS)+G1 * VI(MS)+SF(I) * SI(MS)
300 JS=KV(MS)
302 KS=KZ(MS)
304 JS=KF(JS)
306 KS=KF(KS)
308 A(JR, JS)=A(JR, JS)+G1
310 B(JR, JS)=B(JR, JS)+G2
312 A(JR, KS)=A(JR, KS)-G1
314 B(JR, KS)=B(JR, KS)-G2
316 A(KR, JS)=A(KR, JS)-G1
318 B(KR, JS)=B(KR, JS)-G2
320 A(KR, KS)=A(KR, KS)+G1
322 B(KR, KS)=B(KR, KS)+G2
324 X(JR)=X(JR)-S1
326 Y(JR)=Y(JR)-S2
328 X(KR)=X(KR)+S1
330 Y(KR)=Y(KR)+S2
332 NEXT I
334 REM
336 EP=1E-12
338 IF IG=0 THEN 346
340 GOSUB 3300
342 GOSUB 3000
344 GOTO 354
346 GOSUB 3200
348 REM
350 REM
352 REM : Вычисление узловых напряжений
354 GOSUB 3800
356 REM
358 REM : Вычисление и вывод
360 REM : параметров ветвей
362 GOSUB 3900
364 IF MA=0 THEN 382
366 FOR I=1 TO MA
368 Y=MU(I)
370 IF J>IR THEN 380
372 AR=UA(Y)
374 BI=UB(Y)
376 GOSUB 31
378 IW=100 * PH+.5 * SGN(PH)
380 EW=.01 * IW
382 ER=RP
384 PRINT «U»; J; «=»; ER; «V/»; EW; «x»
386 NEXT
388 IF MB=Ø THEN 414
390 FOR I=1 TO MB
392 J=MI(I)
394 IF J>IR THEN 410
396 AR=SA(J)

```

```

398 BI=SB(J)
400 GOSUB 3100
402 IW=100 * PH+.5 * SGN(PH)
404 EW=.01 * IW
406 ER=RP
408 PRINT «I»; J; «=»; ER; «A/»EW; «х»
410 NEXT
412 REM
414 NEXT F
416 PRINT
418 INPUT «Следующее значение частоты? (J/N) .»;FR$
420 IF FR$ =«J» THEN 192
422 IF FR$ =«N» THEN 430
424 GOTO 416
426 REM
428 REM: Изменение параметров ветви
430 PRINT
432 PRINT «В скольких ветвях должны быть»
434 INPUT изменены элементы?»
436 IF ME<1 THEN 9999
438 FOR ML=1 TO ME
440 INPUT «Ветвь»;I
442 GOSUB 3500
444 NEXT
446 GOTO 174
1000 REM:UPR 2100, 2400
1002 REM: UPR 3000, 3100, 3200, 3300, 3400
1004 REM; UPR 3500, 3600, 3800, 3900

```

**3.3.5. Расчет и распечатка параметров ветви.** После решения системы комплексных уравнений с помощью подпрограммы UPR 3800, которая является комплексной версией подпрограммы UPR 2800, определяются узловые напряжения  $UN(KN) + jUI(KN)$ .

#### Подпрограмма UPR 3800

а) необходимые переменные и массивы в вызывающей программе (см. подробности в приложении ПЗ):

IS;KA(IS);KF(IS);KN;KU(IS);QI(IS);QT(IS);X(N);Y(N);

б) используемые имена переменных:

I,L — номер идеального источника напряжения;

J,K — номер узлов;

в) результат:

UI(KN) — мнимая часть узлового напряжения;

UN(KN) — действительная часть узлового напряжения.

3800 REM: Вычисление узловых напряжений

3802 REM

3804 FOR J=1 TO KN

3806 K=KF(J)

3808 UN(J)=X(K)

3810 UI(J)=Y(K)

3812 NEXT

3814 UN(0)=0

3816 UI(0)=0

```

3818 IF IS=0 THEN 3832
3820 FOR IL=1 TO IS
3822 J=KA(IL)
3824 K=KU(IL)
3826 UN(J)=UN(K)+QT(IL)
3828 UI(J)=UI(K)+QI(IL)
3830 NEXT
3832 RETURN

```

В табл. 3.1 сопоставляются реальные величины программы GLEICH и соответствующие комплексные величины программы SINUS.

**Таблица 3.1. Сопоставление комплексных величин двух программ (пояснение см. в приложении ПЗ)**

| GLEICH  | SINUS                 |
|---------|-----------------------|
| A(N, N) | $A(N, N) + j B(N, N)$ |
| GD(IR)  | $GD(IR) + j GI(IR)$   |
| GI      | $GI + j G2$           |
| QT(IS)  | $QT(IS) + j QI(IS)$   |
| SA(IR)  | $SA(IR) + j SB(IR)$   |
| SD(IR)  | $SD(IR) + j SI(IR)$   |
| S1      | $S1 + j S2$           |
| UA(IR)  | $UA(IR) + j UB(IR)$   |
| UN(KN)  | $UN(KN) + j UI(KN)$   |
| VR(IR)  | $VR(IR) = j VI(IR)$   |
| X(N)    | $X(N) = j Y(N)$       |

Подпрограмма UPR 3900, с помощью которой вычисляются параметры ветви, является комплексной версией подпрограммы UPR 2900. После определения всех параметров ветвей они выводятся подпрограммой UPR 3400 в P-форме с обычной точностью. Вычисления параметров повторяются для каждого следующего значения частоты. После завершения цикла по частоте могут (строка 414) быть заданы другие ее значения или, как в программе GLEICH, изменены элементы в ME ветвях.

#### Подпрограмма UPR 3900

а) — в) см. UPR 2900; однако величины GD(IR), SD(IR), UN(KN), S1, SA(IR), UA(IR) являются комплексными (см. табл. 3.1).

```

3900 REM: Вычисление параметров ветвей
3902 REM
3904 FOR I=1 TO IR
3906 J=KV(I)
3908 K=KZ(I)
3910 UA(I)=UN(J)—UN(K)

```

```

3912 UB(I)=UI(J)-UI(K)
3914 SA(I)=GD(I)*UA(I)-GI(I)*UB(I)+SD(I)
3916 SB(I)=GI(I)*UA(I)+GD(I)*UB(I)+SI(I)
3918 NEXT
3920 FOR I=1 TO IR
3922 MS=ABS(MZ(I))
3924 IF MS=0 THEN 3942
3926 IF MZ(I)<0 THEN 3934
3928 S1=SF(I)*UA(MS)
3930 S2=SF(I)*UB(MS)
3932 GOTO 3938
3934 S1=SF(I)*SA(MS)
3936 S2=SF(I)*SB(MS)
3938 SA(I)=SA(I)+S1
3940 SB(I)=SB(I)+S2
3942 NEXT
3944 RETURN

```

**3.3.6. Примеры.** С помощью программы SINUS можно рассчитывать и трехфазные схемы. При этом необходимо иметь в виду, что идеальные источники напряжения, собранные по схеме «треугольник» включать не допустимо (см. п. 2.2.1). Если напряжения источников  $\underline{U}_{12}$ ,  $\underline{U}_{23}$ ,  $\underline{U}_{31}$  симметричны, то для схемы «треугольник» (рис. 3.10, а) можно сразу же сформировать схему замещения (рис. 3.10, б). В результате преобразования

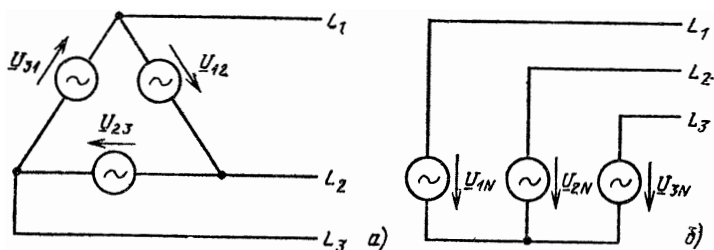


Рис. 3.10. Недопустимая в методе узловых потенциалов схема «треугольник» из идеальных источников напряжения (а) и допустимая схема замещения (б)

симметричных фазных напряжений в схеме «треугольник»

$$\left. \begin{aligned} \underline{U}_{12} &= \sqrt{3}U_{\angle} \angle 30^\circ; \\ \underline{U}_{23} &= \sqrt{3}U_{\angle} \angle -90^\circ; \\ \underline{U}_{31} &= \sqrt{3}U_{\angle} \angle 150^\circ. \end{aligned} \right\} \quad (3.3-1)$$

Получаем симметричные фазные напряжения в схеме «звезда»

$$\left. \begin{aligned} \underline{U}_{1N} &= U_{\lambda}; \\ \underline{U}_{2N} &= U_{\lambda} \angle -120^{\circ}; \\ \underline{U}_{3N} &= U_{\lambda} \angle 120^{\circ}, \end{aligned} \right\} \quad (3.3-2)$$

где  $U_{\lambda}$  — эффективное значение фазного напряжения в схеме «звезда».

Если напряжения источников, включенных по схеме «треугольник», несимметричны, то трехфазную схему необходимо исследовать посредством метода симметричных составляющих [32].

Рассчитаем фазные напряжения и токи несимметричной трехфазной нагрузки, включенной по схеме «звезда» в трехфазную сеть (рис. 3.11, в R-форме при

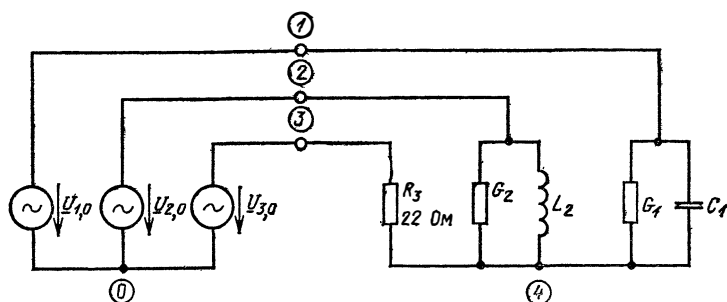


Рис. 3.11. Схема «звезда» несимметричной трехфазной нагрузки в трехфазной сети

эффективным значением симметричного напряжения  $\sqrt{3} \cdot 220$  В:

$$\begin{aligned} U_{1,0} &= 220 \text{ В}, \\ U_{2,0} &= -110 - j190,5 \text{ В}, \\ U_{3,0} &= -110 - j190,5 \text{ В}. \end{aligned}$$

Параметры двухполюсников  $G_1 = 9,397$  мСм,  $C_1 = 10,887$  мкФ,  $G_2 = 22,727$  мСм и  $L_2 = 80,864$  мГн выбираются таким образом, чтобы для 50 Гц комплексные сопротивления  $Z_1 = 100 \text{ Ом} \angle -20^{\circ}$ , а  $Z_2 = 22 \text{ Ом} \angle 60^{\circ}$  [12].

RUN

SINUS

Узловой анализ линейных  
цепей переменного тока  
с управляемыми источниками тока

Количество ветвей? 6

Количество узлов (без базового)? 4

Количество идеальных источников напряжения? 3

Ввод структуры цепи

Ветвь 1

из узла? 1

к узлу? 4

Элемент? G

Проводимость?. 009397

Элемент? C

Емкость? 10.887E—6

Элемент? N

Ветвь 2

из узла? 2

к узлу? 4

Элемент? G

Проводимость?. 022727

Элемент? L

Индуктивность?. 0808674

Элемент? N

Ветвь 3

из узла? 3

к узлу? 4

Элемент? R

Сопротивление? 22

Элемент? N

Ветвь 4

из узла? 1

к узлу? 0

Напряжение источника

Действительное? 220

Мнимое? 0

Ветвь 5

из узла? 2

к узлу? 0

Напряжение источника

Действительное? —110

Мнимое? —190.5

Ветвь 6

из узла? 3

к узлу? 0

Напряжение источника

Действительное? —110

Мнимое? 190.5

Какие параметры ветвей  
должны быть введены?

Количество напряжения ветвей? 3

Количество токов ветвей? 3

Частота

Начальное значение? 50

Значение шага? 10  
 Конечное значение? 51  
 $f = 50$  Гц  
 $U_1 = 405,1364$  В/ $-5,15^\circ$   
 $U_2 = 238,4158$  В/ $-72,05^\circ$   
 $U_3 = 170,823$  В/ $64,52^\circ$   
 $I_1 = 4,051399$  А/ $14,86^\circ$   
 $I_2 = 10,8368$  А/ $-132,05^\circ$   
 $I_3 = 7,764681$  А/ $64,52^\circ$   
 Следующие значения частоты? (J/N)? N  
 Количество ветвей, в которых  
 будут изменяться элементы? 0  
 Ок

В качестве следующего примера рассчитаем частотную характеристику селективного фильтра (рис. 3.12) со сложной обратной связью [55]. Операционный усилитель типа  $\mu A 741$  упрощенно описывается схемой, приведенной на рис. 2.16.

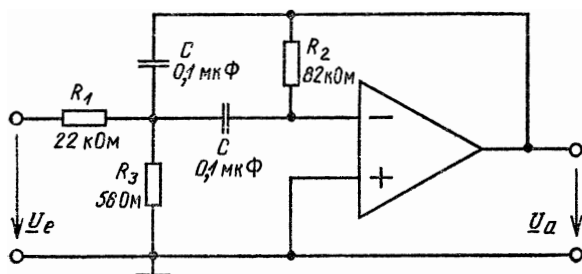


Рис. 3.12. Полосовой фильтр

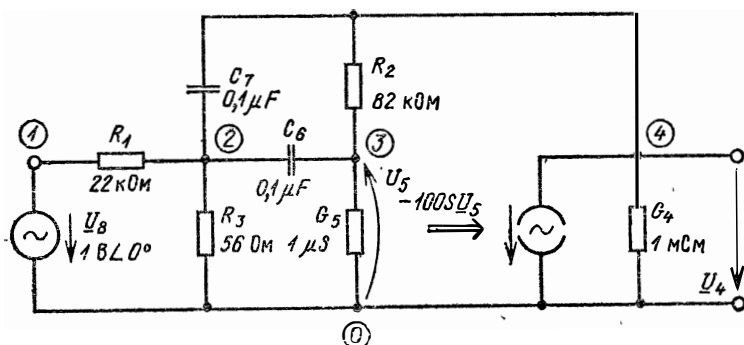


Рис. 3.13. Полосовой фильтр со схемой замещения операционного усилителя

С учетом схемы замещения операционного усилителя на рис. 2.15 и полосового фильтра на рис. 3.12 получим «комбинированную» схему полосового фильтра (рис. 3.13). Частотная характеристика коэффициента передачи  $A = U_a / U_e$  при  $U_e = U_8 = 1$  В  $\angle 0^\circ$  является численным значением выходного напряжения  $U_q = U_4$ . Для расчета ширины полосы пропускания разделим коэффициент передачи  $A_r = 1,85$  при резонансной частоте 743,6 Гц на коэффициент  $\sqrt{2}$  и найдем частоту, для которой  $A_r = 1,308$ . Если  $f_{gu} = 724,3$  Гц и  $f_{g0} = 763,4$  Гц, то ширина полосы  $b_f = f_{g0} - f_{gu} = 39,1$  Гц.

Теперь определим добротность фильтра:

$$Q = \frac{f_r}{b_f} = \frac{743,6}{39,1} = 19.$$

На резонансной частоте  $f_r$  входное  $U_e$  и выходное  $U_a$  напряжения находятся в противофазе. В заключение рассчитаем коэффициент передачи  $A$  для нескольких значений частоты и графически представим зависимости  $A$  и угла  $\varphi$  от частоты (рис. 3.14):

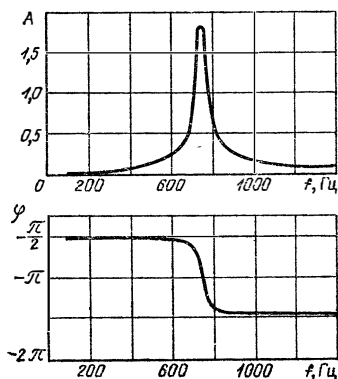


Рис. 3.14. Амплитудная и фазовая характеристики полосового фильтра рис. 3.13

RUN  
SINUS

Узловой анализ линейной  
цепи переменного тока

с управляемыми источниками тока

Количество ветвей? 8

Количество узлов (без базового)? 4

Количество идеальных источников напряжения? 1

Ввод структуры цепи

Ветвь 1

из узла? 1

к узлу? 2

Элемент? R

Сопротивление? 22000

Элемент? N

Ветвь 2

из узла? 4

к узлу? 3



Элемент? R  
 Сопротивление? 82000  
 Элемент? N  
 Ветвь 3  
     из узла? 2  
     к узлу? 0  
 Элемент? R  
 Сопротивление? 56  
 Элемент? N  
 Ветвь 4  
     из узла? 4  
     к узлу? 0  
 Элемент? G  
 Проводимость?. 001  
 Элемент? S  
 Управляемый источник тока  
 Управляющий параметр?  $(I/U)$ ? U  
 Коэффициент управления  $S=?-100$   
 Управляющая ветвь? 5  
 Элемент? N  
 Ветвь 5  
     из узла? 0  
     к узлу? 3  
 Элемент? G  
 Проводимость? 1.E-6  
 Элемент? N  
 Ветвь 6  
     из узла? 2  
     к узлу? 3  
 Элемент? C  
 Емкость?. 1-6  
 Элемент? N  
 Ветвь 7  
     из узла? 4  
     к узлу? 2  
 Элемент? C  
 Емкость?. 1E-6  
 Элемент? N  
 Ветвь 8  
     из узла? 1  
     к узлу? 0  
 Напряжение источника  
 Действительное? 1  
 Мнимое? 0  
 Какие параметры ветвей  
 должны быть выведены?  
 Количество напряжений ветвей? 1  
 Номера ветвей  
     ? 4  
 Количество токов ветвей? 0  
 Частота  
 Начальное значение? 743  
 Значение шага?. 2  
 Конечная величина? 744  
 $f=743$  Гц

$$U_4 = 1,848857 \text{ В} / -178,28^\circ$$

$$f = 743,2 \text{ Гц}$$

$$U_4 = 1,84933 \text{ В} / -178,87^\circ$$

$$f = 743,4 \text{ Гц}$$

$$U_4 = 1,849609 \text{ В} / -179,45^\circ$$

$$f = 743,6001 \text{ Гц}$$

$$U_4 = 1,849694 \text{ В} / 179,97^\circ$$

$$f = 743,8001 \text{ Гц}$$

$$U_4 = 1,849586 \text{ В} / 179,38^\circ$$

Следующие значения частоты? (J/N)? N

Количество ветвей, в которых

будут изменяться элементы? 0

Ок

## Глава 4

### ЛИНЕЙНЫЕ ДИНАМИЧЕСКИЕ ЦЕПИ

В предыдущих главах были исследованы цепи в установившихся режимах, когда все их параметры либо постоянны во времени, либо, для синусоидальных величин, постоянны по амплитуде. Подобные режимы работы цепей называются статическими, и считается, что временная зависимость параметров этих цепей известна.

Рассчитаем параметры цепи, структура которой изменяется в течение переходного процесса. После подключения источника питания в момент  $t=0$  и до перехода в установившийся режим параметры цепи меняются во времени до тех пор, пока не примут окончательные значения. В этом переходном состоянии цепь называется динамической.

#### 4.1. Структура динамической цепи

Ниже речь пойдет исключительно о цепях с линейными двухполюсниками и управляемыми источниками, коэффициент управления которых постоянен. Линейными двухполюсниками являются базовые двухполюсники  $R$ ,  $G$ ,  $L$  и  $C$  и идеальные источники, собственные параметры которых не зависят от нагрузки.

В дополнение к уже перечисленным в § 2.2 требованиям к структуре цепей идеальные источники вносят свои ограничения. Некоторые из них определяются тем, что напряжение на зажимах идеального источника напряжения уже задано. Поэтому двухполюсник, парамет-

ром состояния которого является напряжение, может быть связан с идеальным источником напряжения только в установившемся режиме.

Например, в цепи с синусоидальным напряжением можно соединять идеальный источник напряжения с емкостью, поскольку исследуется установившийся режим, для которого в каждый момент времени  $u_q = u_c$ . В динамической цепи для  $t \geq 0$  связь идеального источника напряжения с емкостью (рис. 4.1, а) недопустима, так как

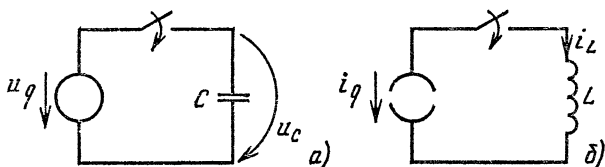


Рис. 4.1. Недопустимое соединение идеального источника с двухполюсником в динамической цепи

в общем случае при  $u_q \neq u_c$  не выполняется закон контурных токов. В соответствии с этим цепь не может состоять из последовательно включенных емкостей и идеальных источников напряжения.

Следующая часть ограничений в отношении структуры цепи базируется на том, что идеальный источник тока определяет значение тока в ветви. Связь этого источника тока с индуктивностью в соответствии с рис. 4.1, б недопустима, поскольку в общем случае при  $i_q \neq i_L$  не выполняется закон узловых потенциалов. Вообще говоря, это означает, что в динамической цепи нельзя задавать узлы, связывающие друг с другом только источники тока и индуктивности.

## 4.2. Анализ во времени

**4.2.1. Анализ переходного процесса.** Если цепь содержит индуктивность  $L$  или емкость  $C$ , то аналитически параметры цепи, зависящие от времени, можно рассчитывать только путем решения дифференциальных уравнений. На рис. 4.2 показан простой пример такой цепи, в которой емкость подключается к линейному источнику постоянного напряжения.

В начальный момент времени  $t=0$   $u_c = u_A$ . При по-

стоянной времени  $\tau = R_E C$  решение [12] выглядит следующим образом:

$$u_C = U_E + (U_A - U_E)e^{-t/\tau}. \quad (4.2-1)$$

Аналитическое решение для переходных значений параметров динамической цепи можно получить также с помощью преобразования Лапласа. Однако при реализации этого метода на ЭВМ возникает ряд трудностей [12].

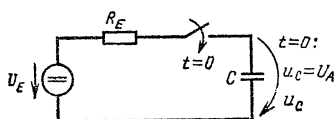


Рис. 4.2. Включение емкости

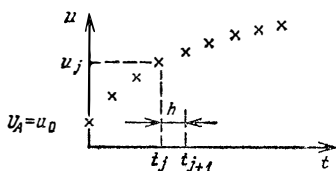


Рис. 4.3. Дискретизация напряжения по времени

Цифровые методы расчета предназначаются для реализации на ЭВМ и могут быть использованы в случае нелинейных цепей при анализе переходных процессов. Они позволяют исследовать физические процессы во времени. Аналитические методы расчета способствуют получению математического выражения для параметров цепи во всем временном диапазоне [например, уравнение (4.2-1)].

**4.2.2. Дискретизация по времени.** Как уже упоминалось, при анализе переходных процессов определяются мгновенные значения каждого параметра цепи для дискретных моментов времени. На основании начальных условий ( $t=0$ ) вычисляются параметры цепи сначала в момент  $t_1$ , затем в моменты  $t_2, t_3, \dots, t_E$ . В конечный момент  $t_E$  вычисления прекращаются.

Индекс  $j$  напряжения  $u_j$  или тока  $i_j$  указывает на момент времени  $t_j$ , в который определяется тот или иной параметр (рис. 4.3). В этом случае говорят, что параметр цепи дискретизирован по времени.

Каждый параметр вычисляется на основании значений, полученных в предыдущие моменты времени. Если, например, напряжение  $u_1$  определяется исходя из начального значения  $U_A$ , то при вычислении  $u_2$  в качестве начального значения выбирается  $u_1$ . В общем случае обозначим последние уже вычисленные значения пара-

метров цепи индексом  $j$ , а еще неизвестные параметры, которые предстоит определить на следующем шаг вычислений, — индексом  $j+1$ .

Интервал между двумя расположенными друг за другом моментами времени

$$h = t_{j+1} - t_j. \quad (4.2-2)$$

При переменном шаге дискретизации символ  $h$  также должен был бы иметь индекс. Однако в данном случае это лишнее, поскольку  $h$  обозначает текущее значение шага.

Пример:  $h = \text{const}$ . В этом случае значения времени эквидистантны, т.е. расположены на одинаковых расстояниях друг от друга по оси времени.

### 4.3. Метод Эйлера

При анализе переходных процессов цепи с несколькими реактивными элементами необходимо для каждого момента времени  $t_{j+1}$  решить систему обыкновенных дифференциальных уравнений. Рассмотрим в качестве примера дифференциальное уравнение первого порядка, которое содержит кроме переменной  $x$  еще и производную первого порядка от этой переменной

$$x' = dx/dt = \lambda x. \quad (4.3-1)$$

Для дискретного вычисления значения  $x_{j+1}$  в момент

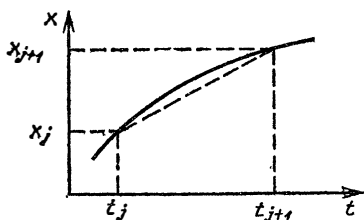


Рис. 4.4. Определение тангенса угла наклона кривой в методе Эйлера

времени  $t_{j+1}$  необходимо определить производную  $dx/dt$ . Воспользуемся для этого методом конечных приращений Эйлера. Заметим, что функцию  $x(t)$  между точками  $t_j$  и  $t_{j+1}$  можно аппроксимировать прямой линией (рис. 4.4). Наклон прямой

$$\frac{dx}{dt} = \frac{x_{j+1} - x_j}{t_{j+1} - t_j} = \frac{x_{j+1} - x_j}{h}. \quad (4.3-2)$$

Уравнение (4.3-2) описывает производную как в момент времени

$$x'(t_j) = \frac{x_{j+1} - x_j}{h}, \quad (4.3-3)$$

так и в момент времени  $t_{j+1}$ :

$$x'(t_{j+1}) = \frac{x_{j+1} - x_j}{h}. \quad (4.3-4)$$

Решим (4.3-4) относительно искомой величины:

$$x_{j+1} = x_j + hx'(t_{j+1}). \quad (4.3-5)$$

Если в обеих частях уравнения находятся неизвестные, то оно называется неявным. Поэтому метод конечных приращений для решения дифференциальных уравнений типа (4.3-5) также называется неявным или обратным методом Эйлера.

Решим теперь (4.3-3) относительно искомой величины:

$$x_{j+1} = x_j + hx'(t_j). \quad (4.3-6)$$

Уравнения, содержащие неизвестные только в одной части, называются явными, а метод конечных приращений для решения дифференциальных уравнений (4.3-6) называется соответственно явным методом Эйлера.

**4.3.1. Устойчивость.** При нестабильности дискретного метода решение дифференциальных уравнений с увеличивающейся амплитудой колеблется около искомого значения. Устойчивость метода в значительной степени зависит от значения шага. Покажем это на примере (4.3-1), причем коэффициент

$$\lambda = \alpha + j\beta. \quad (4.3-7)$$

При начальном значении  $x(t=0) = x_0$  решение выглядит следующим образом:

$$x(t) = x_0 e^{\lambda t} = x_0 e^{\alpha t} (\cos \beta t + j \sin \beta t). \quad (4.3-8)$$

Если  $\alpha < 0$ , то функция  $x(t)$  описывает затухающие колебания с круговой частотой  $\beta$ . При  $t \rightarrow \infty$   $x(t) \rightarrow 0$ . Будем считать метод, при котором решение уравнения (4.3-1) для  $\alpha < 0$  и  $t \rightarrow \infty$  стремится к нулю, стабильным.

Прежде всего исследуем устойчивость явного метода Эйлера и для этого подставим (4.3-1) в (4.3-6). Тогда

$$x_{j+1} = x_j + hx' = x_j + h\lambda x_j. \quad (4.3-9)$$

Чтобы определить зависимость решения  $x_{j+1}$  от начального значения  $x_0$ , положим в уравнении (4.3-6)  $j=0$  и  $j=1$  и запишем

$$\begin{aligned}x_1 &= x_0 + h\lambda x_0 = (1 + h\lambda) x_0, \\x_2 &= (1 + h\lambda) x_1 = (1 + h\lambda)^2 x_0.\end{aligned}$$

Отсюда можно вывести правило образования всех значений:

$$x_j = (1 + h\lambda)^j x_0. \quad (4.3-10)$$

Если метод стабилен, то для  $j \rightarrow \infty$  решение стремится к нулю. Таким образом, из (4.3-10) получаем условие устойчивости явного метода Эйлера в виде

$$|1 + h\lambda| < 1. \quad (4.3-11)$$

Теперь исследуем устойчивость неявного метода Эйлера. Подставим (4.3-1) в (4.3-5) и решим его относительно

$$x_{j+1} = \frac{x_j}{1 - h\lambda}. \quad (4.3-12)$$

Для того чтобы определить устойчивость решения  $x_{j+1}$  при начальном значении  $x_0$ , в уравнении (4.3-12) примем  $j=0$  и  $j=1$ , тогда

$$\begin{aligned}x_1 &= \frac{x_0}{1 - h\lambda}, \\x_2 &= \frac{x_1}{1 - h\lambda} = \frac{x_0}{(1 - h\lambda)^2}.\end{aligned}$$

Отсюда

$$x_j = \frac{x_0}{(1 - h\lambda)^j}. \quad (4.3-13)$$

Метод устойчив, если при  $j \rightarrow \infty$  решение стремится к нулю. Таким образом, из (4.3-13) получаем условие стабильности неявного метода Эйлера в виде

$$|1 - h\lambda| > 1. \quad (4.3-14)$$

Все значения произведения  $h\lambda = h(\alpha + j\beta)$ , для которых условие устойчивости выполняется, образуют область стабильности. На рис. 4.5 изображена эта область при решении дифференциального уравнения (4.3-1) как явным, так и неявным методами Эйлера. Очевидно, что неявный метод для  $\alpha < 0$  устойчив для любых значений

шага  $h$ . Явный же метод устойчив только для небольших значений шага. В качестве примера решим уравнение (4.3-1) при  $x_0=1$  и  $\lambda=0,5 \text{ с}^{-1}$  ( $\alpha=-0,5 \text{ с}^{-1}$ ,  $\beta=0$ ), а также  $h=5 \text{ с}$  (рис. 4.6). Поскольку  $h\lambda=2,5$ , явный метод Эйлера неустойчив.

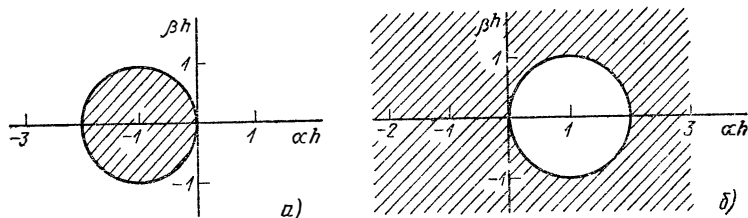


Рис. 4.5. Область стабильности (заштрихована) явного (а) и неявного (б) методов Эйлера при решении дифференциального уравнения (4.3-1)

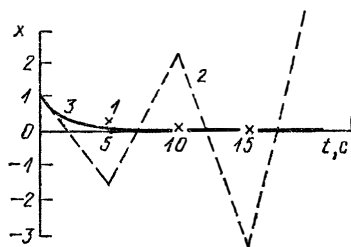


Рис. 4.6. Сравнение неявного 1 и явного 2 методов Эйлера ( $h\lambda=-2,5$ ) с точным решением 3

#### 4.3.2. Интегрирование на уровне двухполюсников.

Метод Эйлера можно использовать и для цепей, содержащих емкости и индуктивности. Для расчета необходимы уравнения связи между последующими  $u_{j+1}$ ,  $i_{j+1}$  и предыдущими  $u_j$ ,  $i_j$  значениями токов и напряжений.

Как уже упоминалось, параметры с индексом  $j+1$  являются искомыми в момент времени  $t_{j+1}$ , тогда как параметры с индексом  $j$  уже известны в этот момент.

Описанная последовательность действий называется интегрированием на уровне двухполюсников. В дальнейшем будем пользоваться исключительно неявным методом Эйлера ввиду его стабильности даже при больших значениях шага.

В первую очередь составим уравнения связи для емкости с учетом уравнения двухполюсника  $i=Cdu/dt$



(при  $C = \text{const}$ ) и рассчитаем в соответствии с (4.3-2) ток, протекающий в момент времени

$$i_{j+1} = C \frac{u_{j+1} - u_j}{h} = \frac{C}{h} u_{j+1} - \frac{C}{h} u_j. \quad (4.3-15)$$

Первый член уравнения характеризует ток в проводимости  $G = C/h$  при напряжении  $u_{j+1}$ . Второй член уравнения можно интерпретировать как собственный ток идеального источника тока. Схема замещения для уравнения (4.3-5) приведена на рис. 4.7.

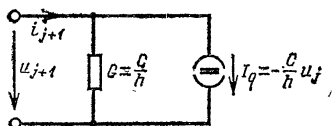


Рис. 4.7. Дискретная токовая модель емкости в неявном методе Эйлера

В дальнейшем будем называть эту схему дискретной моделью токового контура (discrete circuit model) [6].

При анализе электрических цепей с емкостями методом узловых потенциалов необходимо в ходе решения узловых уравнений вычислить напряжение ветви  $u_{j+1}$  в момент  $t_{j+1}$ . Затем определяется ток источника  $I_q$  для следующего шага вычислений.

Теперь сформируем дискретную модель токового контура для индуктивности  $L = \text{const}$ . С учетом уравнения двухполюсника  $u = L di/dt$  в момент времени  $t_{j+1}$  напряжение на нем

$$u_{j+1} = L \frac{i_{j+1} - i_j}{h} = \frac{L}{h} i_{j+1} - \frac{L}{h} i_j. \quad (4.3-16)$$

Первый член в правой части уравнения характеризует падение напряжения, вызванное током  $i_{j+1}$  в активном сопротивлении  $R = L/h$ , а второй член может быть интерпретирован как собственное напряжение идеального источника напряжения (рис. 4.8). Чтобы использовать метод узловых потенциалов для расчета схемы на рис. 4.8, заменим линейный источник напряжения линейным источником тока (рис. 4.9), у которого

$$G = \frac{1}{R} = \frac{h}{L}, \quad (4.3-17)$$

$$I_p = \frac{-u_q}{R} = i_j. \quad (4.3-18)$$

Если методом узловых потенциалов анализируется цепь с индуктивностью, то после вычисления узлового напряжения в момент  $t_{j+1}$  определяется напряжение в ветви  $u_{j+1}$ , а затем ток

$$i_{j+1} = \frac{h}{L} u_{j+1} + i_j, \quad (4.3-19)$$

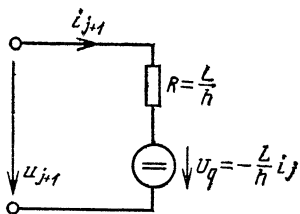


Рис. 4.8. Схема замещения индуктивности в соответствии с уравнением (4.3-16)

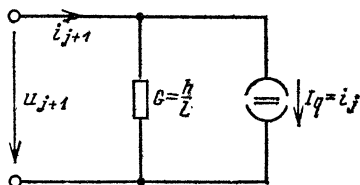


Рис. 4.9. Дискретная токовая модель для индуктивности, используемая в неявном методе Эйлера

являющийся собственным током источника для дальнейших шагов вычислений.

**4.3.3. Локальные и глобальные ошибки.** В методе Эйлера каждый последующий шаг строится на основании результатов предыдущего, причем на каждом шаге возникают как ошибки метода, так и ошибки округления.

Для описываемого метода различают два типа ошибок: ошибки отдельного шага вычислений (локальные) и общие ошибки в интервале  $0 \leq t \leq t_E$  (глобальные).

Дискретные методы анализа переходных процессов недостаточно точны из-за небольших локальных ошибок, возникающих на каждом шаге расчета. Исследуем проблему ошибок на примере схемы, показанной на рис. 4.2, причем выберем  $U_E = 1$  В,  $R_E = 1$  Ом,  $C = 1$  Ф. Постоянная времени для этого случая  $\tau = 1$  с. Допустим, что емкость в момент  $t = 0$  не заряжена, и определим напряжение  $u_{C1}$  на ней в момент  $t_1 = 1$  с. Уравнение (4.2-1) позволяет с высокой точностью вычислить значение искомого напряжения, если  $U_A = 0$ . В результате имеем  $U_{C1} = 0,63212055882856$  В.

В схеме, изображенной на рис. 4.2, заменим линейный источник напряжения линейным источником тока

( $I_E = -1$  А,  $G_E = 1$  См), а емкость — дискретной моделью токового контура в соответствии с рис. 4.7. Для  $t \geq 0$  ключ  $S$  замкнут (рис. 4.10).

Рассчитаем напряжение  $U = u_{j+1}$  с помощью программы EULER-TEST на языке БЕЙСИК, в которой

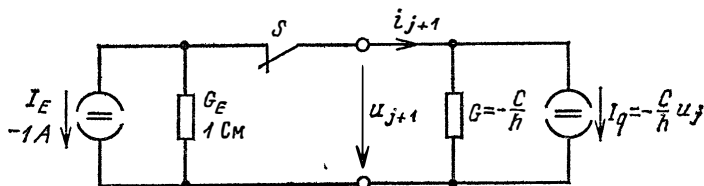


Рис. 4.10. Схема замещения при включении емкости на рис. 4.2

для каждого момента времени поделим отрицательную сумму токов источников на сумму проводимостей (строка 240). Для следующего момента времени это напряжение представляет собой начальное значение  $U = u_j$ .

```

100 PRINT «EULER-TEST»
110 DEFDBL C,G,H,I,T,U
120 DEFINT J,K
130 IE=-1
140 GE=1
150 C=1
160 INPUT «h=»;H
170 K=1/H+.1
180 U=0
190 G=C/H
200 T=0
210 FOR J=1 TO K
220 T=T+H
230 IQ=-G * U
240 U=-(IE+IQ)/(G+GE)
250 NEXT J
260 PRINT «t=»;T;«s; u=»;U;«V»
270 GO TO 160
280 END

```

Запустим программу и зададим несколько значений шага:

EULER=TEST

h=.1

t=.1 c; u=.6144567105704682 В

h=.01

t=.9999999999999998 c; u=.6302887876708808 В

h=.001

t=1.0000000000000001 c; u=.6319366957112229 В

h=.0001

t=1.0000000000000002c; u=.632102165228763 В

Полученные результаты разочаровывают: при  $10^4$  вычислительных шагах значение напряжения  $U_{C1}$  лишь

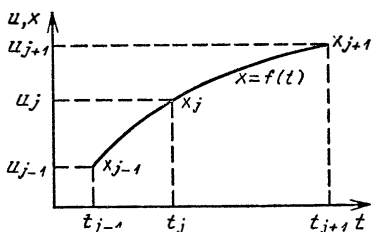
на четыре десятичных знака точнее. Однако глобальная ошибка с уменьшением шага становится меньше. Это объясняется тем, что при более точном вычислении производных ошибка метода уменьшается.

Вместе с тем значительно уменьшать значение шага не имеет смысла по двум причинам: время вычислений становится неоправданно большим и точность результата при большом количестве шагов в конце концов снижается из-за ошибок округления.

## 4.4. Метод обратного дифференцирования

**4.4.1. Обратное дифференцирование.** Увеличение точности расчетов в методе Эйлера может быть получено в том случае, если в процессе вычислений использовать решения для нескольких рассчитанных ранее моментов времени.

Для  $k$  точек, соответствующих этим моментам времени, и искомой точки  $(t_{j+1}; x_{j+1})$  строится интерполяционный полином  $k$ -го порядка, посредством которого описывается кривая  $u=f(t)$ . На рис. 4.11 показан этот полином при  $k=2$ . Для  $k$  точек  $u_j, u_{j-1}, \dots$  вычисленная кривая совпадает с интерполяционным полиномом.



В конце интервала интерполяции, в точке  $(t_{j+1}, x_{j+1})$ , наклон кривой  $x=f(t)$  вычисляется с помощью формулы обратного дифференцирования. Сокращенное название этого метода — BDF образуется от английского *backward differentiation formula*<sup>1</sup>. При больших значениях шага должна быть задана стабильность метода.

Рис. 4.11. Интерполяция кривой  $u=f(t)$  с помощью полинома  $x=f(t)$

Выведем формулу обратного дифференцирования в случае  $k=2$ . Степень интерполяционного полинома  $k$  будем называть порядком BDF-метода.

Для производной  $x'_{j+1}$  от  $x=f(t)$  в момент времени

<sup>1</sup> Формула обратного дифференцирования. — Прим. ред.

$t_{j+1}$  положим

$$x'_{j+1} = -\frac{1}{h} [\alpha_0 x_{j+1} + \alpha_1 x_j + \alpha_2 x_{j-1}]. \quad (4.4-1)$$

При этом на каждом интервале шаг

$$h = t_{j+1} - t_j = t_j - t_{j-1}. \quad (4.4-2)$$

Вычислим теперь коэффициенты  $\alpha_0 \dots \alpha_k$ , для чего упорядочим интерполяционный полином  $x(t)$  второго порядка по степеням и представим его следующим образом:

$$x(t) = \beta_0 + \beta_1 \frac{t_{j+1} - t}{h} + \beta_2 \left( \frac{t_{j+1} - t}{h} \right)^2. \quad (4.4-3)$$

В момент  $t_{j+1}$  производная этого уравнения

$$x'_{j+1} = -\frac{\beta_1}{h}. \quad (4.4-4)$$

Теперь согласно (4.4-3) вычислим функцию  $x(t)$  для нескольких моментов времени:

если  $t = t_{j+1}$ , то  $x_{j+1} = \beta_0$ ,

если  $t = t_j$ , то  $x_j = \beta_0 + \beta_1 + \beta_2$ ,

если  $t = t_{j-1}$ , то  $x_{j-1} = \beta_0 + 2\beta_1 + 4\beta_2$ .

Эти значения, как и  $x'_{j+1}$ , подставим в (4.4-1) и запишем

$$-\frac{\beta_1}{h} = -\frac{1}{h} [\alpha_0 \beta_0 + \alpha_1 (\beta_0 + \beta_1 + \beta_2) + \alpha_2 (\beta_0 + 2\beta_1 + 4\beta_2)].$$

После преобразований

$$\beta_1 = (\alpha_0 + \alpha_1 + \alpha_2) \beta_0 + (\alpha_1 + 2\alpha_2) \beta_1 + (\alpha_1 + 4\alpha_2) \beta_2.$$

Искомые коэффициенты  $\alpha$  теперь могут быть определены из уравнений

$$\left. \begin{aligned} \alpha_0 + \alpha_1 + \alpha_2 &= 0, \\ \alpha_1 + 2\alpha_2 &= 1, \\ \alpha_1 + 4\alpha_2 &= 0. \end{aligned} \right\} \quad (4.4-5)$$

Представим эту систему линейных уравнений в матричной форме:

$$\begin{bmatrix} 1 & 1 & 1 \\ 0 & 1 & 2 \\ 0 & 1 & 4 \end{bmatrix} \cdot \begin{bmatrix} \alpha_0 \\ \alpha_1 \\ \alpha_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}. \quad (4.4-6)$$

В качестве решения получаем

$$\alpha_0 = -1,5; \quad \alpha_1 = 2; \quad \alpha_3 = -0,5.$$

Для любого значения порядка  $k$  уравнение (4.4-1) в общем случае можно представить следующим образом:

$$x'_{j+1} = -\frac{1}{h} \sum_{i=0}^k \alpha_i x_{j+1-i}. \quad (4.4-7)$$

Коэффициенты  $\alpha_0 \dots \alpha_k$  являются решениями системы  $k+1$  линейных уравнений

$$\begin{array}{|c|c|c|c|c|} \hline 1 & 1 & 1 & \dots & 1 \\ \hline 0 & 1 & 2 & \dots & k \\ \hline 0 & 1 & 4 & \dots & k^2 \\ \hline \vdots & & & & \\ \hline 0 & 1 & 2^k & \dots & k^k \\ \hline \end{array} = \begin{array}{|c|c|} \hline \alpha_0 & 0 \\ \hline \alpha_1 & 1 \\ \hline \alpha_2 & 0 \\ \hline \vdots & \vdots \\ \hline \alpha_k & 0 \\ \hline \end{array}. \quad (4.4-8)$$

Ниже приведены значения коэффициентов  $\alpha$  для BDF-метода при различных  $k$  (протокол расчета):

|                                         |                      |
|-----------------------------------------|----------------------|
| $k=1$                                   | $k=5$                |
| ALPHA(0) = -1                           | ALPHA(0) = -2.283333 |
| ALPHA(1) = 1                            | ALPHA(1) = 5         |
|                                         | ALPHA(2) = -5        |
|                                         | ALPHA(3) = 3.333333  |
|                                         | ALPHA(4) = -1,25     |
|                                         | ALPHA(5) = 2         |
| $k=2$                                   | $k=6$                |
| ALPHA(0) = -1,5                         | ALPHA(0) = -2,45     |
| ALPHA(1) = 2                            | ALPHA(1) = 2         |
| ALPHA(2) = -5                           | ALPHA(2) = -7,5      |
|                                         | ALPHA(3) = 6,666667  |
|                                         | ALPHA(4) = -3,75     |
|                                         | ALPHA(5) = 1,2       |
|                                         | ALPHA(6) = -1.666667 |
| $k=3$                                   |                      |
| ALPHA(0) = -1,833333                    |                      |
| ALPHA(1) = 3                            |                      |
| ALPHA(2) = -1,5                         |                      |
| ALPHA(3) = .3333334                     |                      |
| $k=4$                                   |                      |
| ALPHA(0) = -2,083333                    |                      |
| ALPHA(1) = 4                            |                      |
| ALPHA(2) = -3                           |                      |
| ALPHA(3) = 1, 333333                    |                      |
| ALPHA(4) = -.25                         |                      |
| KE=2; t=1,0122 c; U1=6365993552860772 B |                      |
| KE=3; t=1,0004 c; U1=6322674909757657 B |                      |
| KE=4; t=1,0014 c; U1=632635212157708 B  |                      |
| KE=5; t=1,0024 c; U1=6330023915579264 B |                      |
| KE=6; t=1,0034 c; U1=6333692057476123 B |                      |

**4.4.2. Вычисление коэффициентов.** Для каждого значения  $k$  необходимо вычислить соответствующие коэффициенты  $\alpha_i$ . Однако для этого нет необходимости со-

ставлять и решать систему уравнения (4.4-8), потому что их можно определить значительно проще.

Чтобы показать это, запишем сначала в табл. 4.1 коэффициенты  $\alpha_i$  для  $i \geq 1$  и нескольких значений  $k$  в виде дроби.

Таблица 4.1. Коэффициенты  $\alpha$  в виде дробей

| $k$ | 1   | 2    | 3    | 4    | 5     | 6     |
|-----|-----|------|------|------|-------|-------|
| 1   | 1/1 | 2/1  | 3/1  | 4/1  | 5/1   | 6/1   |
| 2   | —   | —1/2 | —3/2 | —6/2 | —10/2 | —15/2 |
| 3   | —   | —    | 1/3  | 4/3  | 10/3  | 20/3  |
| 4   | —   | —    | —    | —1/4 | —5/4  | —15/4 |
| 5   | —   | —    | —    | —    | 1/5   | 6/5   |
| 6   | —   | —    | —    | —    | —     | —1/6  |

Таблица отображает следующие свойства этих коэффициентов:

все  $\alpha_i$  с нечетным индексом положительны, все  $\alpha_i$  с четным индексом отрицательны;

знаменателем каждого  $\alpha_i$  является собственно индекс  $i$ ;

значения числителей являются числами треугольника Паскаля, которые можно достаточно просто определить как биномиальные коэффициенты  $\binom{k}{i}$  (читай  $k$  над  $i$ ):

$$\binom{k}{i} = \frac{k(k-1)(k-2)\dots(k-i+1)}{1 \cdot 2 \cdot 3 \dots i}. \quad (4.4-9)$$

При определении знака числителей в коэффициентах  $\alpha_i$  (табл. 4.1) условно принято, что множители, меньше  $k$  в числителе (4.4-9), отрицательны.

Обозначим числитель через  $\gamma_i$  и для  $i \leq k$  получим

$$\gamma_i = \frac{k(1-k)(2-k)\dots(i-1-k)}{1 \cdot 2 \cdot 3 \dots i}. \quad (4.4-10)$$

Для  $i \geq 1$  с числителями  $\gamma_i$  коэффициенты

$$\alpha_i = \frac{\gamma_i}{i}. \quad (4.4-11)$$

Коэффициент  $\alpha_0$  в соответствии с первой строкой системы уравнений (4.4-6) является отрицательной суммой  $\alpha_i$  для  $i \geq 1$ .

#### Подпрограмма UPR 4800

а) необходимые переменные вызывающей программы:

KB — значение порядка  $k$  BDF-метода;

б) используемые имена переменных:

I1 — текущая переменная  $i-1$ ;

I2 — текущая переменная  $i$ ;

в) результат:

AL(KB) — коэффициенты  $\alpha_i$  для  $0 \leq i \leq k$ ;

GA(KB) — числитель  $\gamma_i$  для  $1 \leq i \leq k$ .

4800 REM Вычисление коэффициентов

4802 REM

4804 AL(0) = -KB

4806 AL(1) = KB

4808 GA(1) = KB

4810 IF KB < 2 THEN 4824

4812 FOR I2 = 2 TO KB

4814 I1 = I2 - 1

4816 GA(I2) = GA(I1) \* (I1 - KB) / I2

4818 AL(I2) = GA(I2) / I2

4820 AL(0) = AL(0) - AL(I2)

4822 NEXT

4824 RETURN

#### 4.4.3. Интегрирование на уровне двухполюсников.

Неявные уравнения, например (4.4-1), решаются итеративным путем. Для этого существует несколько подходящих методов. В качестве примера можно привести метод корректируемой аппроксимации, когда результирующее значение предварительно определяется с помощью аппроксимирующей формулы, а затем уточняется с учетом корректирующей формулы.

При анализе переходных процессов в двухполюсных цепях с помощью BDF-метода целесообразно использовать интегрирование на уровне двухполюсников, рассмотренное в п. 4.3.2 для метода Эйлера, в качестве одного из способов решения уравнения (4.4-7). Неявный метод Эйлера является вариантом метода обратного дифференцирования порядка  $k=1$ . Подставляя коэффициенты  $\alpha_0$  и  $\alpha_1$  в уравнение (4.4-7), получаем уравнение (4.3-4).

Метод обратного дифференцирования, так же как и неявный метод Эйлера, позволяет для каждой ветви цепи, содержащей емкость  $C$  или индуктивность  $L$ , определить дискретную модель токового контура по мето-



ду узловых потенциалов. Таким образом, анализ переходного процесса в момент времени  $t$  сводится к расчету цепи постоянного тока.

Поясним это утверждение сначала для емкости  $C = \text{const}$ . Заменяя в уравнении (4.4-7) каждое значение  $x$  соответствующим значением напряжения  $u$  и подставив производную  $u'_{j+1}$  в уравнение двухполюсника  $i = Cu'$ , получим

$$i_{j+1} = -\frac{C}{h} \sum_{i=0}^k \alpha_i u_{j+1-i}. \quad (4.4-12)$$

Затем выделяем слагаемое для  $i=0$ :

$$i_{j+1} = -\frac{C}{h} \alpha_0 u_{j+1} - \frac{C}{h} \sum_{i=1}^k \alpha_i u_{j+1-i}. \quad (4.4-13)$$

Первый член в правой части этого уравнения характеризует ток, протекающий через проводимость

$$G = -\frac{\alpha_0 C}{h}, \quad (4.4-14)$$

которая находится под напряжением  $u_{j+1}$ .

Поскольку  $\alpha_0$  для всех значений  $k$  меньше нуля, проводимость  $G$  согласно (4.4-14) всегда положительна.

Второй член в правой части уравнения (4.4-13) может быть интерпретирован как собственный ток идеального источника тока

$$I_q = -\frac{C}{h} \sum_{i=1}^k \alpha_i u_{j+1-i}. \quad (4.4-15)$$

Таким образом, дискретная модель токового контура емкости  $C = \text{const}$  для метода обратного дифференцирования представляет собой параллельное соединение проводимости  $G$  и идеального источника тока с собственным током  $I_q$  (рис. 4.12).

Если емкость является элементом цепи, которая анализируется методом узловых потенциалов, то после решения узловых уравнений для момента времени  $t_{j+1}$  вычисляем напряжение ветви  $u_{j+1}$ , а затем, решая  $k-1$ -е уравнение ветвей для моментов времени  $t_j, t_{j-1}, \dots$ , определяем ток источника на следующем шаге вычислений.

Сформируем теперь дискретную модель токового контура для индуктивности  $L = \text{const}$ . Заменяя в уравнении (4.4-7) каждое значение  $x$  соответствующим значением тока  $i$  и подставив производную  $i'_{j+1}$  в уравнение двухполюсника  $u = Li'$ , получим

$$u_{j+1} = -\frac{L}{h} \sum_{i=1}^k \alpha_i i_{j+1-i}. \quad (4.4-16)$$

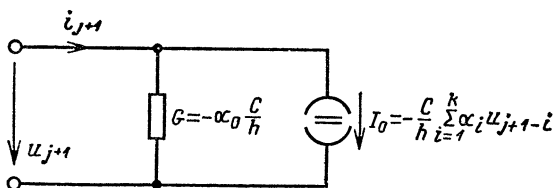


Рис. 4.12. Дискретная токовая модель емкости  $C = \text{const}$  для метода BDF

Затем выделим слагаемое для  $i=0$ :

$$u_{j+1} = -\frac{L}{h} \alpha_0 i_{j+1} - \frac{L}{h} \sum_{i=1}^k \alpha_i i_{j+1-i}. \quad (4.4-17)$$

Первый член в правой части уравнения характеризует падение напряжения на сопротивлении

$$R = -\alpha_0 \frac{L}{h}, \quad (4.4-18)$$

а второй член — собственное напряжение идеального источника напряжения

$$U_q = -\frac{L}{h} \sum_{i=1}^k \alpha_i i_{j+1-i}. \quad (4.4-19)$$

В схеме замещения на рис. 4.13 сопротивление  $R$  и идеальный источник напряжения с собственным значением напряжения  $U_q$  включены последовательно. Для расчета схемы замещения методом узловых потенциалов преобразуем линейный источник напряжения в линейный источник тока.

Проводимость линейного источника тока

$$G = \frac{1}{R} = -\frac{h}{\alpha_0 L}, \quad (4.4-20)$$

а его собственный ток

$$I_q = -\frac{U_q}{R} = -\frac{1}{\alpha_0} \sum_{i=1}^k \alpha_i i_{j+1-i}. \quad (4.4-21)$$

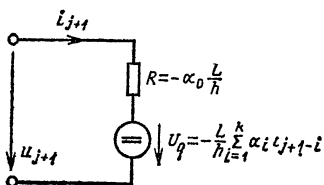


Рис. 4.13. Схема замещения в соответствии с уравнением (4.4-17) для индуктивности

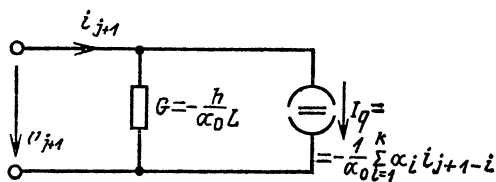


Рис. 4.14. Дискретная токовая модель индуктивности  $L = \text{const}$  для метода BDF

Дискретная модель токового контура индуктивности для метода обратного дифференцирования приведена на рис. 4.14. Если индуктивность является элементом цепи, которая анализируется методом узловых потенциалов, то после решения узловых уравнений в момент времени  $t_{j+1}$  необходимо определить напряжение  $u_{j+1}$  и ток  $i_{j+1}$  ветви, причем

$$i_{j+1} = G u_{j+1} + I_q. \quad (4.4-22)$$

Зная ток ветви, с учетом  $k-1$  уравнения ветвей для предшествующих моментов времени  $t_j, t_{j-1}, \dots$  определяем собственный ток источника на следующем шаге вычислений.

**4.4.4. Начало вычислений.** Метод BDF (обратного дифференцирования) для  $k > 1$  является многошаговым, потому что каждый раз «опирается» на результаты нескольких предыдущих шагов. Неявный метод Эйлера, представляющий собой частный случай метода BDF при  $k=1$ , называется одношаговым, так как расчет заканчивается после первого шага.

Многошаговые методы требуют начального приближения. В соответствии с этим метод BDF не может быть использован при любом значении порядка  $k$ : в начале процесса вычислений необходимо неоднократно повышать на единицу значение порядка системы.

Естественно, возникает вопрос, какое максимальное значение  $k$  нужно выбрать. Необходимо учитывать, что  $k$  является степенью интерполяционного полинома, с помощью которого получают предварительный результат расчета.

График полинома высокой степени, проходящий через заданные точки, может иметь значительные отклонения, прежде всего на границах диапазона вычислений, от интерполирующей кривой (см. § 5.1). С возрастанием порядка гораздо сложнее согласовывать наклон кривой  $u(t)$  с наклоном интерполяционного полинома.

В качестве максимального значения порядка целесообразно принять  $k=6$ . Полученные результаты показывают, что при  $k=6$  еще можно достичь повышения точности расчета по сравнению со случаем, когда  $k=5$ . По поводу устойчивости метода нет единого мнения. Увеличение порядка  $k$  полинома или шага  $h$  приводит к неустойчивости метода (решение не сходится). Из соображений устойчивости  $k=6$  является наибольшим допустимым значением порядка.

Для каждого значения  $k$  или  $h$  необходимо выполнить несколько шагов вычислений  $m$ .

На первом шаге, когда  $k=1$ , значение шага  $h$  выбирается небольшим. За счет этого достигается необходимая точность неявного метода Эйлера. Каждый раз после  $m$  шагов при  $h_{\min}$  на единицу увеличивается значение  $k$  до тех пор, пока не достигается его максимальное значение. Лишь после этого шаг увеличивается. Чтобы снизить нестабильность, его значение удваивается, пока не удовлетворяется условие  $h=h_{\max}=h_E$ .

Однако удвоение шага также может привести к неустойчивости, если узловые потенциалы и, следовательно, параметры ветви значительно изменяются. Поэтому последнее допускается только при незначительных изменениях узловых потенциалов.

Область устойчивости метода BDF при увеличении шага определяется с помощью дополнительных величин  $D_{j+1}$  и  $S_{j+1}$ .

Если выполняется условие  $k_E < h_E$ , то для каждого

момента времени  $t_{j+1}$  сумма всех  $n$  узловых потенциалов

$$S_{j+1} = \sum_{i=1}^n |u_{i+1}|. \quad (4.4-23)$$

Кроме того, разность узловых потенциалов в моменты времени  $t_{j+1}$  и  $t_j$

$$D_{j+1} = \sum_{i=1}^n |u_{i+1} - u_j|. \quad (4.4-24)$$

Значения  $S_{j+1}$  и  $D_{j+1}$  через  $m$  шагов складываются.

Шаг  $h$  увеличивается только в том случае, когда выполняется условие

$$\sum_{j=1}^m D_{j+1} < \varepsilon \sum_{j=1}^m S_{j+1}. \quad (4.4-25)$$

При выполнении вычислений методом BDF для заданных значений параметров элементов необходимо помнить о том, что чем больше мы приближаемся к зоне неустойчивости, тем меньше должно выбираться значение  $\varepsilon$ .

#### **4.5. Программа DYNA-LIN на языке БЕЙСИК**

Программа используется для анализа переходных процессов в линейной двухполюсной цепи методом BDF.

**4.5.1. Ввод структуры цепи.** Поскольку надо предусмотреть возможность изменить параметры цепи после проведения анализа переходного процесса, ввод элементов одной ветви  $I$  целесообразно оформить в виде подпрограммы (UPR 4500). После ввода номеров узлов для каждой нагрузочной ветви задается конфигурация цепи. Нагрузочная ветвь линейной двухполюсной цепи может содержать либо проводимость  $G(I) = \text{const}$ , либо  $C(I)$ , либо  $L(I)$ . К проводимости  $G(I)$  может быть параллельно подключен идеальный или управляемый источник тока. Вид ветви определяется с помощью ключевого символа (см. п. 3.3.1). К сожалению, ключевые символы в БЕЙСИКе не запоминаются, поэтому вид ветви запоминается посредством числа, размещаемого в ячейке  $IA(I)$  массива  $IA(IR)$  (табл. 4.2).

Если для ветви  $I$  задается сопротивление, то его соб-

**Т а б л и ц а 4.2. Ключевые буквы и численные значения, определяющие тип нагрузочной ветви**

| Вид ветви и ключевая буква | Численное значение |
|----------------------------|--------------------|
| Проводимость $G$           | IA (I) = 1         |
| Сопротивление $R$          | IA (I) = 2         |
| Емкость $C$                | IA (I) = 3         |
| Индуктивность $L$          | IA (I) = 4         |

ственное значение запоминается в элементе  $G(I)$  массива проводимостей. Поэтому в программе DYNA-LIN значение  $IA(I)=2$ , соответствующее сопротивлению, запоминать в качестве определителя вида ветви нет необходимости. Для нагрузочной ветви используется только один из трех массивов  $G(I)$ ,  $L(I)$  или  $C(I)$ , и для экономии памяти можно было бы запоминать параметры двухполюсников  $G$ ,  $C$  или  $L$  в одном массиве.

Для каждого реактивного элемента  $C(I)$  или  $L(I)$  дополнительно к параметрам двухполюсников необходимо вводить начальные значения, которые запоминаются в массиве  $ZG(I, 1)$ . Начальным значением для емкости является напряжение ветви, для индуктивности — ток ветви в момент времени  $t=0$ .

Управляемый источник тока задается с помощью программы UPR 2400. Ввод идеального источника напряжения выполняется так же, как в подпрограмме UPR 2500. Ветви с идеальными источниками напряжения должны поэтому иметь наибольшие номера.

**Подпрограмма UPR 4500** (она вызывает подпрограммы UPR 2400 и 4600)

а) необходимые переменные вызывающей программы:

I — номер ветви;

IR — количество нагрузочных ветвей;

б) используемые имена переменных:

IL — номер идеального источника напряжения;

J, K — номера узлов;

RI — сопротивление;

в) результаты:

для  $I \leq IR$ :

$C(I)$ ;  $G(I)$ ;  $IA(I)$ ;  $IG$ ;  $IT$ ;  $(I)$ ;  $KV(I)$ ;  $KZ(I)$ ;  $L(I)$ ;  $MZ(I)$ ;  $SF(I)$ ;  $SR(I)$ ;  $VF(I)$ ;  $ZF(I)$ ;  $ZG(I, 1)$ ...

(подробности см. в приложении ПЗ);

для  $I > IR$ :

$IT(I)$ ;  $KA(I)$ ;  $KU(I)$ ;  $QR(I)$ ;  $VF(I)$ ;  $ZF(I)$ .

```

4500 REM: Ввод элементов цепи
4502 REM
4504 INPUT «из узла»; J
4506 INPUT «к узлу»; K
4508 IF >DIR THEN 4580
4510 KV(I) = J
4512 KZ(I) = K
4514 G(I) = Ø
4516 L(I) = Ø
4518 C(I) = Ø
4520 INPUT «Тип ветви»; FR$
4522 IF FR$ = «R» THEN 4532
4524 IF FR$ = «G» THEN 4538
4526 IF FR$ = «L» THEN 4566
4528 IF FR$ = «C» THEN 4572
4530 GOTO 4520
4532 INPUT «Сопротивление»; R1
4534 G(I) = 1/R1
4536 GOTO 4540
4538 INPUT «Проводимость»; G(1)
4540 IA(I) = 1
4542 SR(I) = 0
4544 MZ(I) = 0
4546 INPUT «Тип источника»; FR$
4548 IF FR$ = «I» THEN 4556
4550 IF FR$ = «S» THEN 4562
4552 IF FR$ = «N» THEN 4564
4554 GOTO 4546
4556 INPUT «Ток источника»; SR(I)
4558 GOSUB 4600
4560 RETURN
4562 GOSUB 2400
4564 RETURN
4566 IA(I) = 4
4568 INPUT «Индуктивность»; L(I)
4570 GOTO 4576
4572 IA(I) = 3
4574 INPUT «Емкость»; C(I)
4576 INPUT «Начальное значение»; ZG(I, 1)
4578 RETURN
4580 IL = I - IR
4582 KA(IL) = J
4584 KU(IL) = K
4586 INPUT «Напряжение источника»; QR(IL)
4588 GOSUB 4600
4590 RETURN

```

#### 4.5.2. Временная зависимость параметров источников.

Для динамической цепи параметры идеального источника напряжения или идеального источника тока могут произвольным образом зависеть от времени. Характер зависимости каждого источника фиксируется во время ввода его параметров в ЭВМ с помощью переменной IT(I).

В программе анализа переходных процессов предусмотрено три типа временной зависимости. Если  $IT(I) = 1$ , то параметр источника  $x(t) = x_{\max}$  является постоянным во времени; если  $IT(I) = 2$ , то параметр источника линейно изменяется во времени по формуле

$$x(t) = mt + c \quad (4.5-1)$$

до тех пор, пока не достигнет значения  $x(t) = x_{\max}$ , после чего  $x(t) = x_{\max} = \text{const}$ ; если  $IT(I) = 3$ , то параметр источника меняется во времени по синусоидальному закону  $x(t) = x_{\max} \cos(\omega t + \varphi_x)$  с круговой частотой  $\omega$  и начальным фазовым углом  $\varphi_x$ .

Крутизна  $m$  или круговая частота  $\omega$  запоминаются в массиве  $ZF(I)$ . Отрезок оси или начальный фазовый угол  $\varphi_x$  запоминаются в массиве  $VF(I)$ . Значение  $x_{\max}$  для собственного тока источника запоминается в массиве  $SR(I)$ , а для собственного напряжения источника — в массиве  $QR(I)$ . Для синусоидальных параметров в  $SR(I)$  и  $QR(I)$  хранятся соответствующие амплитудные значения.

Ввод параметров источника, зависящего от времени, описан в подпрограмме UPR 2400. Вид временной зависимости при этом вводится в программу ключевыми символами K (const), L (линейный) или S (синусоидальный).

#### Подпрограмма UPR 4600

- а) необходимых переменных в вызывающей программе нет;
  - б) дополнительные имена переменных не используются;
  - в) результат:
    - $IT(I)$  — тип временной зависимости;
    - $VF(I)$  — начальный коэффициент (отрезок оси или начальный фазовый угол);
    - $ZF(I)$  — временной коэффициент (крутизна или круговая частота).
- ```

4600 REM: Ввод временной зависимости
4602 REM
4604 INPUT «зависимы от времени (K/L/S)»; FR$
4606 IF FR$ = «K» THEN 4614
4608 IF FR$ = «L» THEN 4618
4610 IF FR$ = «S» THEN 4626
4612 GOTO 4604
4614 IT(I) = 1
4616 RETURN
4618 IT(I) = 2
4620 INPUT «Временной коэффициент»; ZF(I)
4622 INPUT «Отрезок оси»; VF(I)
4624 RETURN
4626 IT(I) = 3
4628 INPUT «Круговая частота»; ZF(I)
```



```
4630 INPUT «Начальный фазовый угол»; VF(I)
4632 RETURN
```

Мгновенное значение параметра источника в момент времени t вычисляется в подпрограмме UPR 4400, причем переменная ST выступает в качестве передаваемого параметра: перед вызовом подпрограммы UPR ST равна амплитудному значению QR(IL) или SR(I) параметра источника. После выполнения UPR 4400 переменная ST равна мгновенному значению параметра источника. Система, с помощью которой описывается временная зависимость, является расширяемой, т. е. можно определить, а затем включить в подпрограммы UPR 4400 и 4600 дополнительные виды временной зависимости.

Подпрограмма UPR 4400

а) необходимые переменные вызывающей программы:

I — номер ветви;
 IT(L) — тип временной зависимости;
 ST — максимальное значение,
 T — момент времени t ;
 VF(I) — отрезок оси или начальный фазовый угол;
 ZF(I) — наклон или круговая частота;

б) используемые имена переменных:

S1 — вспомогательная величина;

в) результат:

ST — мгновенные значения параметра источника.

```
4400 REM: Расчет мгновенных значений
4402 REM: параметра источника
4404 REM
4406 IF ST=0 THEN 4424
4408 ON IT(I) GOTO 4410, 4414, 4420
4410 S1=ST
4412 GOTO 4422
4414 S1=ZF(I) * T+VF(I)
4416 IF ABS(S1)>ABS(ST) THEN 4410
4418 GOTO 4422
4420 S1=ST * COS(ZF(I) * T+VF(I))
4422 ST=S1
4424 RETURN
```

4.5.3. Расчет дискретной токовой модели. Для заполнения матрицы узловых проводимостей требуется некоторая «предварительная подготовка», реализуемая подпрограммой UPR 4000. Сначала вычисляется мгновенное значение QT(IL) напряжения идеальных источников ЭДС для момента времени t (строки 4008—4018). Затем для каждой ветви I определяется, какие идеальные источники напряжения необходимо разместить в данной ветви: вызов подпрограммы UPR 2600 иницирует заполнение массива VR(IR).

В строках 4022—4058 рассчитывается дискретная токовая модель каждого линейного двухполюсника, которая состоит из проводимости $GD(I)$ и параллельно подключенного к ней идеального источника тока с собственным током $SD(I)$ (рис. 4.15). Массивы $GD(I)$ и $SD(I)$ используются не только для запоминания матрицы узловых проводимостей, но и для расчета тока ветви $SA(I)$.

Если ветвь I содержит проводимость, то $IA(I)=1$. Проводимость $GD(I)$ дискретной токовой модели совпадает с проводимостью $G(I)$ ветви. Ток источника $SD(I)$ является мгновенным значением собственного тока источника ветви I .

Если элемент ветви I является реактивным элементом, то проводимость $GD(I)$ дискретной токовой модели для емкости рассчитывается согласно уравнению (4.4-14), а для индуктивности — согласно уравнению (4.4-20).

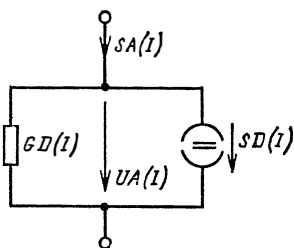


Рис. 4.15. Дискретная токовая модель для ветви I

Ток источника $SD(I)$ дискретной токовой модели для емкости и индуктивности определяется с помощью параметров состояния, которые для k моментов времени запоминаются в массиве $ZG(I, K)$. Здесь k характеризует порядок метода BDF. Например, при анализе переходного процесса в момент времени t_{j+1} в ячейке $ZG(I, 1)$ массива содержится параметр состояния ветви I в момент t_j , а в ячейке $ZG(I, 2)$ — в момент t_{j-1} .

Параметром состояния для емкости является напряжение ветви, а для индуктивности — ток ветви. В соответствии с уравнениями (4.4-15) или (4.4-21) произведения коэффициентов α_i и значений параметров состояний суммируются (строки 4042—4048).

Затем с помощью суммы $S1$ вычисляется для конкретного вида накопителя энергии ток источника $SD(I)$ в дискретной токовой модели.

Подпрограмма UPR 4000 (в ней вызываются UPR 2600 и 4400)

а) необходимые переменные вызывающей программы:

AL(KB); C(I); G(I); H; IA(I); IR; IS; IT(I); KA(IS); KB; KV(I); KZ(I); L(I); NZ; QR(IS); SR(I); T; VF(I); ZF(I);

ZG(I, 11)... (подробности см. в приложении ПЗ);
 б) используемые имена переменных: I; IL; J; S1; ST;
 в) результат:
 GD(IR); QT(IS); SD(IR); VR(IR).
 4000 REM: Расчет дискретной токовой модели
 4002 REM: для каждого линейного двухполюсника
 4004 REM
 4006 IF IS=0 THEN 4020
 4008 FOR IL=1 TO IS
 4010 I=IL+IR
 4012 ST=QR(IL)
 4014 GOSUB 4400
 4016 QT(L)=ST
 4018 NEXT
 4020 COSUB 2600
 4022 FOR I=NZ+1 TO IR
 4024 ON IA(I) GOTO 4026, 4026, 4036, 4040
 4026 GD(I)=G(I)
 4028 ST=SR(I)
 4030 GOSUB 4400
 4032 SD(I)=ST
 4034 GOTO 4058
 4036 GD(I)=-AL(0) * C(I)/H
 4038 GOTO 4042
 4040 GD(I)=-H/(AL(0) * L(I))
 4042 S1=0
 4044 FOR J=1 IO KB
 4046 S1=S1+AL(J) * ZG(I, J)
 4048 NEXT J
 4050 ON IA(I) GOTO 4058, 4058, 4052, 4056
 4052 SD(I)=-C(I) * S1/H
 4054 GOTO 4058
 4056 SD(I)=-S1/AL(0)
 4058 NEXT I
 4060 RETURN

Подпрограмма UPR 4000 может быть использована и для линейных двухполюсников. В программе DYNA-LIN устанавливается в (строке 124) NZ=0.

4.5.4. Вычисление параметров ветви. Для определения параметра D_{j+1} по уравнению (4.4-24) используются потенциалы узлов как в момент времени t_{j+1} , так и в момент времени t_j . Узловые потенциалы для t_j при $j>1$ предварительно размещаются в массиве UN(KN) и запоминаются в массиве UV(KN), а для t_{j+1} вычисляются в подпрограммах UPR 2700 и 2800. Параметры S_{j+1} согласно уравнению (4.4-23) и D_{j+1} , которые в программе обозначаются соответственно SX и DX, не определяются, если управляющий параметр JF=0 (начальные условия).

Вычисление напряжений ветвей UA(I) и токов ветвей SA(I) выполняется в подпрограмме UPR 2900.

Подпрограмма UPR 4100 (она вызывает подпрограммы UPR 2700, 2800, 2900).

а) необходимые переменные вызывающей программы:

KN; UN(KN)...; кроме того, см. п. а) в подпрограммах UPR 2700, 2800, 2900;

б) используемые имена переменных:

UV(KN)...; кроме того, см. п. в) в подпрограммах UPR 2700, 2800, 2900;

в) результат:

DX; SA(IR); SX; UA(IR); UN(KN)

— подробности см. в приложении ПЗ.

4100 REM: Вычисление параметров ветви

4102 REM

4104 FOR I=1 TO KN

4106 UV(I)=UN(I)

4108 NEXT

4110 GOSUB 2700

4112 GOSUB 2800

4114 IF JF=0 THEN 4124

4116 FOR I=1 TO KN

4118 DX=DX+ABS(UN(I)-UV(I))

4120 SX=SX+ABS(UN(I))

4122 NEXT

4124 GOSUB 2900

4126 RETURN

4.5.5. Запоминание параметров состояний. После расчета параметров каждой ветви I параметры состояний запоминаются в строках 4306—4310 подпрограммы, а предыдущие состояния сдвигаются, как в сдвиговом регистре, на одну ячейку. Затем новое значение размещается в ячейке массива ZG(I, 1) (строки 4312—4318). В подпрограмме UPR 4300 заданные параметры нагрузочных ветвей для момента времени t_{j+1} выводятся на периферийное устройство.

Подпрограмма UPR 4300

а) необходимые переменные в вызывающей программе:

IA(IR); IR; MA; MB; MI(IR); MU(IR); SA(IR); T; UA(IR); ZG(IR, 11);

б) используемые имена переменных: I, J;

в) результат: ZG(IR, 11); вывод параметров ветви.

4300 REM: Запоминание параметров состояния

4302 REM

4304 FOR I=1 TO IR

4306 FOR J=11 TO 2 STEP-1

4308 ZG(I, J)=ZG(I, J-1)

4310 NEXT J

4312 ON IA(I) GOTO 4314, 4318, 4314, 4318

4314 ZG(I, 1)=UA(I)

4316 GOTO 4320

4318 ZG(I, 1)=SA(I)

4320 NEXT I

4322 PRINT

```

4324 PRINT «t=»; T; «s»
4326 IF MA<1 THEN 4338
4328 FOR I=1 TO MA
4330 J=MU(I)
4332 IF J>IR THEN 4336
4334 PRINT «U»;J;«=»;UA(J);«V»
4336 NEXT
4338 IF MB<1 THEN 4350
4340 FOR I=1 TO MB
4342 J=MI(I)
4344 IF J>IR THEN 4348
4346 PRINT «I»;J;«=»;SA(J);«A»
4348 NEXT
4350 RETURN

```

4.5.6. Анализ переходных процессов. После ввода структуры цепи и определения выходных параметров в ЭВМ вводятся управляющие параметры НА, НЕ, КЕ, ТЕ, Е0. В методе Эйлера выполнение программы начинается при $k=1$ с шагом НА. Первоначально значение k увеличивается пошагово на единицу. Если достигается максимальное значение порядка интерполяционного полинома КЕ, то шаг Н удваивается и становится равным НЕ. Поэтому вычисления прекращаются не в момент времени ТЕ, а в некоторый момент $T > TE$. Целесообразно выбирать $KE=5$ или $KE=6$.

Выбор начальных значений остальных управляющих величин требует некоторого опыта. В процессе анализа переходных процессов пользователь программ после нескольких попыток получает представление о том, как находить подходящие значения. Если значения управляющих параметров НА и НЕ выбраны очень маленькими, то анализ продолжается слишком долго. При больших значениях управляющих параметров может быть потеряна информация об изменении во времени параметров цепи и, кроме того, не исключено нестабильное решение.

Каждый расчет при анализе переходного процесса должен проверяться на устойчивость решения. Для этого целесообразно оценивать значения параметров при больших временах.

Если все параметры источников не зависят от времени, то подобная проверка осуществляется с помощью вычислений их стационарных конечных значений по программе GLEICH, причем каждая индуктивность должна быть заменена короткозамкнутой ветвью, а каждая емкость — разрывным соединением [12]. Если же все параметры источников являются синусоидальными и их час-

тоты одинаковы, то установившееся состояние можно рассчитать с помощью программы SINUS.

Выполнение анализа цепи для определенного момента времени t осуществляется посредством обращения к подпрограммам UPR 4000, 4100, 4300. Разделение на три подпрограммы признано целесообразным, так как все они по отдельности вызываются в программе DYNA-NL (см. гл. 6).

```
100 PRINT «DYNA-LIN»
102 PRINT «Анализ переходных процессов»
104 PRINT «линейной динамической цепи из двухполюсников»
106 PRINT
108 DEFINT I—K, M—N
110 DEFDBL A—D, G, H, L, P—Z
112 REM
114 REM: Ввод глобальных параметров
116 INPUT «Количество ветвей», IZ
118 INPUT «Количество узлов (без базового)»; KN
120 INPUT «Количество идеальных источников напряж.»; IS
122 PRINT
124 NZ=0
126 N=KN—IS
128 IF IZ<IS THEN 132
130 IF N>0 THEN 138
132 PRINT «Ввод не имеет смысла»
134 PRINT «или обработка невозможна»
136 GOTO 116
138 IR=IZ—IS
140 DIM UN(KN), UV(KN), KF(KN), A(N, N) X(N)
142 DIM SF(IR), MZ(IR), VR(IR), SR(IR)
144 DIM KV(IR), KZ(IR), GD(IR), SD(IR), SA(IR), UA(IR)
146 DIM G(IR), L(IR), C(IR), MI(IR), MU(IR)
148 DIM IA(IR), ZG(IR, 11), AL(6), GA(6)
150 DIM IT(IZ), ZF(IZ), VF(IZ)
152 DIM KA(IS), KU(IS), QR(IS), QT(IS)
154 REM
156 PRINT «Ввод структуры цепи»
158 IG=0
160 FOR I=1 TO IZ
162 PRINT «Ветвь»; I
164 GOSUB 4500
166 NEXT
168 REM
170 REM: Определение массива узлов
172 GOSUB 2100
174 REM: Определение выводимых величин
176 GOSUB 3400
178 REM
180 REM: Ввод параметров управления
182 PRINT
184 INPUT «Начальное значение шага HA»; HA
186 INPUT «Максимальное значение шага HE»; HE
```

```

188 INPUT «Максимальное значение порядка KE»; KE
190 IF KE<2 THEN 196
192 IF KE>THEN 196
194 GOTO 202
196 PRINT «Ввод не имеет смысла»
198 PRINT «повторите, пожалуйста»
200 GOTO 188
202 INPUT «Конечное значение времени TE»; TE
204 INPUT «Значение предела»; EB
206 PRINT
208 REM: При значениях шага меньше H=HA
210 REM: значение порядка повышается от 1 до KE;
212: REM: для каждого значения KB цепь анализируется
214 REM: в MT моментах времени
216 MT=10
218 UN(0)=0
220 T=0
222 H=HA
224 JF=0
226 FOR KB=1 TO KE
228 GOSUB 4800
230 FOR ML=1 TO MT
232 T=T+H
234 GOSUB 4000
236 GOSUB 4100
238 GOSUB 4300
240 NEXT ML
242 NEXT KB
244 KB=KE
246 REM: Для каждого значения MT вычисляется и заданным
248 REM: образом удваивается значение шага
250 JF=1
252 DX=0
254 SX=0
256 FOR ML=1 TO MT
258 T=T+H
260 IF T>(TE+HE) THEN 306
262 GOSUB 4000
264 GOSUB 4100
266 GOSUB 4300
268 NEXT ML
270 IF DX/SX>EB THEN 252
272 H=2 * H
274 FOR J=1 TO 6
276 JD=2 * J-1
278 FOR I=1 TO IR
280 ZG(I, J)=ZG(I, JD)
282 NEXT I
284 NEXT J
286 IF H<HE THEN 252
288 REM: Дальнейшие вычисления выполняются с максималь-
289 REM: ным
290 REM: значением шага до конечного значения времени
292 JF=0
294 T=T+H

```

```

296 GOSUB 4000
298 GOSUB 4100
300 GOSUB 4300
302 IF T<TE THEN 294
304 REM
306 PRINT «Вычисления»
308 INPUT «продолжать (J/N)» FE$
310 IF FR$=«J» THEN 322
312 IF FR$=«N» THEN 9999
314 GOTO 306
316 REM
318 REM: Изменение параметров ветви
320 REM
322 PRINT
324 PRINT «В скольких ветвях должны быть»
326 INPUT «изменены элементы?»; ME
328 IF ME<1 THEN 9999
330 FOR ML=1 TO ME
332 INPUT «Ветвь»; I
334 GOSUB 4500
336 NEXT
338 GOTO 172
1000 REM: UPR 2000, 2100, 2200, 2300, 2400
1002 REM: UPR 2600, 2700, 2800, 2900, 3400
1004 REM: UPR 4000, 4100, 4300, 4400, 4500, 4600, 4800

```

4.5.7. Примеры. В п. 4.3.3 мы проанализировали переходный процесс в схеме на рис. 4.2 методом Эйлера. Теперь выясним для этой схемы, какой точности вычислений можно добиться с помощью метода BDF. Запустим программу DYNA-LIN и введем значения $R_E=1$ Ом; $C=1$ Ф; $U_A=0$; $U_E=1$ В. Затем пошагово увеличим максимальное значение порядка KE от 2 до 6. Остальные управляющие параметры для $KE>2$ получим так же, как для $KE=2$:

начальное значение шага $HA?$ 0.0001;
 максимальное значение шага $HE?$ 0.1;
 максимальное значение порядка $KE?$ 2;
 конечное значение времени $TE?$ 1;
 предел эpsilon? 0.1.

Результаты анализа переходных процессов выглядят следующим образом (машинная распечатка):

```

KE=2;      t=1.0122 c;      U1= 6365993552860772 B
KE=3;      t=1.0004 c;      U1=6322674909757657 B
KE=4;      t=1.0014 c;      U1=632635212157708 B
KE=5;      t=1.0034 c;      U1=6333692057476123 B

```

Сравним эти результаты с точными значениями, полученными из уравнения (4.2-1). Поскольку функцию, например e -функцию, в программе GWBASIC можно вычис-

лить только с обычной точностью, сравнение проведем с простой программой на ПАСКАЛе:

```
PROGRAM bexp; VAR t, u: REAL
BEGIN read (t); u:=1.0—exp(—t);
writeln(Lst, «t=», t:1:5, «с; Uc=», u:1:14, «В»)END
```

В результате получаем значения напряжений u_c из уравнения (4.2-1) (машинная распечатка):

$t=1.01220$ с;	$U_c=0.63658142141962$ В
$t=1.00040$ с;	$U_c=0.63226768117859$ В
$t=1.00140$ с;	$U_c=0.63263522969253$ В
$t=1.00240$ с;	$U_c=0.63300241084166$ В
$t=1.00340$ с;	$U_c=0.63336922499318$ В

Несмотря на то что выполнено менее 100 вычислительных шагов, метод BDF обеспечивает точность до пятого знака для $k=2$ и до восьмого знака для $k=4$. Повышение порядка до $k=6$ нецелесообразно. Это объясняется тем, что интерполяция с помощью полинома при увеличивающейся степени (или увеличивающемся значении порядка) становится проблематичной в силу «колебательного» характера полинома (см. гл. 5).

Поскольку при $k=5$ или $k=6$ увеличивается время вычислений, но не возрастает точность расчета, рекомендуется выбирать в качестве наиболее подходящего значения порядка для метода BDF $k=4$. После 100 вычислительных шагов в (неявном) методе Эйлера мы получили только два достоверных десятичных разряда (см. п. 4.3.3).

В качестве примера переходного процесса в цепи постоянного тока рассчитаем приведенное выходное напряжение u_2/U_q трансформатора без потерь (коэффициент связи $k=0,95$; коэффициент трансформации $\dot{u}=1$), который в момент $t=0$ подключается к линейному источнику напряжения (рис. 4.16). Приведенное выходное на-

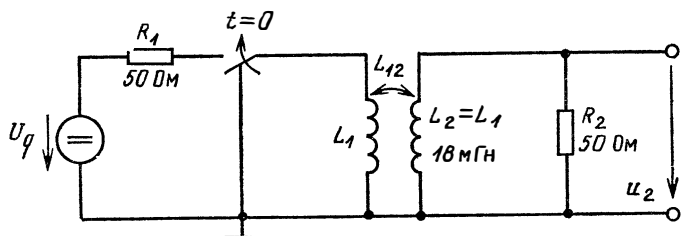


Рис. 4.16. Включение трансформатора в цепь постоянного напряжения

пряжение u_2/U_q равно выходному напряжению u_2 , если $U_q=1$ В. Заменяем трансформатор его схемой замещения (см. рис. 3.3) и рассчитаем индуктивности:

$$L_3 = L_5 = L_1 - L_{12} = 0,9 \text{ мГн},$$

$$L_4 = L_{12} = k \sqrt{L_1 L_2} = 17,1 \text{ мГн}.$$

Затем запустим программу и введем в ЭВМ структуру цепи, приведенную на рис. 4.17.

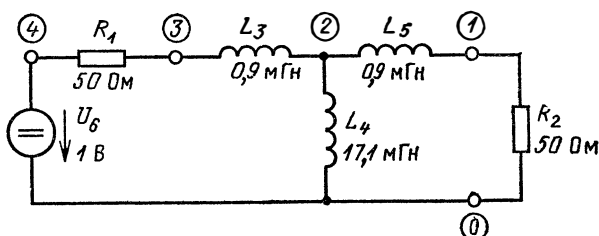


Рис. 4.17. Схема трансформатора ($k=0,95$; $\bar{a}=1$) (рис. 4.16) для $t>0$ со схемой замещения (рис. 3.3)

Анализ переходного процесса линейной динамической цепи из двухполюсников:

Количество ветвей? 6

Количество узлов (без базового)? 4

Количество идеальных источников напряжения? 1

Ввод структуры цепи

Ветвь 1

из узла? 4

к узлу? 3

Тип ветви? R

Сопротивление? 50

Тип источника? N

Ветвь 2

из узла? 1

к узлу? 0

Тип ветви? R

Сопротивление? 50

Тип источника? N

Ветвь 3

из узла? 3

к узлу? 2

Тип ветви? L

Индуктивность? .0009

Начальное значение? 0

Ветвь 4

из узла? 2

к узлу? 0

Тип ветви? L

Индуктивность? .0171
 Начальное значение? 0
 Ветвь 5
 из узла? 2
 к узлу? 1
 Тип ветви? L
 Индуктивность? .009
 Начальное значение? 0
 Ветвь 6
 из узла? 4
 к узлу? 0
 Источник напряжения? 1
 Временная зависимость (K/L/S)? K
 Какие параметры ветви должны быть выведены?
 Количество напряжений ветви? 1
 Номера ветвей
 ? 2
 Количество токов ветви? 0
 Начальное значение шага HA? .000001
 Максимальное значение шага HE? .0001
 Максимальный порядок KE? 5
 Конечное значение времени TE? .001
 Предел эпсилон? 01

Ниже распечатаны несколько пар значений напряжений после включения и несколько пар значений напряжений в области максимума:

t=.000001 c;	U2=.0256045519203414 B
t=.000002 c;	U2=4.982507400715082 D-02 B
t=.000003 c;	U2=7.273446174503484 d-02 B
t=.000004 c;	U2=9.440177394044978 D-02 B
t=.000005 c;	U2=.1148924346522356 B

...и следующие несколько пар значений...

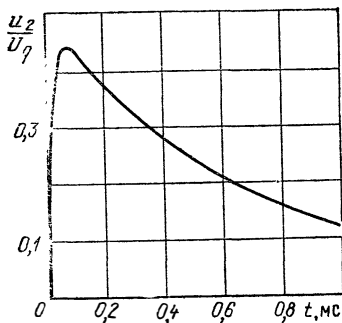


Рис. 4.18. График приведенного выходного напряжения

t=.000062 c;	U2=.4415343698440188 B
t=.000064 c;	U2=.4419362334829114 B
t=.000066 c;	U2=.442162596280349 B
t=.000068 c;	U2=.442232294882046 B
t=.00007 c;	U2=.4421621820194679 B
t=.000072 c;	U2=.4419673363701275 B

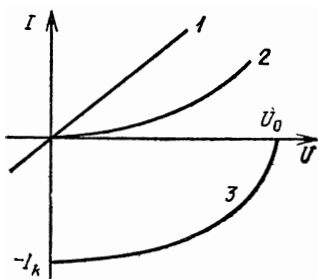
Результаты расчета в виде графика приведены на рис. 4.18.

НЕЛИНЕЙНЫЕ ЦЕПИ ПОСТОЯННОГО НАПРЯЖЕНИЯ

Элементами цепи постоянного напряжения являются сопротивление R или проводимость G , идеальные источники напряжения $U_q = \text{const}$ и тока $I_q = \text{const}$, а также управляемые источники тока, коэффициент S или усиление по току B которых постоянны. Управляемый источник напряжения должен быть перед вводом в ЭВМ преобразован в управляемый источник тока (§ 2.3).

В линейных цепях активные сопротивления R постоянны, а их вольт-амперные характеристики представляют собой прямую линию (рис. 5.1). Рассмотрим теперь

Рис. 5.1. Вольт-амперная характеристика активного (1), нелинейного (2) сопротивлений и нелинейного источника (3)



нелинейные цепи, в которых параметры хотя бы одного двухполюсника зависят от напряжения или тока. Такой двухполюсник будем называть нелинейным.

Нелинейный двухполюсник может быть только пассивным или частично активным. Вольт-амперная характеристика пассивного двухполюсника располагается в I и III квадрантах и всегда проходит через начало координат, активного двухполюсника пересекает ось I при отрицательном значении тока короткого замыкания (т. е. при $-I_k$) и ось U при напряжении холостого хода U_0 .

Если в нагрузочной цепи параллельно активному сопротивлению или проводимости включить идеальный источник тока, то нелинейный двухполюсник не будет содержать идеальных источников.

5.1. Интерполяция

Нелинейный двухполюсник описывается характеристикой (например, вольт-амперной), которая на графике представляет собой кривую. В цифровой форме кривая

задается опорными точками. Остальные точки определяются методом интерполяции.

Самым простым видом интерполяции является линейная интерполяция, когда две соседние опорные точки соединяются друг с другом прямой линией, а промежуточные точки определяются из уравнения прямой. Линейная интерполяция для нелинейной кривой имеет смысл, если опорные точки располагаются достаточно близко друг к другу.

Однако линейная интерполяция в ряде случаев нецелесообразна, поскольку некоторые кривые имеют «гладкий» характер (кривая намагничивания металла). В соответствии с этим требуется, чтобы функция, которую отображает кривая, была на всем интервале непрерывной и дифференцируемой [49]. Для линейной интерполяции подобное требование не выполняется: уже первая производная имеет ступенчатый характер (рис. 5.2).

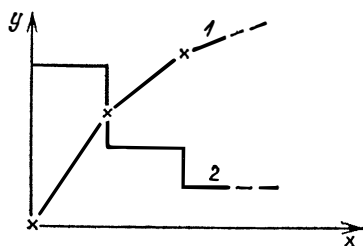


Рис. 5.2. Зависимость $y=f(x)$, линейно интерполированная между двумя опорными точками (1), и ее первая производная y' (2)

Для нелинейной интерполяции не вполне подходят трансцендентные функции ($\sin x$; $\ln x$ и т. д.), так как они требуют больших вычислительных затрат. С этой точки зрения гораздо целесообразнее иметь дело с полиномами, при решении которых используются только основные типы вычислительных операций.

При интерполяции кривой полиномами высоких степеней полиномиальная функция значительно отклоняется от заданной кривой, прежде всего на границах области вычислений. Это объясняется тем, что полином n -й степени имеет $n-1$ экстремальных точек и $n-2$ точек перегиба. Чем выше степень полинома, тем больше количество экстремумов и точек перегибов. Целесообразно выбирать полиномы с невысокими степенями. Ниже приведен полином третьей степени:

$$y = ax^3 + bx^2 + cx + d. \quad (5.1-1)$$

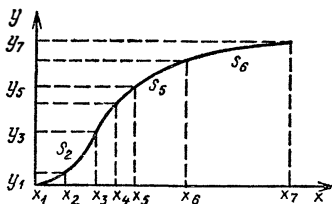
В этом примере используются общепринятые обозначения x и y , поскольку интерполяция применяется для самых различных характеристик.

Разумеется, с помощью полинома третьей степени можно описать не все кривые между сколь угодно большим количеством точек. Гораздо чаще интерполируются кривые в интервалах между двумя соседними опорными точками. Такая интерполяция называется кусочной.

5.1.1. Сплайн-интерполяция. Специальным видом кусочной интерполяции является интерполяция с помощью сплайн-функций. Образованные в процессе такой интерполяции кривые обладают достаточным приближением и образуют кусочно-кубический полином [13]. Сплайн-интерполяция по сравнению с другими методами интерполяции обеспечивает наилучшее приближение.

Ниже исследуется сплайн-интерполяция с помощью кубических полиномов. Перед началом вычислений вводятся $n+1$ опорные точки, т. е. запоминается $n+1$ пара значений $x_j, y_j (j=1, 2, \dots, n+1)$. Опорные точки могут быть не эквидистантны, т. е. интервалы между ними не обязательно равны. Для $n+1$ опорной точки существуют n интервалов (рис. 5.3). В j -м интервале $x_j \leq x \leq x_{j+1}$. На

Рис. 5.3. Кусочная интерполяция в $n=6$ интервалах



каждом интервале для сплайн-интерполяции составляет уравнение

$$S_j = a_j(x - x_j)^3 + b_j(x - x_j)^2 + c_j(x - x_j) + d_j. \quad (5.1-2)$$

Для n интервалов необходимо найти $4n$ неизвестных, поскольку для каждого интерполирующего сплайна S_j вычисляются значения коэффициентов a_j, b_j, c_j, d_j .

Любой сплайн должен удовлетворять четырем условиям.

1) В каждой нижней границе интервала ($x=x_j$) сплайн проходит через опорную точку

$$S_j(x_j) = y_j, \text{ где } j = 1, 2, \dots, n. \quad (5.1-3)$$

2) В каждой верхней границе интервала ($x = x_j + h_j$) сплайн проходит через опорную точку

$$S_j(x_j + h_j) = y_{j+1}, \text{ где } j = 1, 2, \dots, n. \quad (5.1-4)$$

Ширина интервала

$$h_j = x_{j+1} - x_j. \quad (5.1-5)$$

Поскольку опорные точки в общем случае не эквидистантны, коэффициент h_j целесообразно сохранять в памяти ЭВМ.

3) Для каждой граничной точки интервала сплайн имеет одинаковую крутизну в обоих граничащих интервалах

$$S'_{j-1}(x_{j-1} + h_{j-1}) = S'_j(x_j), \quad (5.1-6)$$

где $j = 1, 2, \dots, n-1$.

4) Для каждой граничной точки интервала сплайн имеет в обоих граничащих интервалах одинаковую кривизну

$$S''_j(x_j + h_j) = S''_{j+1}(x_{j+1}), \quad (5.1-7)$$

где $j = 1, 2, \dots, n-1$.

Для n интерполирующих сплайнов требуется $4n-2$ условия, а вычисление $4n$ коэффициентов осуществляется с помощью $4n$ условий. Итак еще два условия можно выбрать произвольно. Эти условия назовем граничными:

а) через обе граничные точки кривая должна проходить с нулевой кривизной, «превращаясь» в прямую. Это означает, что вторая производная любого интерполирующего сплайна на границах области равна нулю:

$$S''_1(x_1) = 0, \quad (5.1-8a)$$

$$S''_n(x_{n+1}) = S''_n(x_n + h_n) = 0; \quad (5.1-8b)$$

б) крутизна любого интерполирующего сплайна в обеих граничных точках фиксирована, т. е.

$$S'_1(x_1) = m_a, \quad (5.1-9a)$$

$$S'_n(x_{n+1}) = m_e = S'_n(x_n + h_n). \quad (5.1-9b)$$

Помимо перечисленных должны быть заданы еще два значения функции. Например, для кривой намагничивания до области насыщения задаются начальная прони-

цаемость $\mu_a = m_a$ и магнитная постоянная $\mu_0 = m_e$.

5.1.2. Система уравнений сплайн-коэффициентов. Вычислим $4n$ коэффициентов с помощью заданных опорных точек. Проще всего определить коэффициенты a_j из первого условия. Подставляя уравнение (5.1-3) и $x = x_j$ в уравнение (5.1-5), получаем

$$d_j = y_j, \text{ где } j = 1, 2, \dots, n. \quad (5.1-10)$$

Для отыскания численных значений коэффициентов из второго условия подставляем уравнение (5.1-4) и $x = x_j + h_j$ в (5.1-2). С учетом уравнения (5.1-10) получаем

$$a_j h_j^3 + b_j h_j^2 + c_j h_j + d_j = d_{j+1}, \quad (5.1-11)$$

где $j = 1, 2, \dots, n$.

Согласно третьему условию необходимо взять первую производную интерполирующего сплайна

$$S'_j = \frac{d(S_j)}{dx} = 3a_j(x - x_j)^2 + 2b_j(x - x_j) + c_j. \quad (5.1-12)$$

Подставляя в выражение (3.1-12) уравнение (5.1-6), имеем

$$3a_{j-1}h_{j-1}^2 + 2b_{j-1}h_{j-1} + c_{j-1} = c_j, \quad (5.1-13)$$

где $j = 2, \dots, n$.

Для удовлетворения четвертого условия вычислим вторую производную интерполирующего сплайна:

$$S''_j = \frac{d^2(S_j)}{dx^2} = 6a_j(x - x_j) + 2b_j. \quad (5.1-14)$$

Подставляя в это выражение уравнение (5.1-7), получаем

$$6a_j h_j + 2b_j = 2b_{j+1}, \quad (5.1-15)$$

где $j = 1, 2, \dots, n-1$.

Посредством исключения сплайн-коэффициентов a_j и c_j из уравнений (5.1-11), (5.1-13) и (5.1-15) вместе с граничными условиями (5.1-8) или (5.1-9) получаем систему линейных уравнений для всех b_j . Теперь решим (5.1-15) относительно

$$a_j = \frac{1}{3h_j} (b_{j+1} - b_j), \quad (5.1-16)$$

где $j = 1, 2, \dots, n-1$, и подставим его в (5.1-11):

$$\frac{h_j^2}{3} (b_{j+1} - b_j) + h_j^2 b_j + h_j c_j = d_{j+1} - d_j. \quad (5.1-17)$$

Это уравнение в свою очередь решим относительно

$$c_j = \frac{d_{j+1} - d_j}{h_j} - \frac{h_j}{3} (b_{j+1} + 2b_j), \quad (5.1-18)$$

где $j=1, \dots, n-1$.

Для первого члена в правой части (5.1-18) в дальнейшем будем пользоваться сокращением

$$e_j = \frac{d_{j+1} - d_j}{h_j} = \frac{y_{j+1} - y_j}{h_j}, \quad (5.1-19)$$

где $j=1, \dots, n$.

Опорные точки должны быть сопоставимы с непрерывной функцией, т. е. функция не должна иметь скачков, поскольку из-за $h_j=0$ значение e_j стремилось бы к бесконечности.

Чтобы исключить коэффициенты a_j и c_j , подставим уравнения (5.1-16) и (5.1-18) в уравнение (5.1-13) и после преобразований запишем

$$h_{j-1} b_{j-1} + 2(h_{j-1} + h_j) b_j + h_j b_{j+1} = 3(e_j - e_{j-1}), \quad (5.1-20)$$

где $j=2, \dots, n-1$.

Оставшиеся два уравнения, необходимые для вычисления всех коэффициентов b_j , получаем с помощью граничных условий. Таким образом задаются две системы уравнений.

а) Подставляя уравнение (5.1-8а) при $j=1$ и $x=x_j$ в уравнение (5.1-14), получаем

$$b_1 = 0. \quad (5.1-21)$$

Для упрощенного расчета граничного условия (5.1-8б) допустим, что за опорной точкой x_{n+1} существует другой интерполирующий сплайн, для которого условия (5.1-3), (5.1-6) и (5.1-7) выполняются. Согласно последнему условию

$$S_n''(x_n + h_n) = S_{n+1}''(x_{n+1}).$$

Подставляя это выражение при $j=n+1$ и $x=x_{n+1}$ в уравнение (5.1-14), получаем

$$b_{n+1} = 0. \quad (5.1-22)$$

Аналогичным образом можно записать (5.1-20) и для $j=n$, т. е.

$$h_{n-1} b_{n-1} + 2(h_{n-1} + h_n) b_n = 3(e_n - e_{n-1}). \quad (5.1-23)$$

Уравнения (5.1-20) и (5.1-23) представляют собой си-

стему линейных уравнений, с помощью которой вычисляются $n-1$ сплайн-коэффициентов $b_2 \dots b_n$. В матрице коэффициентов заполненными оказываются только главная диагональ и две соседние диагонали, а все остальные элементы равны нулю. Такая система уравнений называется трехдиагональной. В целях упрощения заменим индекс $n-1$ в уравнениях (5.1-25) и (5.1-34) на индекс m и запишем

$$m = n - 1. \quad (5.1-24)$$

Тогда

$$\begin{array}{|c|c|c|c|c|c|} \hline 2(h_1 + h_2) & h_2 & 0 & & 0 & 0 \\ \hline h_2 & 2(h_2 + h_3) & h_3 & & 0 & 0 \\ \hline & & & & & \\ \hline & & & & & \\ \hline 0 & 0 & 0 & & h_m & 2(h_m + h_n) \\ \hline \end{array} \times$$

$$\begin{array}{|c|} \hline b_2 \\ \hline b_3 \\ \hline b_4 \\ \hline b_m \\ \hline b_n \\ \hline \end{array} \times = \begin{array}{|c|} \hline 3(e_2 - e_1) \\ \hline 3(e_3 - e_2) \\ \hline \\ \hline 3(e_n - e_m) \\ \hline \end{array}. \quad (5.1-25)$$

После решения системы уравнений, приведенной в п. 5.1.3, предполагаем, что все коэффициенты b_j известны. С помощью уравнений (5.1-16) и (5.1-20) вычисляются все n коэффициентов a_j , а с помощью (5.1-18) — все коэффициенты c_j .

б) С учетом граничного условия (5.1-9) создается система из n уравнений для коэффициентов b_j . Подставляя (5.1-9а) для $j=1$ и $x=x_1$ в (5.1-12), получаем

$$c_1 = m_a. \quad (5.1-26)$$

Из уравнения (5.1-18)

$$2h_1 b_1 + h_1 b_2 = 3(e_1 - m_a). \quad (5.1-27)$$

После подстановки (5.1-9б) в (5.1-12) имеем

$$3a_n h_n^2 + 2b_n h_n + c_n = m_e. \quad (5.1-28)$$

Из уравнения (5.1-11) для $j=n$

$$a_n h_n^2 + b_n h_n + c_n = e_n, \quad (5.1-29)$$

а для $j=n-1$

$$a_{n-1} h_{n-1}^2 + b_{n-1} h_{n-1} + c_{n-1} = e_{n-1}; \quad (5.1-30)$$

из уравнения (5.1-14) для $j=n$

$$3a_{n-1} h_{n-1}^2 + 2b_{n-1} h_{n-1} + c_{n-1} = c_n; \quad (5.1-31)$$

из уравнения (5.1-15) для $j=n-1$

$$3a_{n-1} h_{n-1} + b_{n-1} = b_n \quad (5.1-32)$$

Таким образом, получаем систему уравнений

$$2h_{n-1} b_{n-1} + (4h_{n-1} + 3h_n) b_n = 9e_n - 6e_{n-1} - 3m_e, \quad (5.1-33)$$

которую необходимо решить относительно b_{n-1} и b_n .

Уравнения (5.1-20), (5.1-27) и (5.1-33) образуют систему уравнений всех сплайн-коэффициентов $b_1 \dots b_n$. Эта система трехдиагональна, однако ее матрица коэффициентов в отличие от уравнения (5.1-25) несимметрична. Однако в п. 5.1.4 показано, что несимметричную систему уравнений (5.1-34), если ее дополнить двумя операторами, можно рассчитать с помощью подпрограммы, использующейся для решения симметричной системы уравнений. Несимметричная система уравнений записывается в виде

$$\begin{array}{|c|c|c|c|c|c|} \hline 2h_1 & h_1 & 0 & \dots & 0 & 0 \\ \hline h_1 & 2(h_1 + h_2) & h_2 & \dots & 0 & 0 \\ \hline 0 & h_2 & 2(h_2 + h_3) & \dots & 0 & 0 \\ \hline \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ \hline \vdots & \vdots & \vdots & \vdots & h_m & \vdots \\ \hline 0 & 0 & 0 & \vdots & 2h_m & 4h_m + 3h_n \\ \hline \end{array} \times$$

$$\times \begin{array}{|c|} \hline b_1 \\ \hline b_2 \\ \hline b_3 \\ \hline \vdots \\ \hline b_m \\ \hline b_n \\ \hline \end{array} = \begin{array}{|c|} \hline 3(e_1 - m_a) \\ \hline 3(e_2 - e_1) \\ \hline 3(e_3 - e_2) \\ \hline \vdots \\ \hline 9e_n - 6e_m - 3m_e \\ \hline \end{array}. \quad (5.1-34)$$

После решения этой системы по уравнениям (5.1-10), (5.1-29) и (5.1-13) рассчитаем коэффициенты a_j для $j=1, \dots, n-1$, a_n и c_j для $j=2, \dots, n$. Коэффициент c_1 определяется из уравнения (5.1-26).

5.1.3. Решение трехдиагональной системы уравнений.

Ввиду симметрии системы уравнений (5.1-25) воспользуемся для ее решения алгоритмом Гаусса (п. 2.4.1). Поскольку интерполируемая функция должна быть непрерывна, расстояния между опорными точками положительны, т. е. $h_j > 0$. Из этого условия определяется наибольший по модулю элемент в каждой строке, расположенный на главной диагонали матрицы коэффициентов. Рассмотрим теперь подпрограмму UPR 5000. В отличие от UPR 2000 в ней коэффициенты и правые части системы уравнений запоминаются в трех одномерных массивах. После выполнения подпрограммы правая часть системы уравнений перекрывается вектором решения, а коэффициенты на главной диагонали стираются из памяти ЭВМ.

На каждом шаге редукции в строках исключаются только коэффициенты на главной диагонали и правая часть, поскольку остальные коэффициенты равны нулю. При обратной подстановке каждый диагональный элемент, расположенный рядом с главной диагональю, умножается на предварительно вычисленный элемент вектора решения и вычитается из правой части системы уравнений, а полученная разность делится на коэффициент главной диагонали.

Редукция системы уравнений и обратная подстановка отделяются друг от друга оператором RETURN. Следовательно, при вызове подпрограммы необходимы два оператора GOSUB, однако появляется возможность обрабатывать и несимметричные системы уравнений (п. 5.1.4).

Подпрограмма UPR 5000

а) необходимые переменные в вызывающей программе:

DH(NT) — элементы главной диагонали;

DH(NT) — элементы дополнительной диагонали;

NT — количество уравнений;

RS(NT) — правая часть;

б) используемые имена переменных:

I1, I2 — переменные цикла;

R1 — коэффициент редукции;

в) результат:

RS(NT) — вектор решения.

5000 REM: Решение симметричной

```

5002 REM: трехдиагональной системы уравнений
5004 REM
5006 IF NT>1 THEN 5012
5008 PRINT «NT=»;NT;«Решение невозможно»
5010 GOTO 9999
5012 FOR I2=2 TO NT
5014 I1—I2—1
5016 R1=DN(I1)/DH(I1)
5018 DH(I2)=DH(I2)—R1 * DN(I1)
5020 RS(I2)=RS(I2)—R1 * RS(I1)
5022 NEXT I2
5024 RETURN
5026 RS(NT)=RS(NT)/DH(NT)
5028 FOR I1=NT—1 TO 1 STEP—1
5030 RS(I1)=(RS(I1)—DN(I1) * RS(I1+1)/DH(I1)
5032 NEXT
5034 RETURN

```

5.1.4. Вычисление сплайн-коэффициентов. Эти коэффициенты по граничному условию а) (см. п. 5.1.1) в соответствии с уравнением (5.1-8) определяются с помощью подпрограммы UPR 5100. В строках 5108—5116 для заданных опорных точек коэффициенты e_j вычисляются по уравнению (5.1-19). Затем в строках 5118—5128 составляется трехдиагональная система уравнений (5.1-25). Ее решение дает сплайн-коэффициенты b_j , а после этого вычисляются a_j и c_j .

Подпрограмма UPR 5100 (она вызывает UPR 5000)

а) необходимые переменные в вызывающей программе:

- I — номер ветви;
- NI — количество n интервалов;
- XS(J, I) — опорные точки x_j ;
- YS(J, I) — значение функции y_j для опорной точки;

б) используемые имена переменных:

- DH(NI) — элементы главной диагонали;
- DN(NI) — элементы дополнительной диагонали;
- J — переменная цикла;
- J1 — вспомогательная величина;
- NT — количество уравнений;
- RS(NI) — правая часть (вектор решения);
- SE(NI) — коэффициенты e_j ;
- SH(NI) — коэффициенты h_j ;

в) результат:

- AS(J,I) — сплайн-коэффициенты a_j ветви I;
- BS(J,I) — сплайн-коэффициенты b_j ветви I;
- CS(J,I) — сплайн-коэффициенты c_j ветви I.

```

5100 REM: Вычисление сплайн-коэффициентов согласно
5102 REM: граничному условию а)
5104 REM
5106 NT=NI—1
5108 FOR J=1 TO NI
5110 J1=J+1

```

```

5112 SH(J)=XS(J1,I)—XS(J,I)
5114 SE(J)=(YS(J1, I)—YS(J, I))/SH(J)
5116 NEXT
5118 FOR J=1 TO NT
5120 J1=J+1
5122 DH(J)=2* (SH(J)+SH(J1))
5124 DN(J)=SH(J1)
5126 RS(J)=3* (SE(J1)—SE(J))
5128 NEXT
5130 GOSUB 5006
5132 GOSUB 5026
5134 BS(1, I)=0
5136 RS(NI)=0
5138 FOR J=1 TO NI
5140 BS(J+1, I)=RS(J)
5142 AS(J,I)=(RS(J)—BS(J,I))/(3* SH(J))
5144 CS(J,I)=SE(J)—SH(J)* (RS(J)+2* BS(J,I))/3
5146 NEXT
5148 RETURN

```

В подпрограмме UPR 5200 для расчета сплайн-коэффициентов по граничному условию б) (см. п. 5.1.1) в соответствии с уравнением (5.1-9) сначала вычисляются коэффициенты h_j из (5.1-5), а затем e_j из (5.1-19). В граничных интервалах 1 и n интерполяция осуществляется отрезками прямых линий, причем коэффициенты a_1 , b_1 и a_n , b_n равны нулю. Коэффициенты $c_1=m_a=e_1$ и $c_n=m_e=e_n$ в каждом случае определяют наклон прямой в граничном интервале.

При составлении трехдиагональной системы уравнений (5.1-34) необходимо обратить внимание на то, что первый и последний элементы главной диагонали должны вычисляться отдельно. Поскольку сплайн-коэффициенты граничных областей известны, количество уравнений n_t меньше, чем количество интервалов n (строка 5208). При решении системы (5.1-34) следует учесть, что матрица коэффициентов не симметрична: элемент дополнительной диагонали в строке n_t в 2 раза больше элемента в строке n_t-1 и столбце n_t той же диагонали. Поэтому редукцию элемента главной диагонали и элемента правой части в строке n_t после вызова подпрограммы UPR 5000 необходимо выполнить еще раз (строки 5254 и 5256).

После обратной подстановки через коэффициенты b_j вычисляются a_j и c_j .

Подпрограмма UPR 5200 (она вызывает UPR 5000)

а)—в) — см. UPR 5100; дополнительно используются вспомогательные величины: $N1=NT-1$ и $N3=N1-3$.

```

5200 REM: Вычисление сплайн-коэффициентов согласно
5202 REM: граничному условию б)
5204 REM
5206 N1=NI-1
5208 NT=NI-2
5210 N3=NI-3
5212 FOR J=1 TO NI
5214 J1=J+1
5216 SH(J)=XS(J1, I)-XS(J, I)
5218 SE(J)=(YS(J1, I)-YS(J, I))/SH(J)
5220 NEXT
5222 AS(1, I)=0
5224 BS(1, I)=0
5226 CS(1, I)=SE(1)
5228 AS(NI, I)=0
5230 BS(NI, I)=0
5232 CS(NI, I)=SE(NI)
5234 FOR J=1 TO N3
5236 J1=J+1
5238 DH(J)=2 * (SH(J)+SH(J1))
5240 DN(J)=SH(J1)
5242 RS(J)=3 * (SE(J1)-SE(J))
5244 NEXT
5246 DH(1)=2 * SH(2)
5248 DH(NT)=4 * SH(NT)+3 * SH(N1)
5250 RS(NT)=9 * SE(N1)-6 * SE(NT)-3 * SE(NI)
5252 GOSUB 5006
5254 DH(NT)=DH(NT)-R1* DN(N3)
5256 RS(NT)=RS(NT)-R1* RS(N3)
5258 GOSUB 5026
5260 CS(2, I)=SE(1)
5262 BS(2, I)=RS(1)
5264 FOR J=2 TO NT
5266 J1=J+1
5268 BS(J1, I)=RS(J)
5270 AS(J, I)=(BS(J1, I)-BS(J, I)/13 * SH(J))
5272 CS(J1, I)=2 BS(J, I)+3 * AS(J, I) * SH(J)
5274 CS(J1, I)=CS(J, I)+SH(J) * CS(J1, I)
5276 NEXT
5278 AS(N1, I)=(SE(N1)-CS(N1, I))/SH(N1)
5280 AS(N1, I)=(AS(N1, I)-BS(N1, I))/SH(N1)
5282 RETURN

```

5.1.5. Интерполяция. Если сплайн-коэффициенты a_j , b_j и c_j вычислены для опорных значений x_j , y_j , то x_k , соответствующее y_k , можно определить с помощью интерполяции. Для этого в подпрограмме UPR 5300 проверяется, в каком интервале j лежит значение x_k . Необходимо, чтобы для интервала j выполнялось условие $x_j \leq x_k \leq x_{j+1}$.

Номер интервала j определяется через переменную IV в строке 5322. Если $x_k < x_1$ или $x_k > x_{n+1}$, то подпрограмма выдает сообщение на периферийное устройство.

В этом случае $IV=0$. После определения номера интервала IV согласно уравнениям (5.1-2) и (5.1-10) вычисляется значение

$$y_k = a_j (x_k - x_j)^3 + b_j (x_k - x_j)^2 + c_j (x_k - x_j) + y_j. \quad (5.1-35)$$

Кроме того, с помощью уравнения (5.1-12) определяется наклон интерполирующего сплайна $m_k = S'_j$ в точке $(x_k; y_k)$, т. е.

$$m_k = 3a_j (x_k - x_j)^2 + 2b_j (x_k - x_j) + c_j. \quad (5.1-36)$$

При интерполяции предполагается, что каждому значению x_k однозначно соответствует значение y_k . Это условие выполняется, например, для функции, изображенной на рис. 5.4, а для функциональной зависимости, изо-

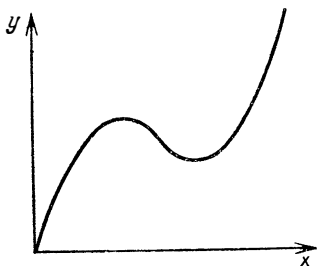


Рис. 5.4. Функция $y=f(x)$

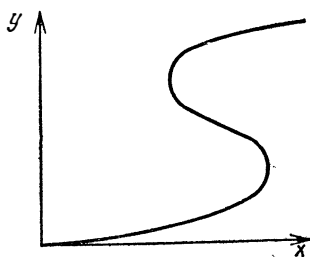


Рис. 5.5. Зависимость y - и x -значений

браженной на рис. 5.5, описанная сплайн-интерполяция не применяется.

При создании подпрограммы UPR 5300 учитывалось, что в цепи может существовать всего NZ нелинейных двухполюсников. Номер ветви каждого такого двухполюсника является вторым индексом сплайн-коэффициента и опорной точки.

Подпрограмма UPR 5300

- а) необходимые переменные в вызывающей программе:
- AS(J, I); BS(J, I); CS(J, I) — сплайн-коэффициенты;
 - I — номер ветви;
 - NS(I) — количество опорных точек;
 - XK — координата x ;
 - XS(J, I) — опорные точки;
 - YS(J, I) — значения функции для опорных точек;

б) используемые имена переменных:

AI — вспомогательные величины;

II — переменная цикла;

в) результат:

IV — номер интервала;

SK — наклон m_k интерполирующего x_k сплайна для y ;

YK — координата y .

5300 REM: Интерполяция

5302 REM

5304 IV=0

5306 IF XK XS(1, I) THEN 5316

5308 FOR II=2 TO NS(I)

5310 IF XK>XS(II, I) THEN 5314

5312 GOTO 5322

5314 NEXT II

5316 PRINT «Ветвь»;I;«:Значение XK лежит»

5318 PRINT «вне области определения»

5320 RETURN

5322 IV=II-1

5324 X1=XK-XS(IV, I)

5326 AI=AS(IV, I) * X1

5328 YK=((AI+BS(IV, I))* X1+CS(IV, I)) * X1+YS(IV, I)

5330 SK=(3 * AI+2 * BS(IV, I)) * X1+CS(IV, I)

5332 RETURN

5.1.6. Программа SPLINE. С помощью этой программы выполняется интерполяция нелинейной кривой $y=f(x)$. После ввода опорных точек (строки 128—140) сплайн-коэффициенты вычисляются в строках 142—154 (см. п. 5.1.1). При этом учитывается также, что в цепи может существовать NZ нелинейных источников. Номер ветви нелинейного источника I является вторым индексом сплайн-коэффициента и опорных точек. В программе SPLINE I=1.

В зависимости от выбора опорных точек интерполируемая кривая может значительно отклоняться от заданной кривой (см. п. 5.1.7). В программе SPLINE проверка точности интерполяции осуществляется достаточно просто, так как рассчитывается экстремальное значение интерполирующего сплайна. Для экстремального значения производная интерполирующего сплайна равна нулю. С учетом уравнения (5.1-12) и условия $x_1=x-x_j$ получаем

$$3a_j x_1^2 + 2b_j x_1 + c_j = 0. \quad (5.1-37)$$

Каждое действительное решение этого квадратного или для $a_j=0$ линейного уравнения определяет x -координату экстремума. Ее значение вычисляется в строке 200, а соответствующее значение y определяется с помощью подпрограммы UPR 5300.

```

100 PRINT «SPLINE»
102 PRINT «Интерполяция с помощью»
104 PRINT «сплайн функций»
106 PRINT
108 DEFINT I—K, M—N
110 DEFDBL A—D, G, L, P—Z
112 I=1
114 INPUT «Количество опорных точек»; NM
116 DIM XS(NM, 1) YS(NM, 1) BS(NM, 1) NS(1)
118 NS(I)=NM
120 N1=NM—1
122 DIM AS(N1, 1), CS(N1, 1), SE(N1), SH(N1)
124 DIM DH(N1), DN(N1), RS(N1)
126 PRINT
128 PRINT «Ввод опорных точек»
130 FOR J=1 TO NS(I)
132 PRINT «X»; J;
134 INPUT «:»; XS(J, I)
136 PRINT «Y»; J;
138 INPUT «:»; YS(J, I)
140 NEXT J
142 INPUT «Граничные условия» (A/B): »:FR$
144 IF FR$=«A» THEN 150
146 IF FR$=«B» THEN 154
148 GOTO 142
150 GOSUB 5100
152 GOTO 156
154 GOSUB 5200
156 FOR J=1 TO N1
158 IF AS(J, I)=0 THEN 178
160 W1=BS(J, I) 2—3 * AS(J, I) * CS(J, I)
162 IF W1>0 THEN 168
164 IF W1=0 THEN 172
166 GOTO 184
168 X1=(—BS(J, I)+SQR(W1))/(3 * AS(J, I))
170 GOSUB 200
172 X1=(—BS(J, I)—SQR(W1))/(3 * AS(J, I))
174 GOSUB 200
176 GOTO 184
178 IF BS(J, I)=0 THEN 184
180 X1=—CS(J, I)/(2 * BS(J, I))
182 GOSUB 200
184 NEXT J
186 PRINT
188 PRINT «Интерполяция»
190 INPUT «x=»; XK
192 GOSUB 5300
194 IF IV=0 THEN 186
196 PRINT «y=»; YK
198 GOTO 186
200 XK=X1+XS(J, I)
202 IF XK>XS(J+1, I) THEN 210
204 IF XK<XS(J+1, I) THEN 210
206 GOSUB 5300
208 PRINT «Экстремальные значения:»; XK; «:»; YK

```

210 RETURN

1000 REM: UPR 5000, 5100, 5200, 5300

5.1.7. Пример. Исследуем интерполяцию точек $A (-1, 0)$, $B (0, 1)$, $C (1, 0)$. Функция $y=f(x)$ для $x < -1$ и $x > 1$ должна быть равна нулю. Сплайн-коэффициенты определяются с учетом граничного условия б) (см. п. 5.1.1). На рис. 5.6 приведена кривая, соответствующая рассмотренному примеру.

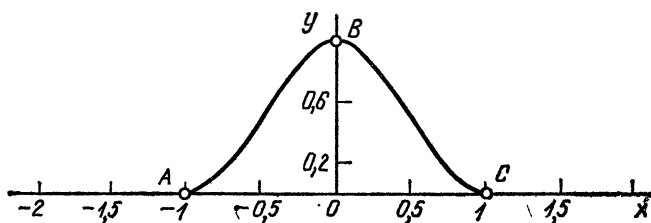


Рис. 5.6. Интерполирующий сплайн между точками A , B , C при линейной интерполяции в граничных областях

Если сплайн-коэффициенты для тех же опорных точек вычисляются при граничном условии а) (см. п. 5.1.1), то в результате интерполяции при $x < -1$ и $x > 1$ $y \neq 0$, причем для $x = 4,8$ может быть даже $y = -2,4$.

Интерполяция с помощью сплайн-функции

Количество опорных точек? 5

Ввод опорных точек

X1: ? — 10

Y1: ? 0

X2: ? —1

Y2: ? 0

X3: ? 0

Y3: ? 1

X4: ? 1

Y4: ? 0

X5: ? 10

Y5: ? 0

Граничное условие (а/б) : ?

Значение экстремума: —4.8384749174118;

—2.3982244195095368

Значение экстремума: —1.986821492050428 D—08;

.9999999999999994

Значение экстремума: 1.98682149135653 D—08;
.99999999999999994

Значение экстремума: 4.803848072886467;
—2.398224195095336.

Если описанная интерполяция выполняется при граничном условии а) в областях $x < -1$ и $x > 1$, то должно быть задано значительно большее количество опорных точек, чем в приведенном примере. Только в этом случае функция будет удовлетворять заданным требованиям.

5.2. Моделирование схемных элементов

Для описанной программы анализа цепей реальные элементы заменяются двухполюсниками и управляемыми источниками, и такой процесс называется моделированием.

С помощью модели упрощенно описывается процесс или объект. Однако в ряде случаев простая модель не отражает существенных свойств реального процесса или объекта. Для более точных моделей, с одной стороны, может быть потеряна их наглядность, а с другой, определение модельных параметров и их обработка вызывают дополнительные сложности.

Систематическое отображение процесса моделирования схемных элементов в данной книге, к сожалению, невозможно. Ниже будут схематично рассмотрены нелинейные двухполюсные элементы при постоянном напряжении и транзисторы.

5.2.1. Двухполюсные схемные элементы. При постоянном напряжении их можно описать вольт-амперной характеристикой. Предварительно необходимо рассмотреть вопрос о том, какие оси (I или U) соответствуют осям x , y .

Если вольт-амперная характеристика нелинейного двухполюсника не имеет частных экстремумов, то ось U может соответствовать оси x , а ось I — оси y (рис. 5.7). Нелинейный двухполюсник с подобной характеристикой будем называть нелинейной проводимостью, поскольку тангенс угла наклона кривой к оси I в каждой точке соответствует дифференциальной проводимости.

Если вольт-амперная характеристика линейного двухполюсника имеет один или несколько частных экстремумов, то необходимо исследовать, к какому типу кривых

она относится. В случае зависимости $I=f(U)$ каждому значению напряжения U соответствует одно значение тока I . Характеристика этого типа с двумя экстремумами называется характеристикой N -типа [32] (рис. 5.8).

Для вольт-амперной характеристики N -типа ось U можно сопоставить оси x , а ось I — оси y . Таким образом,

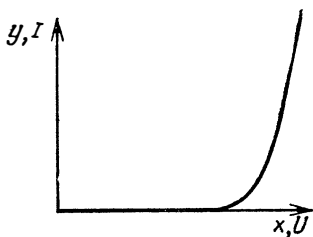


Рис. 5.7. Вольт-амперная характеристика диода без относительных экстремумов

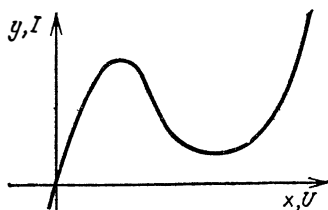


Рис. 5.8. Вольт-амперная характеристика туннельного диода N -типа

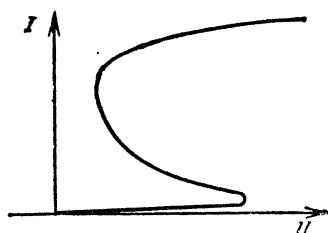


Рис. 5.9. Вольт-амперная характеристика лампы накаливания S -типа

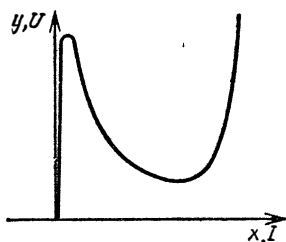


Рис. 5.10. Вольт-амперная характеристика лампы накаливания для сплайн-интерполяции

нелинейный двухполюсник N -типа также оказывается нелинейной проводимостью. Если напряжение является функцией тока, то каждому значению тока I соответствует единственное значение напряжения U . Характеристику этого типа с двумя экстремумами называют кривой S -типа (рис. 5.9). Теперь поменяем оси местами и изобразим кривую. Оси x соответствует ось T , а оси y — ось U (рис. 5.10). Назовем изображенный таким образом двухполюсник нелинейным сопротивлением, поскольку тангенс угла наклона кривой в каждой точке соответствует дифференциальному сопротивлению.

Как уже упоминалось, описанная в § 5.1 сплайн-интерполяция применима только в том случае, когда функция $y=f(x)$ является непрерывной. В программе анализа цепей нелинейные проводимости и сопротивления обрабатываются отдельно. С помощью переменной `zweigart` [IA(I) в программе на языке БЕЙСИК] определяется, является ли ветвь I проводимостью [IA(I)=1] или сопротивлением [IA(I)=2].

5.2.2. Моделирование транзистора. Представление биполярного транзистора с эмиттерным включением линейной схемой замещения (см. рис. 2.19) не полностью отражает его реальные характеристики по двум причинам. Во-первых, транзистор имеет нелинейную входную характеристику (рис. 5.11). Во-вторых, управляемый источник

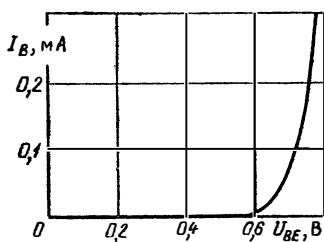


Рис. 5.11. Входная характеристика транзистора BC 107 с эмиттерным включением при температуре окружающей среды 25 °C

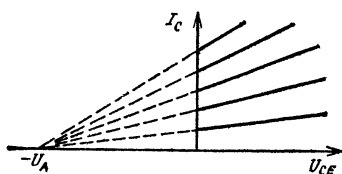


Рис. 5.12. Выходные характеристики транзистора BC 107 с эмиттерным включением при температуре окружающей среды 25 °C

тока с постоянной проводимостью G_{CE} описывает выходную характеристику транзистора с постоянным углом наклона. Однако вследствие эффекта Эрли выходные характеристики обладают различными углами наклона [33].

Упрощенные отображения эффекта Эрли можно получить с помощью так называемого напряжения Эрли. Для этого необходимо продлить зависимости токов коллектора от напряжения до точки их пересечения с осью напряжения (рис. 5.12). Отмеченные недостатки линейной схемы замещения устраняются в случае применения нелинейной схемы замещения. Для этого в ветви база—эмиттер воспроизведем входную характеристику нелинейной проводимости G_{BE} и учтем эффект Эрли за счет нелинейной проводимости G_{CE} в ветви коллектор—эмиттер.

Выходные характеристики тока базы I_B при токе коллектора I_{C0} пересекают ось I_C (рис. 5.13), причем

$$I_{C0} = BI_B, \quad (5.2-1)$$

где B — коэффициент усиления по току в режиме короткого замыкания.

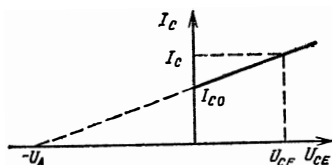


Рис. 5.13. Создание нелинейной схемы замещения биполярного транзистора с эмиттерным включением

Проводимость

$$G_{CE} = \frac{I_{C0}}{U_A} = \frac{BI_B}{U_A} \quad (5.2-2)$$

соответствует тангенсу угла наклона прямой, являющейся выходной характеристикой при токе базы I_B . При напряжении U_{CE} ток коллектора

$$I_C = G_{CE} U_{CE} + I_{C0}. \quad (5.2-3)$$

Подставляя в это уравнение выражения (5.2-1) и (5.2-2), получаем

$$I_C = \frac{BI_B}{U_A} U_{CE} + BI_B. \quad (5.2-4)$$

Второй член в правой части уравнения (5.2-4) описывает уже известный источник тока, управляемый током базы I_B с коэффициентом усиления по току B (рис. 2.19). Первый член в правой части уравнения (5.2-4) характеризует ток, протекающий через проводимость

$$G_{CE} = \frac{B}{U_A} I_B = k_A I_B. \quad (5.2-5)$$

Здесь

$$k_A = \frac{B}{U_A}, \quad (5.2-6)$$

где k_A — константа Эрли.

Напряжение Эрли U_A определяется как точка пересечения множества продленных прямых, являющихся вы-

ходными характеристиками транзисторов. Для транзистора BC 107 напряжение $U_A = 180$ В. С помощью проводимостей G_{BE} и G_{CE} получаем улучшенную схему замещения биполярного транзистора, включенного по схеме с общим эмиттером (рис. 5.14). Проводимость G_{CE} ,

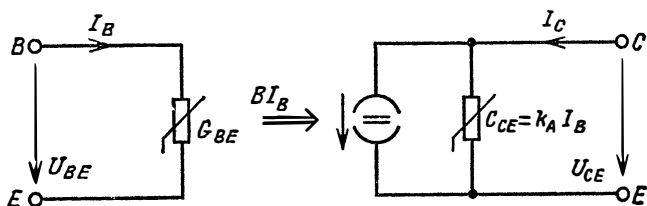


Рис. 5.14. Нелинейная схема замещения биполярного транзистора с общим эмиттером

зависящая от тока базы I_B , является нелинейным двухполюсником, расчет которого приводится в § 5.4.

При рассмотрении схемы замещения транзистора на рис. 2.19 прежде чем приступить к анализу цепи, необходимо оценить положение рабочей точки на его входной характеристике, поскольку от этого зависят параметры элементов схемы. С помощью нелинейной схемы замещения, приведенной на рис. 5.14, можно описать и рассчитать рабочий режим транзистора, причем задаются только его входная характеристика, напряжение Эрли U_A и коэффициент усиления по току B в режиме холостого хода.

5.3. Метод Ньютона

5.3.1. Обобщение метода. Если в цепи больше двух нелинейных источников, то рабочая точка характеристики определяется итерационными методами. В этом случае приблизительное решение, обозначенное в программе как начальное значение, многократно уточняется в процессе вычислений. Если приближенные решения x_1, x_2, x_3, \dots близки к искомому решению, то метод называется конвергентным, т. е. сходящимся. Итерации выполняются до тех пор, пока не достигается требуемая точность решения.

Вполне возможно, что метод последовательных приближений окажется не сходящимся, а расходящимся (ди-

вергентным). В этом случае вычислительный процесс необходимо прервать и все начать сначала, но уже с другим начальным приближением.

Из множества итерационных методов рассмотрим только метод Ньютона, поскольку он является наиболее эффективным для расчета цепей с нелинейными двухполусниками. Иногда метод Ньютона называют также методом Ньютона—Рафсона.

В методе Ньютона улучшенное значение x_{k+1} приближенного значения x_k определяется по следующей итерационной формуле:

$$x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)}. \quad (5.3-1)$$

При этом $f(x_k)$ является значением функции в точке x_k , а $f'(x_k)$ — производной функции $f(x)$ в этой точке.

Геометрическая интерпретация метода приведена на рис. 5.15. Производная в точке x_k для значения функции $f(x_k)$ равняется тангенсу угла наклона α , т. е.

$$f'(x_k) = \operatorname{tg} \alpha = \frac{f(x_k)}{\Delta x_k}. \quad (5.3-2)$$

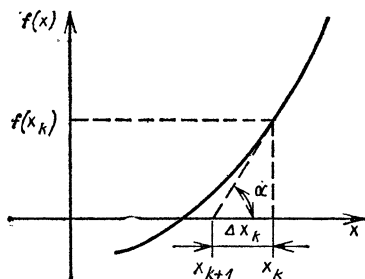


Рис. 5.15. Геометрическая интерпретация метода Ньютона

Если в уравнение, описывающее улучшенное приближенное значение, подставить значение Δx_k из (5.3-2), то получим алгоритм итерации

$$x_{k+1} = x_k - \Delta x_k. \quad (5.3-3)$$

5.3.2. Сходимость метода Ньютона. Метод сходится, если начальное приближение достаточно близко к решению. Выбор подходящего начального значения функции связан с двумя проблемами: во-первых, производная $f'(x)$ может быть равна нулю, тогда улучшение приближенного решения с помощью уравнения (5.3-1) невозмож-

но. Во-вторых, нелинейное уравнение $f(x)=0$ может иметь несколько решений в зависимости от выбранного начального значения.

Исследования показывают, что сходимость метода Ньютона зависит не только от значения функции $f(x)$ и наклона $f'(x)$, но также и от кривизны $f''(x)$ [49]. Метод сходится, если для всех $x_{\text{нач}} \leq x \leq x_{\text{иск}}$ выполняется следующее условие:

$$|f(x)f''(x)| \neq [f'(x)]^2. \quad (5.3-4)$$

Из этого неравенства видно, что $f'(x)$ не должна быть равна нулю. Выполнение условия (5.3-4) означает, что для всех x значение первой производной $f'(x)$ функции не должно быть слишком малым, а второй производной $f''(x)$ — слишком большим.

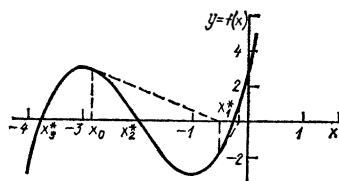
Пример 5.1. Рассмотрим сходимость метода Ньютона при решении нелинейного уравнения

$$1,5x^3 + 9x^2 + 13,5x + 3 = 0 \quad (5.3-5)$$

с начальным значением $x_0 = -2,85$. Дополнительно к значению функции $f(x_0) = 2,90$ подставим в уравнение (5.3-4) наклон $f'(x_0) = -1,25$ и кривизну $f''(x_0) = -7,65$. При выбранном значении x_0 условие сходимости не выполняется. На рис. 5.16 показано, что при $x_0 = -2,85$ достигается не ближайшее решение x_2^* , а x_1^* .

Под термином «сходимость метода Ньютона» будем понимать получение не любого, а лишь ближайшего к на-

Рис. 5.16. Решение уравнения (5.3-5) методом Ньютона с начальным значением $x_0 = -2,85$



начальному значению решения. Это имеет смысл прежде всего в том случае, когда область определения функции $f(x)$ ограничена. Неподходящее начальное значение решения может в этом случае привести к тому, что первое приближенное значение будет найдено за пределами области определения.

Отмеченные трудности при ограниченной области определения устраняются достаточно просто. Если началь-

ное или приближенное значение лежит вне области определения, то рассчитываются значения функции, образующие границу этой области (см. подпрограмму UPR 5400).

Подпрограмма UPR 5400

а) необходимые переменные вызывающей программы:

- I — номер ветви;
- NS(I) — количество опорных точек;
- SW(I) — начальное или приближительное значение;
- XS(I, I) — опорные точки вблизи нижней границы;
- XS(N1, I) — опорные точки вблизи верхней границы области

определения;

б) используемые имена переменных:

- N1 — количество опорных точек;

в) результат:

- XK — координата x для интерполяции.

```

5400 REM: Подготовка интерполяции
5402 REM
5404 N1=NS(I)
5406 IF SW(I)>SX(N1, I) THEN 5414
5408 IF SW(I)<XS(I, I) THEN 5418
5410 XK=SW(I)
5412 RETURN
5414 XK=XS(N1, I)
5416 RETURN
5418 XK=XS(1, I)
5420 RETURN

```

5.3.3. Улучшение сходимости. Известно, что сходимость метода Ньютона зависит не только от угла наклона, но и от знака второй производной (кривизны) функции $f(x)$ между начальным значением и решением. При определенных значениях угла наклона и кривизны может возникнуть ситуация, когда метод не сходится, хотя начальное значение лежит относительно близко от искомого решения.

Пример 5.2. Исследуем сходимость метода при решении нелинейного уравнения

$$(x - 8)^{1/3} = 0, \quad (5.3-6)$$

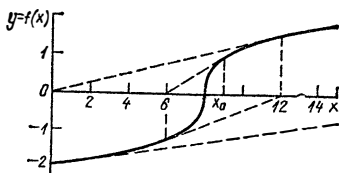
причем исходить будем из начального значения решения $x_0=9$, которое лежит относительно близко к искомому решению $x^*=8$. На рис. 5.17 показано, что приближенное значение выбрано неудачно, так как каждое следующее приближение все больше удаляется от искомого решения и метод расходится. При итерациях приближенные решения совершают циклические колебания вокруг искомого решения. Однако можно достаточно простым,

но эффективным образом добиться сходимости метода Ньютона.

Для этого вместо приближенного значения x_{k+1} , вычисленного с помощью уравнения (5.3-1), воспользуемся «сокращенным» значением

$$x_{k+1}^{(r)} = \frac{x_k + x_{k+1}}{2}. \quad (5.3-7)$$

Рис. 5.17. Несходимость метода Ньютона при решении уравнения (5.3-6) с начальным значением $x_0=9$



С учетом уравнения (5.3-1) запишем

$$x_{k+1}^{(r)} = x_k - 0,5 \frac{f(x_k)}{f'(x_k)}. \quad (5.3-8)$$

В табл. 5.1 показано, как «сокращенные» значения постепенно приближаются к искомому решению несмотря на то, что итерационный процесс расходится.

Таблица 5.1. Сравнение приближенных значений x_{k+1} , полученных из уравнения (5.3-1), с «сокращенными» значениями

k	x_{k+1}	$x_{k+1}^{(r)}$
0	6	7,5
1	12	8,25
2	0	7,875
3	24	8,0625
4	-24	7,9688
5	74	8,0156
6	-120	7,9922
7	264	8,0039

Если условие сходимости выполняется, то уравнение (5.3-8) не обеспечивает такой быстрой сходимости итерационного процесса, как уравнение (5.3-1). С другой стороны, с помощью уравнения (5.3-8) можно обеспечить сходимость даже в том случае, если при невыполненном условии сходимости уже совершено несколько итераций.

В дальнейшем рассмотрении объединим оба метода

и будем при $k \leq 4$ использовать уравнение (5.3-1), а при $k > 4$ — уравнение (5.3-8).

5.3.4. Использование метода Ньютона. Распространим этот метод на расчет цепей, содержащих один двухполюсник. Заменим линейную цепь двухполюсников, соединенную с нелинейным двухполюсником, соответствующим источником тока с собственным значением I_{qe} и внутренней проводимостью G_{ie} (рис. 5.18).

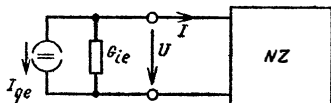


Рис. 5.18. Связь линейной цепи, состоящей из двухполюсников, с нелинейным двухполюсником NZ

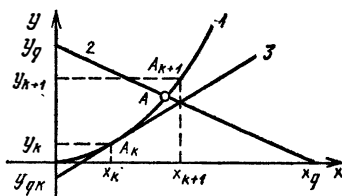


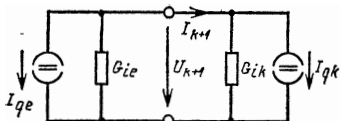
Рис. 5.19. y — x -характеристика нелинейного 1 и линейного 2 двухполюсников

Нелинейный двухполюсник в зависимости от его типа описывается выходной характеристикой ток—напряжение (I — U) или напряжение—ток (U — I). Выходная характеристика двухполюсника представляет собой прямую линию с соответствующими проекциями на оси координат x_q и y_q (рис. 5.19). В качестве приближенного значения для ближайшего шага итерации задается точка A_k с координатами (x_k, y_k) . С помощью уравнения (5.1-36) можно вычислить наклон m_k выходной характеристики нелинейного двухполюсника. Определим точку A_{k+1} с координатами (x_{k+1}, y_{k+1}) как улучшенное по сравнению с точкой A_k приближение к рабочей точке A .

Координата x_{k+1} искомой точки A_{k+1} является координатой точки пересечения выходной характеристики источника (прямая 2 на рис. 5.19) с линеаризованной в точке A_k характеристикой нелинейного двухполюсника (прямая 3 на рис. 5.19), пересекающейся с осью y_{qk} и имеющей наклон m_k . Касательную (прямая 3) к характеристике нелинейного двухполюсника (кривая 1) в точке A_k можно интерпретировать как U — I - или I — U -характеристику линейного источника тока с собственным значением I_{qk} и внутренней проводимостью G_{ik} , который называ-

ют дискретной токовой моделью. С помощью дискретной токовой модели для нелинейного двухполюсника в точке A_k (рис. 5.20) вычисляется напряжение U_{k+1} или ток ветви I_{k+1} . Если двухполюсник является нелинейной проводимостью, то оси x соответствует ось напряжений U , а оси y — ось токов I . В качестве приближенного значе-

Рис. 5.20. Дискретная токовая модель (I_{qk} ; G_{ik}) нелинейного двухполюсника в точке A_k линейной цепи из двухполюсников (I_{qe} ; G_{ie})



ния x_k задается напряжение ветви нелинейного двухполюсника. Таким образом, ток I_k , т. е. координата y_k , и наклон m_k касательной определяется методом интерполяции.

Затем вычисляются значения параметров дискретной токовой модели.

Отрезок оси y_{qk} характеризует ток источника

$$I_{qk} = I_k - m_k U_k, \quad (5.3-9)$$

наклон m_k — внутреннюю проводимость

$$G_{ik} = m_k, \quad (5.3-10)$$

а искомая координата x_{k+1} точки A_{k+1} — напряжение ветви U_{k+1} нелинейного двухполюсника.

Если двухполюсник является нелинейным сопротивлением, то ось x соответствует оси токов I , а ось y — оси напряжений U . В качестве приближенного значения x_k задается ток ветви I_k нелинейного двухполюсника. Посредством интерполяции определяются координата y_k , т. е. напряжение U_k , и наклон m_k касательной в точке A_k .

Отрезок оси y_{qk} характеризует напряжение источника дискретной модели токового контура

$$U_{qk} = U_k - m_k I_k, \quad (5.3-11)$$

а наклон m_k касательной — внутреннее сопротивление R_{ik} .

Теперь определим ток источника

$$I_{qk} = I_k - U_k/m_k \quad (5.3-12)$$

и внутреннюю проводимость дискретной токовой модели

$$G_{ik} = 1/m_k. \quad (5.3-13)$$

Искомая координата x_{k+1} точки A_{k+1} равняется току ветви нелинейного двухполюсника.

При дальнейшем улучшении приближенного значения x_{k+1} должны быть снова определены элементы дискретной токовой модели для точки A_{k+1} (рис. 5.19). Итерации метода Ньютона заканчиваются, если для двух значений x_k и x_{k+1} выполняется следующее условие:

$$\left| \frac{x_{k+1} - x_k}{x_k} \right| < \varepsilon. \quad (5.3-14)$$

Слишком большое значение ε приводит к быстрому прекращению вычислительного процесса, причем последнее значение x может существенно отличаться от тока и напряжения в рабочей точке. Вместе с тем значение ε выбирается и не очень малым, поскольку значения x_k и x_{k+1} вследствие ошибок дискретизации и округления всегда несколько отличаются друг от друга.

Целесообразнее всего, чтобы $\varepsilon \approx 10^{-5} \div 10^{-6}$. Для $x_k = 0$ можно не проверять выполнение условия (5.3-14). В этом случае итерация оканчивается, когда $|x_{k+1}| \leq \varepsilon$. Если цепь содержит несколько нелинейных двухполюсников, то для каждого итерационного шага двухполюсник замещается своей токовой моделью. Решение системы уравнений определяется напряжениями узлов и соответственно параметрами ветвей нелинейной цепи [7].

5.4. Программа NEWTON на языке БЕЙСИК

С помощью этой программы выполняется анализ цепи постоянного тока, содержащей несколько нелинейных двухполюсников. Метод Ньютона применяется при условии его сходимости для каждой ветви цепи.

После ввода в ЭВМ и обработки глобальных переменных для каждого нелинейного двухполюсника задается его тип и количество опорных точек. Для $NS(I) = 0$ в ветви I принимается проводимость, управляемая током, коэффициент управления которой запоминается в массиве $SF(I)$, а ветвь управления — в массиве $MG(I)$.

Для остальных нелинейных двухполюсников используется сплайн-интерполяция. Когда на границе области определения кривая имеет нулевую кривизну (см. п. 5.1.1), количество опорных точек $NS(I) \geq 4$, поскольку трехдиагональная система уравнений может быть решена только для двух и более уравнений. Когда крутизна

интерполирующего сплайна фокусируется в обеих точках границ, количество опорных точек $NS(I) \geq 5$. Во входном цикле (строки 160—188) определяются максимальное количество опорных точек NM и размеры массивов. Во втором цикле (строки 198—258) задаются коэффициенты управления, управляющие ветви или опорные точки, граничные условия и начальные приближения.

Каждый нелинейный двухполюсник образует ветвь с номером, равным номеру двухполюсника. Ветви цепи должны нумероваться последовательно: нелинейные источники получают наименьшие номера (1, ..., NZ), затем нумеруются линейные двухполюсники ($NZ+1$, ..., IR). Идеальные источники напряжения получают самые большие номера (§ 2.2).

После определения массива узлов подготавливается перемещение идеальных источников напряжения. Далее начинается расчет по методу Ньютона: сначала для каждого нелинейного двухполюсника определяется токовая модель и с ее помощью составляется система линейных уравнений. Посредством узловых напряжений упрощенно вычисляются параметры нелинейных ветвей, поскольку цепь с одним нелинейным двухполюсником не содержит управляемых источников тока.

При нелинейной проводимости величина x_{j+1} является напряжением ветви $UA(I)$, а при нелинейном сопротивлении — током ветви $SA(I)$. В зависимости от типа двухполюсника каждому значению в строках 380—386 сопоставляется элемент $ZS(I)$ массива $ZS(NZ)$. Значение x_k заносится в массив $SW(I)$. Если для каждого нелинейного двухполюсника выполняется условие (5.3-14), то вычисления оканчиваются в строке 396. В последнюю очередь вычисляются все параметры ветвей.

Посредством ввода нового начального значения проверяется целесообразность дальнейшего продолжения расчета. Если условия (5.3-14) для двухполюсника не выполняются, то цикл запросов игнорируется. Затем осуществляется следующий шаг итерации, причем переменная цикла k (в программе KL) увеличивается на единицу.

Перед следующим выполнением цикла x_{k+1} параметры ветви [в программе это $ZS(I)$] запоминаются (строки 402—414) в качестве x_k параметров [в программе $SW(I)$]. С целью улучшения сходимости вычислительного процесса для $k > 4$ в соответствии с уравнением (5.3-7) запоминаются x_{k+1} параметров ветви.

Если условие (5.3-14) после 20 итерационных шагов выполняется не для всех NZ нелинейных двухполюсников, то итерационный процесс прерывается, поскольку метод либо вообще не сходится, либо сходится плохо.

```

100 PRINT «NEWTON»
102 PRINT «Узловой анализ цепи»
104 PRINT «постоянного тока»
106 PRINT «с нелинейными двухполюсниками»
108 PRINT
110 DEFINT I—K, M—N
112 DEFDBL A—D, G, L, P—S, U—Z
114 REM
116 REM:
118 INPUT «Количество ветвей»; IZ
120 INPUT «Количество узлов без базового»; KN
122 INPUT «Количество идеальных источников напряжения»; IS
124 PRINT
126 N=KN-IS
128 IF IZ IS THEN 132
130 IF N>0 THEN 138
132 PRINT «Ввод не имеет смысла»
134 PRINT «или решение невозможно»
136 GOTO 118
138 IR=IZ-IS
140 DIM UN(KN), KF(KN), A(N, N), X(N)
142 DIM SF(IR), MZ(IR), VR(IR), MI(IR), MU(IR)
144 DIM KV(IR), KZ(IR), GD(IR), SD(IR), SA(IR), UA(IR)
146 DIM KA(IS), KU(IS), QT(IS)
148 REM
150 REM: Ввод нелинейных двухполюсников
152 INPUT «Количество нелинейных двухполюсников»; NZ
154 IF NZ<1 THEN 152
156 DIM IA(NZ), MG(NZ), NS(NZ), SW(NZ), ZS(NZ)
158 NM=3
160 FOR I=1 TO NZ
162 PRINT «Двухполюсник»; I;
164 INPUT »: «Тип двухполюсника (G/R)»; FR$
166 IF FR$=«G» THEN 172
168 IF FR$=«R» THEN 176
170 GOTO 164
172 IA(I)=1
174 GOTO 178
176 IA(I)=2
178 INPUT «Количество опорных точек»; NS(I)
180 IF NS(I)=0 THEN 188
182 IF NS(I) 3 THEN 178
184 IF NS(I) NM THEN 188
186 NM=NS(I)
188 NEXT
190 NI=NM-1
192 DIM XS(NM, NZ), YS(NM, NZ), BS(NM, NZ)
194 DIM AS(NI, NZ), CS(NI, NZ), SE(NI), SH(NI)
196 DIM DH(NI), DN(NI), RS(NI)

```

```

198 FOR I=1 TO NZ
200 PRINT «Ветвь»; I
202 PRINT «из узла» »; KV(I)
204 PRINT «к узлу» »; KZ(I)
206 IF NS(I)>0 THEN 216
208 PRINT «Проводимость, управляемая током»
210 INPUT «Коэффициент управления»; SF(I)
212 INPUT «Ветвь управления»; MG(I)
214 GOTO 258
216 PRINT «Ввод опорных точек»
218 NI=NS(I)-1
220 PRINT «Двухполюсник»; I
222 FOR J=1 TO NS(I)
224 PRINT «X»; J;
226 INPUT «: »; XS(J, I)
228 PRINT «Y»; J;
230 INPUT «: »; YS(J, I)
232 NEXT J
234 INPUT «Граничные условия (A/B);»; FR$
236 IF FR$=«A» THEN 242
238 IF FR$=«B» THEN 246
240 GOTO 234
242 GOSUB 5100
244 GOTO 248
246 GOSUB 5200
248 PRINT «Начальное значение»;
250 ON IA(I) GOTO 252, 256
252 INPUT «напряжения»; SW(I)
254 GOTO 258
256 INPUT «тока»; »; SW(I)
258 NEXT I
260 REM
262 REM: Ввод линейных двухполюсников
264 IG=0
266 FOR I=NZ+1 TO IZ
268 PRINT «Ветвь»; I
270 GOSUB 2500
272 NEXT I
274 REM
276 REM: Определение выводимых величин
278 GOSUB 3400
280 REM
282 REM: Определение массива узлов
284 GOSUB 2100
286 REM
288 REM: Подготовка перемещения
290 REM: идеальных источников напряжения
292 GOSUB 2600
294 REM
296 REM: Итерация метода Ньютона
298 EN=.00001
300 FOR KL=1 TO 20
302 REM: Определение дискретной токовой модели
304 REM: для каждого нелинейного двухполюсника
306 FOR I=1 TO NZ

```

```

308 IF NS(I)=0 THEN 336
310 GOSUB 5400
312 GOSUB 5300
314 IF IV=0 THEN 420
316 ON IA(I) GOTO 320, 328
318 REM: Нелинейная проводимость
320 GD(I)=SK
322 SD(I)=YK-SK * XK
324 GOTO 342
326 REM: Нелинейное сопротивление
328 GD(I)=1/SK
330 SD(I)=XK-YK/SK
332 GOTO 342
334 REM: Управляемая проводимость
336 J=MG(I)
338 GD(I)=SF(I) * (SD(J)+GD(J) * SW(J))
340 SD(I)=0
342 NEXT
344 REM
346 REM: Составление и решение
348 REM: системы линейных уравнений
350 GOSUB 2700
352 REM
354 REM: Расчет узловых напряжений
356 GOSUB 2800
358 REM: Вычисление параметров ветвей
360 REM: нелинейных двухполюсников
362 FOR I=1 TO NZ
364 J=KV(I)
366 K=KZ(I)
368 UA(I)=UN(J)-UN(K)
370 SA(I)=UA(I) * GD(I)+SD(I)
372 ON IA(I) GOTO 374, 378
374 ZS(I)=UA(I)
376 GOTO 380
378 ZS(I)=SA(I)
380 NEXT
382 REM: Запрос, пригодно ли приближение
384 FOR I=1 TO NZ
386 IF NS(I)=0 THEN 396
388 IF ABS(SW(I))<1E-30 THEN 394
390 IF ABS((ZS(I)-SW(I))/SW(I))>EN THEN 402
392 GOTO 396
394 IF ABS(ZS(I))>EN THEN 402
396 NEXT
398 GOTO 434
400 REM: Запоминание параметров ветвей
402 FOR I=1 TO NZ
404 IF NS(I)=0 THEN 414
406 IF KL>4 THEN 412
408 SW(I)=ZS(I)
410 GOTO 414
412 SW(I)=.5 * (ZS(I)+SW(I))
414 NEXT
416 NEXT KL

```

```

418 PRINT «Прерывание ввиду расходимости»
420 PRINT «Выбрать новое начальное значение»
422 FOR I=1 TO NZ
424 IF NS(I)=0 THEN 430
426 PRINT «Ветвь» ;I;
428 INPUT«: »; SW(I)
430 NEXT
432 GOTO 300
434 REM
436 REM: Вывод параметров ветви
438 GOSUB 2900
440 GOSUB 4326
442 GOTO 420
1000 REM: UPR 2000, 2100, 2200, 2300, 2400
1002 REM: UPR 2500, 2600, 2700, 2800, 2900
1004 REM: UPR 3400, 4326
1006 REM: UPR 5000, 5100, 5200, 5300, 5400

```

В качестве примера рассчитаем параметры рабочей точки транзистора в схеме, представленной на рис. 5.21.

Рис. 5.21. Транзистор BC 107 с делителем базового напряжения с подключенным линейным источником напряжения

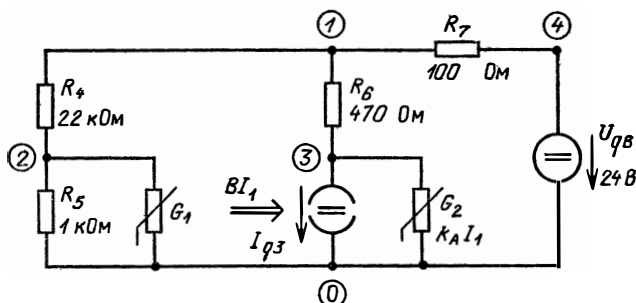
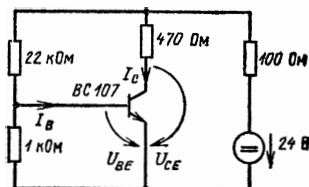


Рис. 5.22. Схема на рис. 5.21 со схемой замещения на рис. 5.14 для транзистора с эмиттерным включением

Заменяем транзистор нелинейной схемой замещения, приведенной на рис. 5.14. Сначала пронумеруем ветви и узлы. Значения параметров для нелинейной входной характеристики позаимствуем из рис. 5.11. С помощью напря-

жения Эрли $U_A=180$ В и коэффициента усиления по току $B=2,25$ в режиме короткого замыкания вычислим с учетом уравнения (5.2-6) постоянную Эрли $k_A=1,25$ В⁻¹ для управляемой проводимости G_2 (рис. 5.22). В результате получим $I_E=I_1=151,7$ мкА, $U_{BE}=U_1=0,74$ В, $I_C=I_6=34,9$ мА, $U_{CE}=U_2=U_3=4,02$ В.

Рассчитаем рабочую точку транзистора по программе NEWTON.

Узловой анализ цепи постоянного тока с нелинейными двухполюсниками

Количество ветвей? 8

Количество узлов (без базового)? 4

Количество идеальных источников напряжения? 1

Количество нелинейных двухполюсников? 2

Двухполюсник 1: тип двухполюсника

Количество опорных точек? 7

Двухполюсник 2: тип двухполюсника

Количество опорных точек? 0

Ветвь 1

от узла? 2

к узлу? 0

Ввод опорных точек

Двухполюсник 1

X1 : ? 0

Y1 : ? 0

X2 : ? .55

Y2 : ? 0

X3 : ? .6

Y3 : ? 5E-6

X4 : ? .65

Y4 : ? 20E-6

X5 : ? .7

Y5 : ? 75E-6

X6 : ? .75

Y6 : ? 180 E-6

X7 : ? .8

Y7 : ? 400 E-6

Граничные условия (a/b) : ? В

Начальное значение напряжения? .75

Ветвь 2

от узла? 3

к узлу? 0

Проводимость, управляемая током

Коэффициент управления? 1.25

Управляющая ветвь? 1

Ветвь 3

от узла? 3

к узлу? 0

Элемент? S

Управляемый источник тока

Управляющая величина (I/U)? I

Усиление по току B=? 225

Управляющая ветвь? 1

Элемент? N

Ветвь 4

от узла? 1

к узлу? 2

Элемент? R

Сопротивление? 22000

Элемент? N

Ветвь 5

от узла? 2

к узлу? 0

Элемент? R

Сопротивление? 1000

Элемент? N

Ветвь 6

от узла? 1

к узлу? 3

Элемент? R

Сопротивление? 470

Элемент? N

Ветвь 7

от узла? 4

к узлу? 1

Элемент? R

Сопротивление? 100

Элемент? N

Ветвь 8

от узла? 4

к узлу? 0

Напряжение источника? 24

Какие параметры ветвей дожны быть выведены?

Количество напряжений ветвей? 7

Количество токов ветвей? 7

U1=.7427526478890945 В

U2=4.019486444029783 В

U3=4.019486444029783 В

U4=19.67814030128992 В

U5=.7427526478890945 В

U6=16.40140650514924 В

U7=3.579107050820981 В

I1=1.517082748968112 D—04 А

I2=7.622477336413847 D —04 А

I3=3.413436185178251 D—02 А

I4=8.944609227859056 D—04 А

I5=7.427526478890945 D—04 А

I6=3.489660958542391 D—02 А

I7=3.579107050820981 D—02 А

НЕЛИНЕЙНЫЕ ДИНАМИЧЕСКИЕ ЦЕПИ

Для расчета динамической цепи в гл. 4 мы заменяли индуктивности $L = \text{const}$ и емкости $C = \text{const}$ их дискретными токовыми моделями. Таким образом, динамическая цепь в момент времени t_{j+1} сводилась к цепи постоянного тока.

Для расчета нелинейной цепи постоянного тока нелинейные проводимость и сопротивление в каждой точке их амперных характеристик при итерационном шаге в методе Ньютона замещались их токовыми моделями. Анализ полученной линейной цепи постоянного тока повторялся до тех пор, пока не получалось нужное решение.

Дискретные токовые модели индуктивности и емкости, а также вольт-амперные характеристики нелинейных сопротивлений и проводимостей можно использовать и для расчета динамических цепей. Для этого заменим в определенные моменты времени t_{j+1} емкости и индуктивности их дискретными токовыми моделями и выполним итерацию методом Ньютона для нелинейных проводимостей и сопротивлений, причем будем изменять токовые модели нелинейных проводимостей и сопротивлений до тех пор, пока не добьемся сходимости процесса.

Вычислительные затраты в этом случае оказываются существенно выше, чем при анализе линейной динамической цепи, прежде всего потому, что ньютоновские итерации необходимо выполнять для каждого момента времени.

6.1. Нелинейные двухполюсные накопители энергии

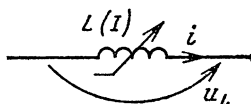
Предположение о том, что емкость C и индуктивность L должны быть постоянны, не позволяет рассчитывать целый ряд схем. Во многих случаях индуктивность L двухполюсника не постоянна, а зависит от тока. Например, для катушки индуктивности зависимость ее индуктивности от тока характеризуется нелинейной кривой намагничивания материала сердечника. Емкость двухполюсника также часто не постоянна, а зависит от напряжения, например в случае электролитического конденсатора или конденсатора с керамическим диэлектриком [12].

Поэтому в дальнейшем будем рассматривать индук-

тивности, зависящие от тока, и емкости, зависящие от напряжения. Эти двухполюсники назовем нелинейными двухполюсными накопителями энергии. Наша цель состоит в том, чтобы заменить каждый нелинейный двухполюсный накопитель энергии его дискретной токовой моделью.

6.1.1. Нелинейная индуктивность. Для индуктивности, зависящей от тока, взаимосвязь между напряжением u_L

Рис. 6.1. Использование стрелок для отображения отношений индуктивности $L(I)$, зависящей от тока



и током i в функции времени выглядит следующим образом (рис. 6.1):

$$u_L = \frac{d(Li)}{dt}. \quad (6.1-1)$$

Произведение индуктивности L и тока I называется потоком сцепления и записывается в виде

$$\Psi_m = LI. \quad (6.1-2)$$

Подставим (6.1-2) в (6.1-1), тогда

$$u_L = \frac{d\Psi_m}{dt} = \frac{d\Psi_m}{dI} \frac{di}{dt}. \quad (6.1-3)$$

Выражение

$$L_d = \frac{d\Psi_m}{dI} \quad (6.1-4)$$

назовем дифференциальной индуктивностью [12], которая представляет собой тангенс угла наклона кривой $\Psi_m = f(I)$ к оси абсцисс.

При использовании понятия дифференциальной индуктивности связь между током и напряжением задается уравнением

$$I_L = L_d \frac{di}{dt}. \quad (6.1-5)$$

С помощью этой индуктивности удастся заменить индуктивность, зависящую от тока, в каждой точке ее характеристики дискретной токовой моделью (п. 6.1.3).

Если функция $\Psi_m = f(I)$ представляет собой прямую

линию, то L_d не зависит от тока и постоянна во времени. Если же прямая проходит через начало координат, то предполагается особый случай, когда $L = L_d = \text{const}$. Если $\Psi_m = f(I)$ нелинейна, то как индуктивность

$$L = \frac{\Psi_m}{I}, \quad (6.1-6)$$

так и дифференциальная индуктивность L_d не зависит от тока. Из уравнения (6.1-4) не очевидно, что дифференциальная индуктивность L_d в общем случае зависит от тока.

Путем модернизации программы SPLINE была разработана программа DIFF-IND для расчета индуктивности L и дифференциальной индуктивности L_d магнитного контура. Строки 100—104 выглядят теперь следующим образом:

```
100 PRINT «DIFF—IND»
102 PRINT «Дифференциальная индуктивность»
104 PRINT «магнитного контура»
```

Для зависимости $\Psi_m = f(I)$ переменной I соответствует ось x , а переменной Ψ_m — ось y . В результате изменяется комментарий при вводе:

```
130 FOR J=1 TO NS(I)
132 PRINT «I»; J;
134 INPUT «;»; XS(J, I)
136 PRINT «Psi»; J;
138 INPUT «;»; YS(J, I)
140 NEXT J
```

Перед интерполяцией считывается значение тока, которое обозначается переменной ХК. Результат УК интерполяции представляет собой потокосцепление. Индуктивность L является согласно уравнению (6.1-4) частным от деления УК на ХК, а дифференциальная индуктивность — тангенсом угла наклона SK кривой $\Psi_m = f(I)$ к оси абсцисс в точке ХК. В соответствии с вышеизложенным изменяются строки 188—196 программы SPLINE и завершается формирование программы DIFF-IND:

```
188 INPUT «I в A:»; ХК
190 GOSUB 5300
192 IF IV=0 THEN 186
194 ЕК=YK/ХК
196 PRINT «L=»; ЕК; «H; Ld=»; SK; «H»
198 GOTO 188
```

Пример 6.1. Рассчитаем дифференциальную индуктивность L_d кольцевой катушки с количеством витков

$N=800$. Сердечник из магнитомягкого материала, средний диаметр которого $D=30$ см, не имеет воздушного зазора, а площадь его поперечного сечения $A=9$ см². Кривая намагничивания материала сердечника совпадает с характеристиками материала V 170—50 А [12].

В первую очередь определим на основе кривой намагничивания характеристику $\Psi_m=f(I)$ магнитного контура: для каждого значения напряженности поля $H=IN/l$ рассчитаем соответствующий ток I (здесь $l=\pi D$ является средней длиной силовой линии поля в сердечнике); каждое значение индукции B умножим на площадь поперечного сечения сердечника A и количество витков N , чтобы вычислить потокосцепление в веберах.

Теперь запустим программу DIFF-IND. В области крутого подъема кривой намагничивания выберем большое, а в остальных областях — малое количество опорных точек.

В обеих граничных областях необходимо осуществить линейную интерполяцию; для этого определим сплайн-коэффициенты при граничном условии, согласно которому крутизна сплайна в обеих точках границ области фиксирована. Опорные точки для первого интервала выберем таким образом, чтобы наклон прямой в этом интервале был равен начальной проницаемости $\mu_a=1.5 \cdot 10^{-3}$ Вб/(А·м). Наклон прямой в последнем интервале равен магнитной проницаемости воздуха μ_0 .

Не распечатанные значения напряженности поля H можно вычислить, умножив значение тока на коэффициент $N/l=849$ м⁻¹. В результате получаем значение индукции B , в котором потокосцепление делится на коэффициент $NA=0,72$ м².

Программа DIFF-IND

Дифференциальная индуктивность магнитного контура

Количество опорных точек? 15

Ввод опорных точек

I1:? 0

Psi 1:? 0

I2:? 1.1781 E-3

Psi 2:? 1.0872 E-3

I3:? .0471

Psi 3:? .079

I4:? .0,707

Psi 4:? .1872

I5:? .0942

Psi 5:? .389

I6:? .1178

Psi 6: ? .547
 I7: ? .165
 Psi 7: ? .706
 I8: ? .212
 Psi 8: ? .792
 I9: ? .377
 Psi 9: ? .922
 I10: ? .589
 Psi 10: ? .986
 I11: ? .907
 Psi 11: ? .1.03
 I12: ? 1.708
 Psi 12: ? 1.08
 I13: ? 4.477
 Psi 13: ? 1.17
 I14: ? 16.73
 Psi 14: ? 1.3248
 I15: ? 17.91
 Psi 15: ? 1.3257

Граничные условия (а/б): ? б

После ввода опорных точек вычислим индуктивность L и дифференциальную индуктивность L_d для нескольких значений тока I . Результаты расчета в виде распечатки с ЭВМ приведены ниже, а характеристика L ; $L_d = f(I)$ показана на рис. 6.2.

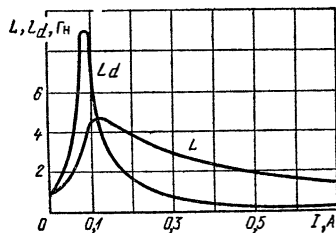


Рис. 6.2. Зависимость индуктивности L и дифференциальной индуктивности L_d кольцевой катушки от тока I

I в А: ? .02
 L = 1.11885 Гн: Ld = 1.415013851815779 Гн
 I в А: ? .05
 L = 1.756229 Гн: Ld = 3.168419019941874 Гн
 I в А: ? .08
 L = 3.278391 Гн Ld = 8.821657209143091 Гн

6.1.2. Нелинейная емкость. Для этой емкости связь между током i и напряжением, зависящим от времени, выглядит следующим образом:

$$i = \frac{d(Cu)}{dt}, \quad (6.1-7)$$

где

$$C = \frac{Q}{U}. \quad (6.1-8)$$

Подставляя выражение (6.1-8) в уравнение (6.1-7), получаем

$$i = \frac{dQ}{dt} = \frac{dQ}{dU} \frac{du}{dt}. \quad (6.1-9)$$

Обозначим через

$$C_d = \frac{dQ}{dU} \quad (6.1-10)$$

один из сомножителей в правой части уравнения (6.1-9) и назовем его дифференциальной емкостью. Этот параметр характеризует тангенс угла наклона функции $Q = f(U)$ к оси абсцисс. С учетом C_d уравнение (6.1-7) можно переписать в виде

$$i = C_d \frac{du}{dt}. \quad (6.1-11)$$

Емкость, зависящую от напряжения, в каждой точке ее характеристики можно заменить с помощью дифференциальной емкости дискретной токовой моделью (п. 6.1.3).

Если функция $Q = f(U)$ линейна, то C_d не зависит от напряжения и времени. Если характеристика проходит через начало координат, то предполагается особый случай, когда $C = C_d = \text{const}$. Если функция $Q = f(U)$ нелинейна, то емкость C и дифференциальная емкость C_d непостоянны. Емкость, зависящую от напряжения, называют также нелинейной.

Для вычисления параметров емкости C и дифференциальной емкости C_d имеется программа на БЕЙСИКЕ DIFF-KAP, представляющая собой версию программы SPLINE. Строки 100—104 здесь выглядят следующим образом:

```
100 PRINT «DIF-KAP»  
102 PRINT «Расчет»  
104 PRINT «дифференциальной емкости»
```

Для кривой $Q = f(U)$ сопоставим переменную U оси x , а переменную Q — оси y . Поэтому изменим комментарии соответствующего ввода:

```
130 FOR J=1 TO NS(I)  
132 PRINT «U»; J;
```

```

134 PRINT «:»; XS(J, I)
136 PRINT «Q»; J;
138 INPUT «:»; YS(J, I)
140 NEXT J

```

Перед каждой интерполяцией считывается значение напряжения и присваивается переменной ХК. Результат УК интерполяции соответствует заряду Q. Параметры емкости C являются согласно уравнению (6.1-8) частными от деления УК и ХК. Рассчитаем эти значения с обычной точностью.

В результате изменения строк 188—196 программы SPLINE получаем программу DIFF-KAP:

```

188 INPUT «U в V:»; XK
190 GOSUB 5300
192 IF IV=0 THEN 186
194 EK=YK/XK
196 PRINT «C=»; EK; «F; Cd=»; SK; «F»

```

Пример 6.2. Рассчитаем значение дифференциальной емкости C_d бариево-титанатового конденсатора (рис. 6.3). Поскольку кривая $Q=f(U)$ в области меньших значений напряжения имеет большую кривизну, зададим дополнительные опорные точки. Ниже приведен расчет дифференциальной емкости по программе DIFF-KAP.

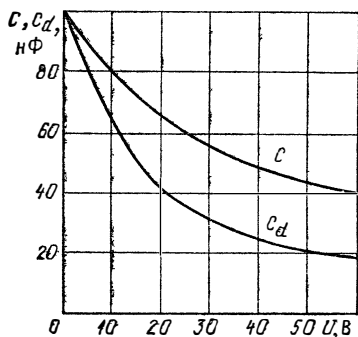


Рис. 6.3. Зависимость емкости C и дифференциальной емкости C_d бариево-титанатового конденсатора от постоянного напряжения U

```

Программа DIFF-KAP
Расчет дифференциальной
емкости
Количество опорных точек? 8
Ввод опорных точек
U1:? 0
Q1:? 0
U2:? 1
Q2:? .0975 E-6
U3:? 10
Q3:? .8E-6
U4:? 20
Q4:? 1.32 E-6
U5:? 30
Q5:? 1.68 E-6
U6:? 40
Q6:? 1.96 E-6
U7:? 50
Q7:? 2.2 E-6
U8:? 60
Q8:? 2.4 E-6
Граничные условия (a/b)? a
U в B:? 1

```

```

C=9.75E-08 F; Cd=9.594511343515253 D-08 F
U в B:? 5
C=8.928274E-08 F; Cd=7.912434769155703 D-08 F
U в B:? 10

```

U в В: ? 20

C=6.6E-08 F; Cd=4.245044580962752 D-08 F

U в В: ? 60

C=4E-08 F; Cd=1.911804583308894 D-08 F

6.1.3. Дискретные токовые модели. Рассмотрим дискретные токовые модели нелинейных емкости и индуктивности, которые используются в методе обратного дифференцирования.

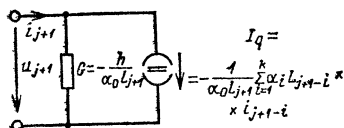
Зависимость дифференциальной емкости C_d от напряжения обозначим индексом, согласованным с соответствующим индексом напряжения. Например, C_{j+1} является дифференциальной емкостью для напряжения u_{j+1} .

Таким образом, связь между током i_{j+1} и напряжением u_{j+1} в t_{j+1} момент времени описывается следующим выражением:

$$i_{j+1} = -\frac{\alpha_0}{h} C_{j+1} u_{j+1} - \frac{1}{h} \sum_{i=1}^k \alpha_i C_{j+1-i} u_{j+1-i}. \quad (6.1.12)$$

Теперь определим токовую модель для метода обратного дифференцирования. Первый член в правой части уравнения (6.1-12) представляет собой ток, выраженный через проводимость G , находящуюся под напряжением u_{j+1} , а второй член — собственный ток идеального источника тока (рис. 6.4). Поскольку емкость C_{j+1-i} и напря-

Рис. 6.4. Дискретная токовая модель нелинейной емкости для метода обратного дифференцирования (BDF)



жение u_{j+1-i} при расчете тока i_{j+1} уже известны, ток источника I_q вычисляется достаточно просто. В уравнении (6.1-12) остается неизвестной дифференциальная емкость C_{j+1} для напряжения u_{j+1} . Поэтому при расчете i_{j+1} или u_{j+1} проводимость G_i также сначала не известна.

Значения тока i_{j+1} и напряжения в методе Ньютона определяются так же, как и нелинейные сопротивления. Напряжение u_{j+1} изменяется от своего начального значения до тех пор, пока решение не сойдется.

Для нелинейной индуктивности во внимание принимается зависимость дифференциальной индуктивности L_d от тока, обозначенная индексом, согласованным с индек-

сом тока. Например, L_{j+1} является дифференциальной индуктивностью при токе i_{j+1} .

Таким образом, в соответствии с уравнением (4.4-17) связь между напряжением u_{j+1} и током i_{j+1} описывается следующим образом:

$$u_{j+1} = -\frac{\alpha_0}{h} L_{j+1} i_{j+1} - \frac{1}{h} \sum_{i=1}^k \alpha_i L_{j+1-i}. \quad (6.1-13)$$

Первый член в правой части уравнения соответствует падению напряжения на сопротивлении, через которое протекает ток i_{j+1} . Второй член может быть интерпретирован как собственное напряжение идеального источника напряжения (рис. 6.5).

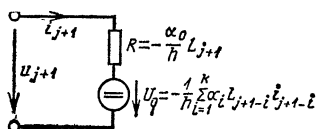


Рис. 6.5. Схема замещения нелинейной индуктивности согласно уравнению (6.1-13)

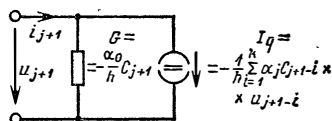


Рис. 6.6. Дискретная токовая модель нелинейной индуктивности для метода обратного дифференцирования

Для метода узловых потенциалов преобразуем линейный источник напряжения в линейный источник тока. При $G=1/R$ и $I_q=-U_q/R$ получаем дискретную токовую модель (рис. 6.6). В уравнении (6.1-13) параметр L_{j+1} является еще неизвестной дифференциальной индуктивностью для тока i_{j+1} . При расчете параметров ветвей для момента t_{j+1} как проводимость G , так и ток источника I_q в дискретной токовой модели не известны. Однако напряжение u_{j+1} может быть вычислено, как и при нелинейном сопротивлении: начиная с начального значения параметр u_{j+1} изменяется до тех пор, пока не будет достигнута сходимость вычислительного процесса.

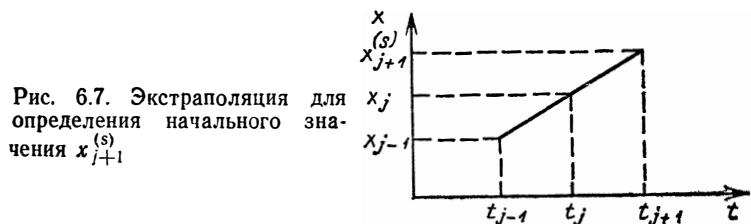
6.2. Начальное значение

Как уже упоминалось, в нелинейной динамической цепи для каждого момента времени выполняются итерации метода Ньютона. Вычислительные затраты зависят от

выбора начального значения (первого приближения): чем ближе оно к решению, тем лучше и быстрее сходится метод. При анализе переходного процесса в момент t_1 начальные значения предварительно задаются только для нелинейных проводимостей и сопротивлений. В нелинейных двухполюсных накопителях энергии могут быть использованы начальные значения параметров состояния. Для всех остальных моментов времени выбирается решение, полученное из предшествующих итераций.

При использовании метода обратного дифференцирования в случае $k > 1$ время расчета можно уменьшить, если для k вычисленных точек построить полином $k-1$ -й степени и с его помощью экстраполировать первое приближение.

В простейшем случае ($k=2$) через точки с коэффициентами $(t_j; x_j)$ и $(t_{j-1}; x_{j-1})$ проводится прямая. После экстраполяции в момент t_{j+1} определяется начальное значение $x_0 = x_{j+1}$ упомянутой ветви (рис. 6.7). Для опорных



точек с одинаковым расстоянием между ними h получаем

$$x_{j+1}^{(s)} = x_j + (x_j - x_{j-1}) = 2x_j - x_{j-1}. \quad (6.2-1)$$

Обобщенное выражение для любого порядка k выглядит следующим образом:

$$x_{j+1}^{(s)} = \sum_{i=1}^k \gamma_i x_{j+1-i}. \quad (6.2-2)$$

Сравнивая (6.2-2) с уравнением (6.3-1), констатируем, что в случае $k=2$ коэффициенты γ_i имеют соответственно значения $\gamma_1=2$ и $\gamma_2=-1$.

Теперь рассчитаем коэффициенты γ_i для $k=3$. В этом случае уравнение (6.2-2) принимает вид

$$x_{j+1}^{(s)} = \gamma_1 x_j + \gamma_2 x_{j-1} + \gamma_3 x_{j-2}. \quad (6.2-3)$$

Попытаемся упорядочить экстраполирующий полином второго порядка по степеням для момента времени $(t_{j+1}-t)$:

$$x(t) = \beta_0 + \beta_1 \frac{t_{j+1}-t}{h} + \beta_2 \left(\frac{t_{j+1}-t}{h} \right)^2. \quad (6.2-4)$$

Экстраполированным для момента $t = t_{j+1}$ первым приближением является

$$x_{j+1}^{(s)} = \beta_0. \quad (6.2-5)$$

Затем вычисляем значения функции $x(t)$ согласно уравнению (6.2-4): в различные моменты времени:

если $t = t_j$, то $x_j = \beta_0 + \beta_1 + \beta_2$;

если $t = t_{j-1}$, то $x_{j-1} = \beta_0 + 2\beta_1 + 4\beta_2$;

если $t = t_{j-2}$, то $x_{j-2} = \beta_0 + 3\beta_1 + 9\beta_2$.

Эти значения и $x_{j+1}^{(s)}$ подставляем теперь в уравнение (6.2-3) и получаем

$$\begin{aligned} \beta_0 = \gamma_1 (\beta_0 + \beta_1 + \beta_2) + \gamma_2 (\beta_0 + 2\beta_1 + 4\beta_2) + \\ + \gamma_3 (\beta_0 + 3\beta_1 + 9\beta_2). \end{aligned} \quad (6.2-6)$$

После преобразований имеем

$$\begin{aligned} \beta_0 = (\gamma_1 + \gamma_2 + \gamma_3) \beta_0 + (\beta_1 + 2\gamma_2 + 3\gamma_3) \beta_1 + \\ + (\gamma_1 + 4\gamma_2 + 9\gamma_3) \beta_2. \end{aligned} \quad (6.2-7)$$

Коэффициенты γ_i теперь могут быть определены с помощью системы уравнений

$$\left. \begin{aligned} \gamma_1 + \gamma_2 + \gamma_3 &= 1, \\ \gamma_1 + 2\gamma_2 + \gamma_3 &= 0, \\ \gamma_1 + 4\gamma_2 + 9\gamma_3 &= 0. \end{aligned} \right\} \quad (6.2-8)$$

Решение этой системы линейных уравнений выглядит следующим образом:

$$\gamma_1 = 3; \quad \gamma_2 = -3; \quad \gamma_3 = 1. \quad (6.2-9)$$

Для любых значений порядка k коэффициенты γ_i являются решениями системы линейных уравнений k -го порядка:

$$\begin{array}{|c|c|c|c|c|} \hline 1 & 1 & 1 & \dots & 1 \\ \hline 1 & 2 & 3 & \dots & k \\ \hline 1 & 4 & 9 & \dots & k^2 \\ \hline \dots & \dots & \dots & \dots & \dots \\ \hline 1 & 2^{k-1} & 3^{k-1} & \dots & k^{k-1} \\ \hline \end{array} \cdot \begin{array}{|c|} \hline \gamma_1 \\ \hline \gamma_2 \\ \hline \gamma_3 \\ \hline \dots \\ \hline \gamma_k \\ \hline \end{array} = \begin{array}{|c|} \hline 1 \\ \hline 0 \\ \hline 0 \\ \hline \dots \\ \hline 0 \\ \hline \end{array}$$

Все γ_i определяются из выражения $\alpha_i = \gamma_i / i$ для $1 \leq i \leq k$ (см. табл. 4.1) с учетом уравнения (4.4-10). В подпрограмме UPR 4800 решение системы уравнений (6.2-10) определяется при каждом заданном значении порядка k по методу обратного дифференцирования.

6.3. Программа DYNA-NL на языке БЕЙСИК

С помощью этой программы выполняется анализ переходного процесса двухполюсной цепи, содержащей NZ нелинейных двухполюсников.

6.3.1. Ввод структуры цепи. При определении глобальных параметров (строки 116—152 в программе DYNA-LIN) следует ввести в ЭВМ также нелинейные двухполюсники (строки 158—288). Для этого программа NEWTON дополняется подпрограммой ввода нелинейных двухполюсных накопителей энергии.

Каждый нелинейный двухполюсник образует ветвь, номер которой соответствует номеру двухполюсника в схеме. Как и для программы NEWTON, в программе DYNA-NL ветви должны всегда нумероваться таким образом, чтобы нелинейные источники получали наименьшие, а идеальные источники — наибольшие номера ветвей.

Тип ветви нелинейного двухполюсника, впрочем, как и линейного, фиксируется переменной IA(I) (табл. 4.2). Массив ZG(I, 1) содержит текущие параметры состояния: в случае емкости — это напряжение, в случае индуктивности — ток. Для каждого нелинейного двухполюсного накопителя энергии, таким образом, в строках 278—286 с помощью интерполяции определяется текущее значение дифференциальной емкости для индуктивности и запоминается в элементе ZE(I, 1) массива ZE(NE, 11). Для функций $C_d = f(U)$ или $L_d = f(I)$, которые должны быть непрерывными, по оси x откладывается ток или напряжение, а по оси y — индуктивность или емкость.

Линейные двухполюсники и идеальные источники напряжения могут быть учтены с помощью подпрограммы UPR 4500.

6.3.2. Анализ переходного процесса в момент времени $T = t_{j+1}$ для цепи с NZ нелинейными двухполюсниками выполняются в подпрограмме UPR 6000. После расчета токовых моделей линейных двухполюсников по подпрограмме UPR 4000 для нелинейных двухполюсников в со-

ответствии с уравнением (6.2-2) определяются начальные значения. Затем для каждого нелинейного двухполюсника на итерационном цикле в методе Ньютона рассчитываются свои дискретные токовые модели (строки 6032—6090).

При нелинейных проводимости и сопротивлении (управляемой проводимости) расчет токовых моделей осуществляется по программе NEWTON. Дискретная токовая модель нелинейного двухполюсного накопителя энергии рассчитывается в соответствии с алгоритмом, приведенным в п. 6.2.3. Текущее значение дифференциальной емкости ветви I запоминается в массиве C(I), а текущее значение дифференциальной проводимости — в массиве L(I).

После вычисления параметров ветви в подпрограмме UPR 4100 определяется целесообразность дальнейшего приближения. Остальные строки итерационного цикла (6110—6144) взяты из программы NEWTON (382—416). По окончании каждой итерации подпрограмма UPR 4300 перезаписывает в память ЭВМ параметры ветвей, содержащиеся в массиве ZG(IR, 11). При этом текущее значение параметра ветви заносится в элемент ZG(I, 1). Соответственно изменяются и находящиеся в массиве ZE(NZ, 11) значения нелинейных двухполюсных накопителей энергии. Затем текущие значения C_d или L_d каждого нелинейного двухполюсного накопителя энергии заносятся в элемент ZE(I, 1).

Анализ заканчивается выводом на периферийное устройство заданных параметров ветвей с помощью подпрограммы UPR 4300.

```
6000 REM: Анализ в момент времени t
6002 REM
6004 GOSUB 4000
6006 REM: Начальное значение каждого нелинейного двухпо-
6007 REM: лусника
6008 FOR I=1 TO NZ
6010 SR(I)=0
6012 SW(I)=0
6014 FOR J=1 TO KB
6016 SR(I)=SR(I)+AL(J)* ZG(I, J) *ZG(I, J)
6018 SW(I)=SW(I)+GA(J) * ZG(I, J)
6020 NEXT J
6022 NEXT I
6024 REM: Итерация метода Ньютона
6026 FOR KL=1 TO 50
6028 REM: Определение дискретной токовой модели
6030 REM: для каждого нелинейного двухполюсника
```

```

6032 FOR I=1 TO NZ
6034 IF NS(I)=0 THEN 6048
6036 XK=SW(I)
6038 GOSUB 5400
6040 GOSUB 5300
6042 IF IV=0 THEN 6148
6044 ON IA(I) GOTO 6058, 6066, 6074, 6084
6046 REM: Управляемая проводимость
6048 J=MG(I)
6050 GD(I)=SF(I) * (SD(J)+GD(J) * SW(J))
6052 SD(I)=0
6054 GOTO 6090
6056 REM: Нелинейная проводимость
6058 GD(I)=SK
6060 SD(I)=YK-SK * XK
6062 GOTO 6090
6064 REM: Нелинейное сопротивление
6066 GD(I)=1/SK
6068 SD(I)=XK-YK/SK
6070 GOTO 6090
6072 REM: Нелинейная емкость
6074 GD(I)=-AL(0) * YK/H
6076 SD(I)=-SR(I)/H
6078 C(I)=YK
6080 GOTO 6090
6082 REM: Нелинейная индуктивность
6084 GD(I)=-H/(AL(0) * YK)
6086 SD(I)=-SR(I)/(AL(0) * YK)
6088 L(I)=YK
6090 NEXT I
6092 GOSUB 4100
6094 REM: Вычисление параметров ветвей
6096 REM: нелинейных двухполюсников
6098 FOR I=1 TO NZ
6100 ON IA(I) GOTO 6102, 6106, 6102, 6106
6102 ZS(I)=UA(I)
6104 GOTO 6108
6106 ZS(I)=SA(I)
6108 NEXT I
6110 REM: Запрос, пригодно ли приближение
6112 FOR I=1 TO NZ
6114 IF NS(I)=0 THEN 6124
6116 IF ABS(SW(I))<1E-30 THEN 6122
6118 IF ABS((ZS(I)-SW(I))/SW(I))>EN THEN 6130
6120 GOTO 6124
6122 IF ABS(ZS(I))>EN THEN 6130
6124 NEXT I
6126 GOTO 6166
6128 REM: Новое запоминание параметров ветвей
6130 FOR I=1 TO NZ
6132 IF NS(I)=0 THEN 6142
6134 IF KL>4 THEN 6140
6136 SW(I)=ZS(I)
6138 GOTO 6142
6140 SW(I)=.5 * (ZS(I)+SW(I))

```

```

6142 NEXT I
6144 NEXT KL
6146 PRINT «Нет сходимости»
6148 PRINT «Выбрать новое начальное значение»
6150 FOR I=1 TO NZ
6152 IF NS(I)=0 THEN 6158
6154 PRINT «Ветвь»; I;
6156 INPUT «:», SW(I)
6158 NEXT
6160 GOTO 6026
6162 REM: Новое запоминание параметров состояния
6164 REM: нелинейных двухполюсных накопителей энергии
6166 FOR I=1 TO NZ
6168 FOR J=L1 TO STEP-1
6170 ZE(I, J), ZS(I, J-1)
6172 NEXT J
6174 ON IA(I) GOTO 6182, 6182, 6178, 6180
6176 ZE(I, 1)=C(I)
6178 GOTO 6182
6180 ZE(I, 1)=L(I)
6182 NEXT I
6184 GOSUB 4300
6186 RETURN

```

6.3.3. Анализ переходного процесса. Для его выполнения (строки 308—480) можно использовать программу DYNA-LIN с некоторыми изменениями. Вместо подпрограмм UPR 4000, 4100, 4300 в программе DYNA-NL каждый раз вызывается подпрограмма UPR 6000. Кроме того, при удвоении значения шага перезаписываются не только содержащиеся в массиве ZG(IR, 11) параметры ветвей, но и содержащиеся в массиве ZE (NZ, 11) значения нелинейных двухполюсных реактивных элементов (строки 420—424).

```

100 PRINT «DYNA-NL»
102 PRINT «Анализ переходных процессов»
104 PRINT «динамической цепи»
106 PRINT «с нелинейными двухполюсниками»
108 PRINT
110 DEFINT I—K, M—N
112 DEFDBL A—D, G, L, P—Z
114 REM
116 REM: Ввод глобальных параметров
118 INPUT «Количество ветвей»; IZ
120 INPUT «Количество узлов (без базового); KN
122 INPUT «Количество ид. ист. напряж.»; IS
124 PRINT
126 N=KN-IS
128 IF IZ<IS THEN 132
130 IF N>0 THEN 138
132 PRINT «Ввод не имеет смысла»
134 PRINT «или решение невозможно»

```

```

136 GOTO 118
138 IR=IZ-IS
140 DIM UN(KN), UV(KN), KF(KN), A(N, N), X(N)
142 DIM AL(6), GA(6)
144 DIM SF(IR), MZ(IR), VR(IR), SR(IR), ZG(IR, 11)
146 DIM KV(IR), KZ(IR), GD(IR), SD(IR), SA(IR), UA(IR)
148 DIM G(IR), L(IR), C(IR), MI(IR), MU(IR), IA(IR)
150 DIM IT(IZ), ZF(IZ), VF(IZ)
152 DIM KA(IS), KU(IS), QR(IS), QT(IS)
154 REM
156 REM: Ввод нелинейных двухполюсников
158 INPUT «Количество нелинейных двухполюсников»; NZ
160 IF NZ<1 THEN 158
162 DIM MG(NZ), NS(NZ), SW(NZ), ZE(NZ, 11), ZS(NZ)
164 NM=3
166 FOR I=1 TO NZ
168 PRINT «Двухполюсник»; I;
170 INPUT «Тип двухполюсника (G/R/C/L)»; FR$
172 IF FR$=«G» THEN 182
174 IF FR$=«R» THEN 186
176 IF FR$=«C» THEN 190
178 IF FR$=«L» THEN 194
180 GOTO 170
182 IA(I)=1
184 GOTO 186
186 IA(I)=2
188 GOTO 196
190 IA(I)=3
192 GOTO 196
194 IA(I)=4
196 INPUT «Количество опорных точек»; NS(I)
198 IF NS(I)=0 THEN 206
200 IF NS(I)<3 THEN 196
202 IF NS(I)<NM THEN 206
204 NM=NS(I)
206 NEXT I
208 NI=NM-1
210 DIM XS(NM, NZ) YS(NM, NZ), BS(NM, NZ)
212 DIM AS(NI, NZ), CS(NI, NZ), SE(NI), SH(NI)
214 DIM DH(NI), DN(NI), RS(NI)
216 FOR I=1 TO NZ
218 PRINT «Ветвь»; I
220 INPUT «из узла»; KV(I)
222 INPUT «к узлу»; KZ(I)
224 IF NS(I)>0 THEN 236
226 PRINT «Проводимость, управляемая током»
228 INPUT «Коэффициент управления»; SF(I)
230 INPUT «Ветвь управления»; MG(I)
232 GOTO 288
234 PRINT
236 PRINT «Ввод опорных точек»
238 NI=NS(I)-1
240 PRINT «Двухполюсник»; I
242 FOR J=1 TO NS(I)
244 PRINT «X»; J;

```

```

246 INPUT «;»; XS(J, I)
248 PRINT «Y»; J;
250 INPUT «;»; P; YS(J, I)
252 NEXT
254 INPUT «Граничные условия»; FR$
256 IF FR$ = «A» THEN 262
258 IF FR$ = «S» THEN 266
260 GOTO 254
262 GOSUB 5100
264 GOTO 268
266 GOSUB 5200
268 PRINT «Начальное значение»
270 ON IA(I) GOTO 272, 276, 272, 276
272 INPUT «напряжения»; ZG(I, 1)
274 GOTO 278
276 INPUT «тока»; ZG(I, 1)
278 ON IA(I) GOTO 288, 288, 280, 280
280 XK = ZG(I, 1)
282 GOSUB 5300
284 IF IV = 0 THEN 168
286 ZE(I, 1) = YK
288 NEXT I
290 REM
292 REM: Ввод линейных двухполюсников
294 IG = 0
296 FOR I = NZ + 1 TO IZ
298 PRINT «Ветвь»; I
300 GOSUB 4500
302 NEXT I
304 REM
306 REM: Определение массива узлов
308 GOSUB 2100
310 REM
312 REM: Определение выводимых параметров
314 GOSUB 8400
316 REM
318 REM: Ввод параметров управления
320 PRINT
322 INPUT «Начальное значение шага HA»; HA
324 INPUT «Максимальное значение шага HE»; HE
326 INPUT «Максимальное значение порядка KE»; KE
328 IF KE < 2 THEN 334
330 IF KE > 6 THEN 334
332 GOTO 340
334 PRINT «Ввод не имеет смысла»
336 PRINT «повторите, пожалуйста»
338 GOTO 326
340 INPUT «Конечное значение времени TE»; TE
342 INPUT «Значение предела»; EB
344 PRINT
346 REM
348 REM: При значении шага, меньшем H-HA
350 REM: Значение порядка KB возрастает от 1 до KE;
352 REM: для каждого значения KB цепь анализируется
354 REM: в моменты времени MT

```

```

356 MT=10
358 UN(0)=0
360 EN=.00001
362 T=0
364 H=HA
366 JF=0
368 FOR KB=1 TO KE
370 GOSUB 4000
372 FOR ML=1 TO MT
374 T=T+H
376 GOSUB 6004
378 NEXT ML
380 NEXT KB
382 KB=KE
384 REM
386 REM: Для каждого значения MT рассчитывается
388 REM: и заданным образом удваивается значение шага
390 JF=1
392 DX=0
394 SX=0
396 FOR ML=1 TO MT
398 T=T+H
400 IF T>(TE+HE) THEN 446
402 GOSUB 6004
404 NEXT ML
406 IF DX/SX>EB THEN 392
408 H=2 * H
410 FOR J=1 TO 6
412 JD=2 * J-1
414 FOR I=1 TO IR
416 ZG(I, J)=ZG (I, JD)
418 NEXT I
420 FOR I=1 TO NZ
422 ZE(I, J)=ZE(I, JD)
424 NEXT I
426 NEXT J
428 IF H<HE THEN 392
430 REM
432 REM: Дальнейшие вычисления выполняются с максималь-
434 REM: ным значением шага вплоть до конечного времени
436 JF=0
438 T=T+H
440 GOSUB 6004
442 IF T<TE THEN 438
444 REM
446 PRINT «Вычисления»
448 INPUT «продолжать? (J/N)»; FR$
450 IF FR$=«J» THEN 460
452 IF FR$=«N» THEN 9999
454 GOTO 446
456 REM
458 REM. Изменение параметров ветви
460 PRINT
462 PRINT «В скольких ветвях с линейными двухполюсниками»
464 INPUT «необходимо изменить элементы?»; ME

```



```

466 IF ME=1 THEN 9999 9999
468 FOR ML=1 TO ME
470 INPUT «Ветвь»: 1
472 IF I<(NZ+1) THEN 476
474 GOSUB 4500
476 NEXT
478 GOTO 308
1000 REM: UPR 2000, 2100, 2200, 2300, 2400
1002 REM: UPR 2600, 2700, 2800, 2900, 3400
1004 REM: UPR 4000, 4100, 4300, 4400, 4500, 4600, 4800
1006 REM: UPR 5000, 5100, 5200, 5300, 5400, 6000

```

6.3.4. Пример. Рассчитаем выходное напряжение однопериодного выпрямителя (рис. 6.8). Диод D_1 пред-

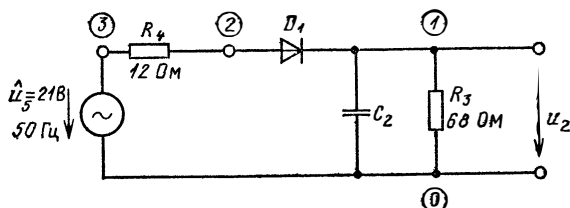


Рис. 6.8. Схема однопериодного выпрямителя, нагруженного активным сопротивлением

ставлен в схеме нелинейной проводимостью (тип двухполюсника G), а для его вольт-амперной характеристики вводятся шесть опорных точек. В момент времени $t=0$ синусоидальное напряжение идеального источника имеет амплитудное значение $\hat{u}_5 = 21$ В, а конденсатор разряжен (начальное значение напряжения $u_2 = 0$).

Сначала вычислим выходное напряжение u_2 для случая, когда $C_2 = 470$ мкФ = const.

Программа DYNA-NL

Анализ переходных процессов динамической цепи с нелинейными двухполюсниками

Количество ветвей? 5

Количество узлов (без базового)? 3

Количество идеальных источников напряжения? 1

Количество нелинейных двухполюсников? 1

Двухполюсник 1. Тип двухполюсника (G/R/C/L)? G

Количество опорных точек? 6

Ветвь 1

из узла? 2

к узлу? 1

Ввод опорных точек

Двухполюсник 1

X1: — 1000

У1:? — 1,Е—5

Х2:? .6

У2:? .6 Е—8

Х3:? .7

У3:? .04

Х4:? .8

У4:? .3

Х5:? .9

У5:? .1

Х6:? 1

У6:? 2

Граничные условия (а/б):? а

Начальное значение напряжения? .7

Ветвь 2

из узла? 1

к узлу? 0

Тип ветви? С

Емкость? .47Е—3

Начальное значение? 0

Ветвь 3

из узла? 1

к узлу? 0

Тип ветви? R

Сопротивление? 68

Тип источника ? N

Ветвь 4

из узла? 3

к узлу? 2

Тип ветви? R

Сопротивление? 12

Тип источника? N

Ветвь 5

из узла? 3

к узлу? 0

Напряжение источника? 21

Зависимость от времени (K/L/S)? S

Круговая частота? 314.15

Начальный угол фазы? 0

Какие параметры ветвей необходимо выводить?

Количество напряжений в ветви? 1

Номера ветвей

? 2

Количество токов в ветвях? 0

Начальное значение шага НА? .00005

Максимальное значение шага HE? .0003

Максимальное значение порядка КК? 5

Конечное значение времени TE? .184

Значение эпсилон? .02

t=.0001 c;	U2= .349641 B
t= 00015 c;	U2= .5216318 B
t=.0002 c;	U2= .691702 B
t=.00025 c;	U2= .8598261 B
t=.0003 c;	U2=1.025979 B
t=.00035 c;	U2=1.190137 B
t=.0004 c;	U2=1.352274 B

Через 160 мс после включения цепи переходный процесс в ней заканчивается и параметры цепи в течение 20 мс имеют неизменные значения.

Ниже распечатаны значения выходного напряжения в области второго максимума напряжения u_2 :

$t = .1612$ с;	$U_2 = 12.6092$ В
$t = .1616$ с;	$U_2 = 12.83615$ В
$t = .162$ с;	$U_2 = 12.96031$ В
$t = .1624$ с;	$U_2 = 12.97105$ В
$t = .1628$ с;	$U_2 = 12.86581$ В
$t = .1632$ с;	$U_2 = 12.69749$ В
$t = .1812$ с;	$U_2 = 12.60883$ В
$t = .1816$ с;	$U_2 = 12.83591$ В
$t = .182$ с;	$U_2 = 12.96024$ В
$t = .1824$ с;	$U_2 = 12.97115$ В
$t = .1828$ с;	$U_2 = 12.86601$ В
$t = .1832$ с;	$U_2 = 12.6977$ В

Во второй части примера заменим в схеме на рис. 6.8 конденсатор с постоянной емкостью C_2 электролитиче-

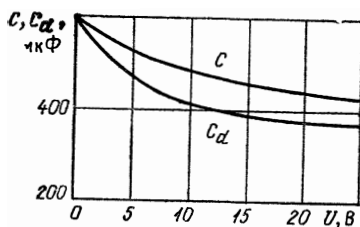


Рис. 6.9. Емкость и дифференциальная емкость электролитического конденсатора в зависимости от постоянного напряжения U

ским конденсатором с емкостью $C_2 = f(U_2)$ и рассчитаем дифференциальную емкость C_{d2} (рис. 6.9) последнего с помощью программы DIFF-KAP. Входные значения для второй ветви распечатаны ниже; остальные входные значения остались без изменения.

```

Ветвь 2
из узла? 1
к узлу? 0
Ввод опорных точек
Двухполюсник 2
X1: 0
U1: .6 E-3
X2: 3
U2: .51 E-3

```

X3: 6
 Y3: 46 E-3
 X4: 10
 Y4: 417 E-3
 X5: 15
 Y5: 385 E-3
 X6: 25
 Y6: 37 E-3
 Граничные условия (а/б): а
 Начальное значение напряжения? 0

Полученные протоколы расчета результатов после включения цепи:

$t = .00005$ с;	$U_2 = .139043$ В
$t = .0001$ с;	$U_2 = .139043$ В
$t = .00015$ с;	$U_2 = .4200586$ В
$t = .0002$ с;	$U_2 = .5620112$ В
$t = .00025$ с;	$U_2 = .7049025$ В
$t = .0003$ с;	$U_2 = .8487032$ В

и значения напряжения для установившегося режима в области второго максимума напряжения:

$t = .1616$ с;	$U_2 = 13.67977$ В
$t = .162$ с;	$U_2 = 13.76249$ В
$t = .1624$ с;	$U_2 = 13.68658$ В
$t = .1628$ с;	$U_2 = 13.47911$ В
$t = .1632$ с;	$U_2 = 13.2189$ В
$t = .1636$ с;	$U_2 = 12.96975$ В
$t = .1816$ с;	$U_2 = 13.67952$ В
$t = .182$ с;	$U_2 = 13.76247$ В
$t = .1824$ с;	$U_2 = 13.6868$ В
$t = .1828$ с;	$U_2 = 13.47946$ В

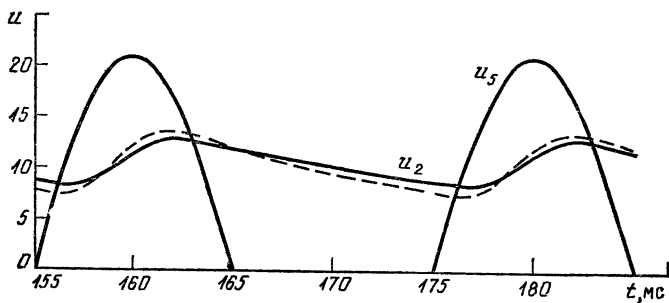


Рис. 6.10. Выходная характеристика выпрямителя, представленного на рис. 6.8:

— $C_2 = 470$ мкФ-const; - - - - $C_2 = f(U)$

$$t = .1832 \text{ с;}$$

$$U_2 = 13.21926 \text{ В}$$

$$t = .1836 \text{ с;}$$

$$U_2 = 12.97007 \text{ В}$$

На рис. 6.10 показана диаграмма выходного напряжения u_2 в установившемся режиме для схемы на рис. 6.3, в.

Глава 7

ПРОГРАММЫ НА ЯЗЫКЕ ПАСКАЛЬ

Поскольку Турбо-ПАСКАЛЬ в отличие от БЕЙСИКА позволяет обрабатывать более сложные программы, все описанные виды анализа цепей можно объединить в одной программе NWA, имеющей две версии. Версия 1.0 (исходные данные для расчета цепей вводятся с клавиатуры ПЭВМ) рассмотрена в этой главе, а версия 2.0, разработанная для ПЭВМ, совместимых с ЭВМ фирмы IBM (исходные данные для цепей расположены в файле на дискете), подробно изложена в приложении Пб.

7.1. Основная программа

После запуска программа NWA выдает на экран «меню» выбора режимов, которое выглядит следующим образом: G — анализ цепей постоянного тока; S — анализ линейных цепей синусоидального напряжения; D — анализ динамических цепей (переходных процессов). Ключевые буквы программно определяют значение переменной `progart`. С помощью этой переменной выбираются три процедуры: G (процедура `gleich`); S (процедура `sinus`) или D (процедура `dyna`).

После выполнения выбранной процедуры, т. е. анализа заданного типа, основная программа может быть запущена снова.

7.2. Ввод исходных данных с клавиатуры

С помощью программы NWA (версия 1.0) на ПЭВМ можно проанализировать цепь, состоящую максимум из 140 двухполюсников. Количество узлов цепи (без базового) равно $60 + is$, причем число идеальных источников напряжения $is \leq 30$. В первую очередь вводятся глобальные величины (процедура `einglob`), затем элементы не-

линейных ветвей, сами ветви и наконец — идеальные источники напряжения.

Для линейных цепей синусоидального напряжения ($\text{progart} = 'S'$) нелинейные ветви отсутствуют. Ветви нумеруются точно так же, как и в программе на БЕЙСИКе, т. е. нелинейные двухполюсники получают наименьшие номера ветвей. Последними вводятся идеальные источники напряжения, которые получают наибольшие номера. В отличие от программы на БЕЙСИКе здесь каждая ветвь может содержать только один двухполюсник. Тип двухполюсника обозначается определенной буквой и запоминается в массиве `zweigart [i]`.

Для расчета нелинейных двухполюсников используются следующие типы ветвей:

проводимость — `zweigart [i] = 'G'` (`gleich; dyna`);

сопротивление — `zweigart [i] = 'R'` (`gleich; dyna`);

емкость — `zweigart [i] = 'C'` (`dyna`);

индуктивность — `zweigart [i] = 'L'` (`dyna`).

После ввода опорных точек необходимо дополнительно задать граничные условия *a* (кубическая интерполяция в граничных интервалах) или *b* (линейная интерполяция в граничных интервалах), а также начальное значение напряжения или тока.

Для расчета линейных двухполюсников применяются следующие типы ветвей:

(активная) проводимость — `zweigart [i] = 'G'` (`gleich; sinus; dyna`);

активное сопротивление — `zweigart [i] = 'R'` (`gleich; dyna; sinus`);

емкость — $C = \text{const}$ — `zweigart [i] = 'C'` (`sinus; dyna`);

индуктивность — $L = \text{const}$ — `zweigart [i] = 'L'` (`sinus; dyna`);

идеальный источник тока — `zweigart [i] = 'I'` (`gleich; sinus; dyna`);

управляемый источник тока — `zweigart [i] = 'S'` (`gleich; sinus; dyna`);

управляемая проводимость — `zweigart [i] = 'T'` (`gleich; dyna`).

В качестве параметра управления нелинейной проводимостью используется ток. Таким образом, управляемая проводимость может обрабатываться в режимах *G* (`gleich`) и *D* (`dyna`). Для динамических цепей (переходный процесс) временная зависимость каждого идеального источника задается с помощью процедуры

einzeit [i]. Тип этой зависимости обозначается определенной буквой и запоминается в элементе массива zeitabh [i]:

К — параметр источника постоянен во времени;

L — параметр источника линейно зависит от времени;

S — параметр источника синусоидально зависит от времени.

Кроме того, с помощью процедуры einzeit [i] вводятся также круговая частота m и начальный угол фазы s .

7.3. Цепь постоянного тока

Для анализа цепей постоянного тока вызывается процедура gleich, которая позволяет рассчитать как линейные ($pz=0$), так и нелинейные цепи. После ввода элементов ветвей, фиксации выводимых величин (процедура festlaus) и заполнения массива узлов (процедура knotfeld) начинается интерполяция метода Ньютона.

В линейных цепях ввиду того, что $pz=0$, операторы процедур dis_nlin и zweig_nlin не выполняются. То же самое происходит и при вызове последней процедуры. Сначала рассчитывается дискретная токовая модель каждой ветви (процедура dismod), а затем составляется и решается система линейных уравнений (процедура aufloes). Если система уравнений не может быть решена, булева переменная singul устанавливается в состояние false. При этом вычисление и выдача параметров ветвей могут не выполняться. В противном случае реализуются процедуры knoten, zweig и ausfest.

В нелинейных цепях после каждой итерации запрашивается, остаются ли параметры нелинейных ветвей по сравнению с предыдущими вычислительными шагами практически неизменными; если да — то вычисляются и выводятся все параметры ветвей; если нет — то после вычисления новых начальных значений вычисляются следующие итерационные шаги метода Ньютона. После ввода параметров ветвей и новых значений тока и напряжения расчет повторяется.

7.4. Линейная цепь

Для анализа линейных цепей синусоидального напряжения вызывается процедура sinus (нелинейные двухполюсники не рассматриваются). При использовании

комплексных вычислений предполагается, что все идеальные источники синусоидального напряжения или тока, а соответственно и все параметры ветвей имеют одинаковую частоту f . Ее значение в программе изменяется с помощью специального оператора.

Анализ цепи для каждого значения частоты является комплексной версией анализа линейной цепи постоянного тока.

7.5. Динамическая цепь

Для анализа динамической цепи вызывается процедура `dyna`, причем рассчитываются как линейные ($pz=0$), так и нелинейные цепи. После ввода элементов ветвей задаются управляющие параметры. Анализ выполняется с начальным значением шага. При этом порядок kb метода обратного дифференцирования увеличивается от 1 до ke . Для каждого значения порядка цепь анализируется в определенные моменты времени (обычно $mt=10$). Если достигается kb_{max} , то цепь рассматривается для каждого из десяти моментов. При выполнении условия (4.4-25) шаг удваивается до тех пор, пока не станет максимальным. Затем вычисления продолжаются.

7.6. Распечатка программы

Ниже приводится распечатка программы NWA (версия 1.0) на ПАСКАЛЕ. Поскольку в Турбо-ПАСКАЛЕ каждая функция и процедура определяются до вызова программы, главная программа располагается в конце распечатки.

В приложении П4 указаны страницы, на которых находятся функции и процедуры программы NWA.

Переменные программы на языке ПАСКАЛЬ, имена которых состоят из одной или двух букв, совпадают с переменными на языке БЭЙСИК (см. приложение П3). Остальные переменные программы NWA описаны в приложении П5.


```

PROGRAM NWA; { Программа анализа цепей }
LABEL Start; { Версия 1.0:14.10.1986 }
CONST epsilon=1E-20;
      en=0.00001;
      ismax=30;
      izmax=140;
      knmax=60;
      nsmax=20;
      nzmax=30;
VAR   i, il, ir, is, iv, iz, j, k, kb, ke, kl, kn,
      ma, mb, me, ml, mt, n, ni, nt, nz: INTEGER;
      dx, eb, h, ha, he, sk, sx, t, te, v, xk, yk: REAL;
      sqq, jf, pruefbed, singul: BOOLEAN;
      ch, progart: CHAR;
      al, ga: ARRAY[0..6] of REAL;
      kf: ARRAY[0..knmax] of INTEGER;
      un, ui, uv, x, y: ARRAY[0..knmax] of REAL;
      ns: ARRAY[1..nzmax] of INTEGER;
      sv, zs: ARRAY[1..nzmax] of REAL;
      ze: ARRAY[1..nzmax, 1..11] of REAL;
      randbed: ARRAY[1..nzmax] of CHAR;
      as, bs, cs, xs, ys: ARRAY[1..nsmax, 1..nzmax] of REAL;
      se, sh, dh, dn, rs: ARRAY[1..nsmax] of REAL;
      ka, ku: ARRAY[1..ismax] of INTEGER;
      qr, qi, qt: ARRAY[1..ismax] of REAL;
      kv, kz, mi, mu, mz: ARRAY[1..izmax] of INTEGER;
      G, L, C, gd, gi, sa, sb, sd, sf, si, sr,
      ua, ub, vf, vi, vr, zf: ARRAY[1..izmax] of REAL;
      zg: ARRAY[1..izmax, 1..11] of REAL;
      zeitabh, zweigart: ARRAY[1..izmax] of CHAR;

FUNCTION sgn(a: REAL): INTEGER;
BEGIN
  IF a>0 THEN sgn:=1 ELSE sgn:=-1;
  IF a=0 THEN sgn:=0
END;

FUNCTION betrag(a,b: REAL): REAL;
BEGIN
  betrag:=sqrt(a*a+b*b)
END;

FUNCTION wingrad(a,b: REAL): REAL;
VAR   hv: REAL;
BEGIN
  hv:=90*sgn(b);
  IF a=0 THEN wingrad:=hv ELSE
  wingrad:=90*(1-sgn(a))*sgn(hv+1)+180*Arctan(b/a)/pi
END;

```

```

PROCEDURE knotfeld; { Заполнение массива узлов }
VAR bed: BOOLEAN;
BEGIN
kf[0]:=0;
FOR i:=1 TO kn DO
BEGIN
bed:=true;
FOR k:=1 TO is DO IF ka[k]=i THEN bed:=false;
IF bed THEN kf[i]:=kf[i-1]+1 ELSE kf[i]:=kf[i-1];
END;
FOR k:=1 TO is DO kf[ka[k]]:=kf[ku[k]];
END; {knotfeld}

```

```

PROCEDURE vorver; { Подготовка размещения идеальн, }
BEGIN { источников напряжения }
FOR i:=1 TO ir DO
BEGIN
vr[i]:=0;
FOR il:=1 TO is DO
BEGIN
IF kv[i]=ka[il] THEN vr[i]:=vr[i]+qt[il];
IF kz[i]=ka[il] THEN vr[i]:=vr[i]-qt[il];
END;
END;
END; {vorver}

```

```

PROCEDURE kvorver; { Подготовка размещ. ид. источн. }
BEGIN { напряж. в комплексн. форме }
FOR i:=1 TO ir DO
BEGIN:
vr[i]:=0;
vi[i]:=0;
FOR il:=1 TO is DO
BEGIN
IF kv[i]=ka[il] THEN
BEGIN
vr[i]:=vr[i]+qr[il];
vi[i]:=vi[i]+qi[il];
END;
IF kz[i]=ka[il] THEN
BEGIN
vr[i]:=vr[i]-qr[il];
vi[i]:=vi[i]-qi[il];
END;
END;
END;
END; {kvorver}

```

```

PROCEDURE aufloes; { Сост. и реш.сист.лин.ур-ний }
VAR j1, jr, js, kr, ks, ms: INTEGER;
    al, dl, gl, rl, sl, xl: REAL;
    a: ARRAY[0..knmax, 0..knmax] of REAL;
BEGIN
FOR i:=0 TO n DO BEGIN x[i]:=0; FOR j:=0 TO n+
FOR i:=1 TO ir DO          → DO a[i,j]:=0 END;
BEGIN
    sl:=sd[i]+gd[i]*vr[i];
    jr:=kf[kv[i]];
    kr:=kf[kz[i]];
    IF jr<>kr THEN
    BEGIN
        a[jr, jr]:=a[jr, jr]+gd[i];
        a[jr, kr]:=a[jr, kr]-gd[i];
        a[kr, jr]:=a[kr, jr]-gd[i];
        a[kr, kr]:=a[kr, kr]+gd[i];
        IF zweigart[i]='S' THEN{ упр.источники тока }
        BEGIN
            ms:=abs(mz[i]);
            IF mz[i]>0 THEN BEGIN gl:=sf[i];
                                sl:=sl+sf[i]*vr[ms] END
                            ELSE BEGIN gl:=sf[i]*gd[ms];
                                sl:=sl+gl+vr[ms]+sf[i]*sd[ms] END;
            js:=kf[kv[ms]];
            ks:=kf[kz[ms]];
            a[jr, js]:=a[jr, js]+gl;
            a[jr, ks]:=a[jr, ks]-gl;
            a[kr, js]:=a[kr, js]-gl;
            a[kr, ks]:=a[kr, ks]+gl
        END;
        x[jr]:=x[jr]-sl;
        x[kr]:=x[kr]+sl;
    END;
END;
IF agq THEN
BEGIN { Построение упорядочение системы }
    singul:=true; { линейных уравнений }
    FOR i:=1 TO n DO
    BEGIN
        IF abs(a[i, i])<epsilon THEN
        BEGIN
            j1:=0; r1:=0;
            FOR j:=1 TO n DO
            BEGIN
                al:=abs(a[i, j]*a[j, i]);
                IF al>r1 THEN BEGIN r1:=al; j1:=j; END;
            END;
            IF r1<epsilon THEN singul:=false;

```

```

    IF singul THEN
    BEGIN
        x1:=x[j1]; x[j1]:=x[i1]; x[i1]:=x1;
        FOR j:=1 TO n DO
            BEGIN a1:=a[j1,j]; a[j1,j]:=a[i,j]; a[i,j]→
            END; →:=a1; END;
        END;
    END;
    FOR i:=1 TO n-1 DO { Сокращение }
    BEGIN
        d1:=a[i,i];
        IF abs(d1)<epsilon THEN singul:=false;
        IF singul THEN
        BEGIN
            FOR j:=i+1 TO n DO
            BEGIN
                r1:=a[j,i]/d1;
                IF r1=0 THEN
                    ELSE BEGIN
                        FOR k:=i+1 TO n DO a[j,k]:=a[j,k]-r1*a[i,k];
                        x[j]:=x[j]-r1*x[i]
                    END;
                END;
            END;
        END;
    END;
    END ELSE
    BEGIN { Сокращение сист. лин. ур-ний с симметр. }
        singul:=true; { матрицей коэффициентов }
        FOR i:=1 TO n-1 DO
        BEGIN
            d1:=a[i,i];
            FOR j:=i+1 TO n DO
            BEGIN
                r1:=a[i,j]/d1;
                IF r1=0 THEN
                    ELSE BEGIN
                        FOR k:=j TO n DO a[j,k]:=a[j,k]-r1*a[i,k];
                        x[j]:=x[j]-r1*x[i]
                    END;
                END;
            END;
        END;
    END;
    BEGIN { Обратная подстановка }
    IF singul THEN
    BEGIN
        FOR j:=n DOWNT0 1 DO
        BEGIN
            d1:=x[j];
            FOR k:=n DOWNT0 j+1 DO d1:=d1-a[j,k]*x[k];

```

```

        IF abs(a[j, j]) < epsilon THEN singul := false;
        IF singul THEN x[j] := d1/a[j, j]
    END;
END;
END;
END;    {aufloes}

PROCEDURE kaufloes;    {Составление и реш. системы}
VAR j1, jr, js, kr, ks, ms: INTEGER; {компл. ур-ний}
    a1, b1, b2, d1, d2, d3, e2,
    g1, g2, r1, r2, r3, s1, s2, x1, y1: REAL;
    a, b: ARRAY[0..knmax, 0..knmax] of REAL;
BEGIN
    e2 := epsilon * epsilon;
    FOR i := 1 TO n DO
    BEGIN
        x[i] := 0;
        y[i] := 0;
        FOR j := 1 TO n DO
        BEGIN
            a[i, j] := 0;
            b[i, j] := 0
        END;
    END;
    FOR i := 1 TO ir DO
    BEGIN
        gd[i] := G[i];
        gi[i] := w * C[i];
        IF L[i] > 0 THEN gi[i] := gi[i] - 1 / (w * L[i]);
        s1 := sr[i] + gd[i] * vr[i] - gi[i] * vi[i];
        s2 := si[i] + gi[i] * vr[i] + gd[i] * vi[i];
        jr := kf[kv[i]];
        kr := kf[kz[i]];
        IF jr <> kr THEN
        BEGIN
            a[jr, jr] := a[jr, jr] + gd[i];
            b[jr, jr] := b[jr, jr] + gi[i];
            a[jr, kr] := a[jr, kr] - gd[i];
            b[jr, kr] := b[jr, kr] - gi[i];
            a[kr, jr] := a[kr, jr] - gd[i];
            b[kr, jr] := b[kr, jr] - gi[i];
            a[kr, kr] := a[kr, kr] + gd[i];
            b[kr, kr] := b[kr, kr] + gi[i];
            IF zweigart[i] = 'S' THEN
            BEGIN
                ms := abs(mz[i]);
                IF mz[i] > 0 THEN
                BEGIN
                    g1 := sf[i];

```

```

      g2:=s0;
      s1:=s1+sf[i]*vr[ms];
      s2:=s2+sf[i]*vi[ms]
    END ELSE
    BEGIN
      g1:=sf[i]*gd[ms];
      g2:=sf[i]*gi[ms];
      s1:=s1+g1*vr[ms]-g2*vi[ms]+sf[i]*sr[ms];
      s2:=s2+g2*vr[ms]+g1*vi[ms]+sf[i]*si[ms]
    END;
    js:=kf(kv[ms]);
    ks:=kf(kz[ms]);
    a[jr,js]:=a[jr,js]+g1;
    b[jr,js]:=b[jr,js]+g2;
    a[jr,ks]:=a[jr,ks]-g1;
    b[jr,ks]:=b[jr,ks]-g2;
    a[kr,js]:=a[kr,js]-g1;
    b[kr,js]:=b[kr,js]-g2;
    a[kr,ks]:=a[kr,ks]+g1;
    b[kr,ks]:=b[kr,ks]+g2;
  END;
  x[jr]:=x[jr]-s1;
  y[jr]:=y[jr]-s2;
  x[kr]:=x[kr]+s1;
  y[kr]:=y[kr]+s2
END;
END;
IF agq THEN
BEGIN { Упорядочение системы комплексн. }
  singul:=true; { уравнений }
  FOR i:=1 TO n DO
  BEGIN
    IF betrag(a[i,1],b[i,1])<e2 THEN
    BEGIN '
      j1:=0; r1:=0;
      FOR j:=1 TO n DO
      BEGIN
        b1:=betrag(a[i,j],b[i,j]);
        b2:=betrag(a[j,1],b[j,1]);
        a1:=b1*b2;
        IF a1>r1 THEN BEGIN r1:=a1; j1:=j; END;
      END;
      IF r1<epsilon THEN singul:=false;
      IF singul THEN
      BEGIN
        x1:=x[j1]; x[j1]:=x[i]; x[i]:=x1;
        y1:=y[j1]; y[j1]:=y[i]; y[i]:=y1;
        FOR j:=1 TO n DO
        BEGIN

```

```

        a1:=a[j1,j]; a[j1,j]:=a[i,j]; a[i,j]:=a1;
        b1:=b[j1,j]; b[j1,j]:=b[i,j]; b[i,j]:=b1;
    END;
END;
END;
END;
FOR i:=1 TO n-1 DO
BEGIN
    { Сокращение системы }
    d1:=a[i,i]; d2:=b[i,i]; { комплексн. ур-ний }
    d3:=d1*d1+d2*d2;
    IF d3<e2 THEN singul:=false;
    IF singul THEN
    BEGIN
        FOR j:=i+1 TO n DO
        BEGIN
            r1:=(a[j,i]*d1+b[j,i]*d2)/d3;
            r2:=(b[j,i]*d1-a[j,i]*d2)/d3;
            FOR k:=i+1 TO n DO
            BEGIN
                a[j,k]:=a[j,k]-r1*a[i,k]+r2*b[i,k];
                b[j,k]:=b[j,k]-r1*b[i,k]-r2*a[i,k];
            END;
            x[j]:=x[j]-r1*x[i]+r2*y[i];
            y[j]:=y[j]-r1*y[i]-r2*x[i];
        END;
    END;
    { Сокращение }
END ELSE
BEGIN
    { Сокращение системы комплексн.ур-ний }
    singul:=true; { с симметричной матрицей }
    FOR i:=1 TO n-1 DO
    BEGIN
        d1:=a[i,i];
        d2:=b[i,i];
        d3:=d1*d1+d2*d2;
        FOR j:=i+1 TO n DO
        BEGIN
            r1:=(a[i,j]*d1+b[i,j]*d2)/d3;
            r2:=(b[i,j]*d1-a[i,j]*d2)/d3;
            FOR k:=j TO n DO
            BEGIN
                a[j,k]:=a[j,k]-r1*a[i,k]+r2*b[i,k];
                b[j,k]:=b[j,k]-r1*b[i,k]-r2*a[i,k];
            END;
            x[j]:=x[j]-r1*x[i]+r2*y[i];
            y[j]:=y[j]-r1*y[i]-r2*x[i];
        END;
    END;
END;
END;
END;

```

```

IF singul THEN
BEGIN
    { Обратная подстановка }
    FOR j:=n DOWNT0 1 DO
    BEGIN
        d1:=x[j];
        d2:=y[j];
        FOR k:=n DOWNT0 j+1 DO
        BEGIN
            d1:=d1-a[j,k]*x[k]+b[j,k]*y[k];
            d2:=d2-b[j,k]*x[k]-a[j,k]*y[k];
        END;
        r1:=a[j,j];
        r2:=b[j,j];
        r3:=r1*r1+r2*r2;
        IF r3<e2 THEN
            singul:=false;
        IF singul THEN
        BEGIN
            x[j]:=(d1*r1+d2*r2)/r3;
            y[j]:=(d2*r1-d1*r2)/r3;
        END;
    END;
END;
END;
END;    {kaufloes}

```

```

PROCEDURE knoten; { Расчет узловых напряжений }
BEGIN
    FOR j:=1 TO kn DO un[j]:=x[kf[j]];
    un[0]:=0;
    FOR il:=1 TO is DO un[ka[il]]:=un[ku[il]]+qt[il];
END;    {knoten}

```

```

PROCEDURE kknoten; { Расчет комплексных узловых
                    { напряжений }
BEGIN
    FOR j:=1 TO kn DO
    BEGIN
        un[j]:=x[kf[j]];
        ui[j]:=y[kf[j]];
    END;
    un[0]:=0;
    ui[0]:=0;
    FOR il:=1 TO is DO
    BEGIN
        un[ka[il]]:=un[ku[il]]+qr[il];
        ui[ka[il]]:=ui[ku[il]]+qi[il];
    END;
END;
END;    {kknoten}

```



```

PROCEDURE zweig; {Расчет напряжений и токов ветви }
BEGIN
FOR i:=1 TO ir DO
BEGIN
ua[i]:=un[kv[i]]-un[kz[i]];
sa[i]:=ua[i]*gd[i]+sd[i];
IF zweigart[i]='S' THEN
BEGIN IF mz[i]>0 THEN sa[i]:=sf[i]+ua[mz[i]]
ELSE sa[i]:=sf[i]+sa[-mz[i]] END;
END;
FOR i:=1 TO is DO
BEGIN
j:=i+ir;
ua[j]:=qt[i];
sa[j]:=0;
FOR k:=1 TO ir DO
BEGIN
IF ka[i]=kv[k] THEN sa[j]:=sa[j]-sa[k];
IF ka[i]=kz[k] THEN sa[j]:=sa[j]+sa[k]
END
END;
END; {zweig}

```

```

PROCEDURE kzweig; {Расчет комплексных параметров }
                     { ветви }
BEGIN
FOR i:=1 TO ir DO
BEGIN
ua[i]:=un[kv[i]]-un[kz[i]];
ub[i]:=ui[kv[i]]-ui[kz[i]];
sa[i]:=ua[i]*gd[i]-ub[i]*gi[i]+sr[i];
sb[i]:=ub[i]*gd[i]+ua[i]*gi[i]+si[i];
IF zweigart[i]='S' THEN
BEGIN
IF mz[i]>0 THEN BEGIN sa[i]:=sf[i]+ua[mz[i]];
sb[i]:=sf[i]+ub[mz[i]]END
ELSE BEGIN sa[i]:=sf[i]+sa[-mz[i]];
sb[i]:=sf[i]+sb[-mz[i]]END
END;
END;
FOR i:=1 TO is DO
BEGIN
j:=i+ir;
ua[j]:=qr[i];
ub[j]:=qi[i];
sa[j]:=0;
sb[j]:=0;
FOR k:=1 TO ir DO
BEGIN
IF ka[i]=kv[k] THEN

```

```

BEGIN
  sa[j]:=sa[j]-sa[k];
  sb[j]:=sb[j]-sb[k]
END;
IF ka[i]=kz[k] THEN
BEGIN
  sa[j]:=sa[j]+sa[k];
  sb[j]:=sb[j]+sb[k]
END;
END;
END;
END;  (kzweig)

PROCEDURE festlaus;{Определение выходн. величин }
BEGIN
FOR i:=1 TO iz DO
BEGIN
  mi[i]:=i;
  mu[i]:=i;
END;
writeln;
write('Какие параметры ветви должны быть ');
writeln;  ('выведены? ');
write('Количество напряжений ветвей');
REPEAT read(ma) UNTIL ma<=iz;
IF (ma>0) AND (ma<iz) THEN
BEGIN
  writeln;
  write('Номера ветвей' )
  FOR i:=1 TO ma DO readln(mu[i]);
END;
writeln;
write('Количество токов ветвей' )
REPEAT read(mb) UNTIL mb<=iz;
IF (mb>0) AND (mb<iz) THEN
BEGIN
  writeln;
  write('Номера ветвей' )
  FOR i:=1 TO mb DO readln(mi[i]);
END;
writeln
END;  (festlaus)

PROCEDURE ausfest;{ Вывод параметров ветвей }
BEGIN
FOR i:=1 TO ma DO
  IF mu[i]<=iz THEN writeln('U',mu[i],', '= ', ->
    ->ua[mu[i]],',V');
FOR i:=1 TO mb DO
  IF mi[i]<=iz THEN writeln('I',mi[i],', '= ', ->
    ->sa[mi[i]],',A');
END;  (ausfest)

```

```

PROCEDURE kausfest; {Вывод компл. парам. ветвей }
BEGIN
FOR i:=1 TO ma DO
BEGIN
j:=mu[i];
IF j<=iz THEN writeln('U',j,'= ',
betrag(ua[j],ub[j]),' V ',
wingrad(ua[j],ub[j]):3:2,' Grad')
END;
FOR i:=1 TO mb DO
BEGIN
j:=mi[i];
IF j<=iz THEN writeln('I',j,'= ',
betrag(sa[j],sb[j]),' A ',
wingrad(sa[j],sb[j]):3:2,' Grad')
END;
END; {kausfest}

PROCEDURE tridiag; { Решение трехдиагональной }
LABEL ende; { системы уравнений }
VAR i1,i2: INTEGER; r1: REAL;
BEGIN
IF nt<2 THEN
BEGIN
writeln('nt =',nt,': Решение невозможно');
GOTO ende
END;
FOR i2:=2 TO nt DO
BEGIN
i1:=i2-1;
r1:=dn[i1]/dh[i1];
dh[i2]:=dh[i2]-r1*dn[i1];
rs[i2]:=rs[i2]-r1*rs[i1]
END;
IF randbed[i]='B' THEN
BEGIN
dh[nt]:=dh[nt]-r1*dn[nt-1];
rs[nt]:=rs[nt]-r1*rs[nt-1]
END;
rs[nt]:=rs[nt]/dh[nt];
FOR i1:=nt-1 DOWNTO 1 DO rs[i1]:=(rs[i1]-dn[i1]*
->rs[i1+1]/dh[i1]);
ende: END; { решение трехдиаг. системы ур-ний }
PROCEDURE verteber; { Выбор начальных значений для }
BEGIN { интерполяции внутри области знач. }
xk:=xs[ns[i],i];
IF sv[i]<xk THEN
BEGIN IF sv[i]<xs[1,i] THEN xk:=xs[1,i] ELSE xk:=
END; {verteber} >= sv[i] END

```

```

PROCEDURE interpol; { Интерполяция }
LABEL ende;
VAR a1,x1: REAL;
BEGIN
FOR iv:=1 TO ns[i]-1 DO
  IF xk<=xs[iv+1,i] THEN
    BEGIN
      x1:=xk-xs[iv,i];
      a1:=as[iv,i]*x1;
      yk:=(a1+bs[iv,i])*x1+cs[iv,i]*x1+ys[iv,i];
      sk:=(3*a1+2*bs[iv,i])*x1+cs[iv,i];
      GOTO ende
    END;
  ende: END; {interpol}

PROCEDURE splinea; { Расчет сплайн-коэффициентов }
VAR j1: INTEGER; { для граничных условий а }
BEGIN
  nt:=ni-1;
  FOR j:=1 TO ni DO
    BEGIN
      j1:=j+1;
      sh[j]:=xs[j1,i]-xs[j,i];
      se[j]:=(ys[j1,i]-ys[j,i])/sh[j]
    END;
  FOR j:=1 TO nt DO
    BEGIN
      j1:=j+1;
      dh[j]:=2*(sh[j]+sh[j1]);
      dn[j]:=sh[j1];
      rs[j]:=3*(se[j1]-se[j])
    END;
  tridiag;
  bs[1,i]:=0;
  rs[ni]:=0;
  FOR j:=1 TO ni DO
    BEGIN
      bs[j+1,i]:=rs[j];
      as[j,i]:=(rs[j]-bs[j,i])/(3*sh[j]);
      cs[j,i]:=se[j]-sh[j]*(rs[j]+2*bs[j,i])/3
    END;
  END; {splinea}

PROCEDURE splineb; { Расчет сплайн-коэффициентов }
VAR j1,nv,n1: INTEGER; { для граничных условий б }
BEGIN
  nv:=ni-1;
  nt:=ni-2;
  n1:=nt-1;

```

```

FOR j:=1 TO ni DO
BEGIN
  j1:=j+1;
  sh[j]:=xs[j1,i]-xs[j,i];
  se[j]:=(ys[j1,i]-ys[j,i])/sh[j]
END;
as[1,i]:=0;
bs[1,i]:=0;
cs[1,i]:=se[1];
as[ni,i]:=0;
bs[ni,i]:=0;
cs[ni,i]:=se[ni];
FOR j:=1 TO ni DO
BEGIN
  j1:=j+1;
  dh[j]:=2*(sh[j]+sh[j1]);
  dn[j]:=sh[j1];
  rs[j]:=3*(se[j1]-se[j])
END;
dh[1]:=2*sh[2];
dh[nt]:=4*sh[nt]+3*sh[nv];
rs[nt]:=9*se[nv]-6*se[nt]-3*se[ni];
tridiag;
cs[2,i]:=se[1];
bs[2,i]:=rs[1];
FOR j:=2 TO nt DO
BEGIN
  j1:=j+1;
  bs[j1,i]:=rs[j];
  as[j,i]:=(bs[j1,i]-bs[j,i])/(3*sh[j]);
  cs[j1,i]:=cs[j,i]+sh[j]*(2*bs[j,i]+3*as[j,i]*sh[j])
END;
as[nv,i]:=((se[nv]-cs[nv,i])/sh[nv]-bs[nv,i])/sh[nv]
END; (splineb)

PROCEDURE dis_nlin; { Определение дискретной }
BEGIN { токовой модели для каждого }
FOR i:=1 TO nz DO { нелинейного двухполюсника }
BEGIN
  verteber;
  interpol;
  CASE zweigart[i] OF
    'G': BEGIN gd[i]:=sk; sd[i]:=yk-sk*xk END;
    'R': BEGIN gd[i]:=1/sk; sd[i]:=xk-yk/sk END;
    'C': BEGIN
      gd[i]:=-al[0]*yk/h;
      sd[i]:=-sr[i]/h;
      C[i]:=yk
    END;
  END;

```

```

'L': BEGIN
    gd(i):=-h/(al(0)*yk);
    sd(i):=-sr(i)/(al(0)*yk)
    L(i):=yk
END;
END; {of case}
END;
END; {dis_nlin}

PROCEDURE zweig_nlin;{ Расчет параметров ветви }
BEGIN { нелинейных 2-полюсников }
FOR i:=1 TO nz DO
BEGIN
    ua(i):=un(kv(i))-un(kz(i));
    sa(i):=ua(i)*gd(i)+sd(i);
    CASE zweigart(i) OF
    'G', 'C': zs(i):=ua(i);
    'R', 'L': zs(i):=sa(i)
    END; {of case}
END;
END; {zweig_nlin}

PROCEDURE start_nlin;{ Ввод начальных значений для }
BEGIN { нелинейных 2-полюсников }
writeln(' Выбрать новые начальные значения')
FOR i:=1 TO nz DO
BEGIN
    write('Zweig ',i,' : ');
    read(sv(i));
    writeln;
    IF progart='D' THEN zg(i,1):=sv(i);
END;
END; {start_nlin}

PROCEDURE pruef_nlin;{Проверка нелин.2-полюсников }
BEGIN
pruefbcd:=true;
FOR i:=1 TO nz DO
BEGIN
    CASE zweigart(i) OF
    'G', 'C': xk:=ua(i);
    'R', 'L': xk:=sa(i);
    END; {of case}
    IF xk<xs(1,i) THEN pruefbcd:=false;
    IF xk>xs(ns(i),i) THEN pruefbcd:=false
END;
END; {pruef_nlin}

```

```

PROCEDURE koeff(kb: INTEGER); { Расчет коэфф.альфа }
VAR i1,i2: INTEGER;          { и гамма }
BEGIN
  al[0]:=-kb;
  al[1]:=kb;
  ga[1]:=kb;
  FOR i2:=2 to kb DO
  BEGIN
    i1:=i2-1;
    ga[i2]:=ga[i1]*(i1-kb)/i2;
    al[i2]:=ga[i2]/i2;
    al[0]:=al[0]-al[i2]
  END;
END;    (koeff)

```

```

PROCEDURE augvert(i: INTEGER; ein: REAL; VAR aus: REAL);
VAR s1: REAL; { Расчет мгновенных значений }
BEGIN          { параметров источников }
  IF ein=0 THEN aus:=0 ELSE
  CASE zeitabh(i) OF
    'K': aus:=ein;
    'L': BEGIN
      s1:=zf[i]*t+vf[i];
      IF abs(s1)>abs(ein) THEN aus:=ein ELSE aus:=s1;
    END;
    'S': aus:=ein*cos(zf[i]*t+vf[i]);
  END;    (of case)
END;    (augvert)

```

```

PROCEDURE dismod; { Расчет дискретной токовой }
VAR s1: REAL;     { модели для каждого }
BEGIN            { линейного 2-полосника }
  FOR i1:=1 TO is DO augvert(i1+ir,qr[i1],qt[i1]);
  vorver;
  FOR i:=nz+1 TO ir DO
  BEGIN
    gd[i]:=0; sd[i]:=0;
    CASE zweigart[i] OF
      'R','G': gd[i]:=g[i];
      'I': augvert(i, sr[i], sd[i]);
      'T': gd[i]:=sf[i]*(sd[mz[i]]+gd[mz[i]]*sv[mz[i]]);
      'C': BEGIN
        gd[i]:=-al[0]*C[i]/h;
        s1:=0;
        FOR j:=1 TO kb DO s1:=s1+al[j]*zg[i,j];
        sd[i]:=-C[i]*s1/h;
      END;
    'L': BEGIN
      gd[i]:=-h/(al[0]*L[i]);
    END;
  END;
END;

```

```

        s1:=0;
        FOR j:=1 TO kb DO s1:=s1+a1[j]*zg[i,j];
        sd[i]:=-s1/a1[0];
    END;
END; {of case}
END;
END; {dismod}

PROCEDURE umspaus;{Перезапись параметров }
BEGIN
    { состояния и вывод }
    FOR i:=1 TO ir DO { параметров ветви }
    BEGIN
        FOR j:=11 DOWNT0 2 DO zg[i,j]:=zg[i,j-1];
        CASE zweigart[i] OF
            'G', 'C': zg[i,1]:=ua[i];
            'R', 'L': zg[i,1]:=sa[i];
        END; {of case}
    END;
    FOR i:=1 TO nz DO
    BEGIN
        FOR j:=11 DOWNT0 2 DO ze[i,j]:=ze[i,j-1];
        CASE zweigart[i] OF
            'C': ze[i,1]:=C[i];
            'L': ze[i,1]:=L[i];
        END; {of case}
    END;
    write('t = ',t,' s ; ');
    ausfest
END; {umspaus}

PROCEDURE anazeit; { Анализ динамических цепей в }
LABEL finale,iter,start,weiter;{ данный момент }
BEGIN
    { времени }
    FOR i:=1 TO nz DO
    BEGIN
        sr[i]:=0;
        sv[i]:=0;
        FOR j:=1 TO kb DO
        BEGIN
            sr[i]:=sr[i]+a1[j]*zg[i,j]*ze[i,j];
            sv[i]:=sv[i]+ga[j]*zg[i,j]
        END;
    END;
    uv:=un;
    iter: FOR k1:=1 TO 50 DO { Начало итерации метода }
    BEGIN
        { Ньютона }
        dis_nlin;
        dismod;
        aufloes;
    END;
    END;

```



```

IF singul THEN knoten ELSE
  BEGIN writeln('Решение невозможно '); GOTO → start END;
zweig_nlin;
FOR i:=1 TO nz DO
  BEGIN
    IF abs(sv[i])<1.e-30
    THEN BEGIN IF abs(zs[i])>en THEN GOTO weiter END
    ELSE BEGIN IF abs((zs[i]-sv[i])/sv[i])>en THEN → GOTO weiter END
  END;
GOTO finale;
weiter: FOR i:=1 TO nz DO
  BEGIN IF k1>4 THEN sv[i]:=0.5*(zs[i]+sv[i])
        ELSE sv[i]:=zs[i] END;
END; { Конец итерации метода Ньютона }
writeln('Прерывание итерации:');
writeln('Метод Ньютона не сходится');
start: start_nlin; GOTO iter;
finale: zweig; pruef_nlin;
IF pruefbed THEN umspaus ELSE GOTO start;
IF jf THEN FOR i:=1 TO kn DO
  BEGIN
    dx:=dx+abs(un[i]-uv[i]);
    sx:=sx+abs(un[i]);
  END;
END; { anazeit }

PROCEDURE einglob; { Ввод глобальных параметров }
LABEL start;
BEGIN
  writeln('Ввод глобальных параметров');
  start: writeln;
  write('Количество ветвей');
  REPEAT read(iz);
    IF iz<2 THEN writeln('Ввод не имеет смысла');
  UNTIL iz>1;
  writeln;
  write('Количество узлов ( без базового )');
  REPEAT read(kn);
    IF kn<1 THEN writeln('ввод не имеет смысла');
  UNTIL kn>0;
  writeln;
  write('Количество идеальных источников напряжения');
  REPEAT read(is);
    IF is>iz-1 THEN writeln('Ввод не имеет (
    ( смысла)');
  UNTIL is<iz;
  n:=kn-is; ir:=iz-is;
  IF (n<0) OR (ir<1) THEN GOTO start;
  agq:=false;
END; { einglob }

```

```

PROCEDURE ein_nlin;{ Ввод нелинейных 2-полюсников}
BEGIN
writeln;
write(' Количество нелинейных двухполюсников');
read(nz); writeln;
IF nz>0 THEN writeln('Количество опорных точек');
FOR i:=1 TO nz DO
REPEAT
  write('Двухполюсник', i, ':');
  read(ns[i]); writeln;
UNTIL ns[i]>2;
FOR i:=1 TO nz DO
BEGIN
  writeln;
  write('Ветвь ', i, ' из узла '); read(kv[i]);
  write(' к узлу '); read(kz[i]); writeln;
  writeln('Двухполюсник', i, ':ввод опорных точек');
  FOR j:=1 TO ns[i] DO
  BEGIN
    write('x', j, ': ');
    read(xs[j, i]); writeln;
    write('y', j, ': ');
    read(ys[j, i]); writeln;
  END;
  ni:=ns[i]-1;
  REPEAT
    write(' Граничные условия ( а/б)');
    read(Kbd, randbed[i]); write(randbed[i]);
  UNTIL randbed[i] IN ['A', 'B']; writeln;
  IF randbed[i]='A' THEN splinea ELSE splineb,
  IF progart='G' THEN
  REPEAT
    write('Тип ветви (R / G) ');
    read(Kbd, zweigart[i]); write(zweigart[i]);
  UNTIL zweigart[i] IN ['R', 'G'];
  IF progart='D' THEN
  REPEAT
    write('Тип ветви (R / G / L / C) ');
    read(Kbd, zweigart[i]); write(zweigart[i]);
  UNTIL zweigart[i] IN ['R', 'G', 'L', 'C'];
  writeln; write(' Начальное значение ');
  CASE zweigart[i] OF
    'G', 'C': write('напряжения' );
    'R', 'L': write(' тока' );
  END; {of case}
  read(sv[i]); writeln;
  IF progart='D' THEN zg[i, 1]:=sv[i]
END;
END; {ein_nlin}

```

```

PROCEDURE einzeit(i: INTEGER); {Ввод временной зав.}
BEGIN { идеального источника }
  writeln;
  write('Временная зависимость' ( K/L/S) );
  REPEAT read(kbd, zeitabh[i]) UNTIL zeitabh[i] in 'K', 'L', 'S';
  write(zeitabh[i]); writeln;
  CASE zeitabh[i] OF
    'L': BEGIN
      write('временной коэффициент' );
      read(zf[i]);
      writeln;
      write('отрезок оси ');
      read(vf[i]);
    END;
    'S': BEGIN
      write('Круговая частота' );
      read(zf[i]);
      writeln;
      write('начальный фазовый угол');
      read(vf[i]);
    END;
    'K': END; {of case}
  END; {einzeit}

```

```

PROCEDURE einlin(i: INTEGER); { Ввод элементов }
LABEL schleifel; { линейной ветви }
VAR js: INTEGER; rl: REAL;
BEGIN
  writeln; write(' ветвь ', i);
  write(' из узла '); read(j);
  write(' к узлу '); read(k);
  writeln;
  IF i <= ir THEN
    BEGIN
      schleifel: kv[i] := j; kz[i] := k;
      G[i] := 0; L[i] := 0; C[i] := 0;
      sr[i] := 0; si[i] := 0; mz[i] := 0;
      write(' элемент ');
      REPEAT read(kbd, zweigart[i]) UNTIL
        zweigart[i] in ('R', 'G', 'L', 'C', 'I', 'S', 'T');
      write(' ');
      CASE zweigart[i] OF
        'R': BEGIN write('сопротивление : ');
          REPEAT read(rl) UNTIL rl > 0; G[i] := 1/rl;
        'G': BEGIN write('проводимость: ');
          REPEAT read(G[i]) UNTIL G[i] > 0; END;
        'T': BEGIN IF progart = 'S' THEN GOTO schleifel;
          writeln('проводим. управляемая током');
          write('Коеф. управл. '); read(sf[i]);

```

```

        REPEAT
            writeln; write('Управл.ветвь '); read(mzfil);
        UNTIL (mzfil<=nz) AND (zweigart[mzfil]='G'); END;
    'L': BEGIN IF progart='G' THEN GOTO schleifel;
            write('ИНДУКТИВНОСТЬ');
            REPEAT read(Lfil) UNTIL Lfil>0;
        IF progart='D' THEN BEGIN write('; Нач. знач. : ');
            read(zgfil); END; END;
    'C': BEGIN IF progart='G' THEN GOTO schleifel;
            write('ЕМКОСТЬ: ');
            REPEAT read(Cfil) UNTIL Cfil>0;
        IF progart='D' THEN BEGIN write('; Нач. знач. : ');
            read(zgfil); END; END;
    'S': BEGIN agq:=true;
            REPEAT
                writeln('Управляемый источник тока');
                write('Параметр управления (I/U)');
                read(kbd,ch); write(ch);
            UNTIL ch in ['I','U'];
            IF ch='I' THEN write('Усил.по току B= ');
                ELSE write('Коеф.управл. S = ');
            read(sfil); writeln;
            write('Управляющая ветвь ');
            REPEAT read(js) UNTIL js>0;
            IF ch='I' THEN mzfil:=-js ELSE mzfil:=js
        END;
    'I': BEGIN write('Идеал.ист.тока:');read(srfil);
            CASE progart OF 'S': BEGIN writeln;
                write('МНИМ.ч. ');
                read(sifil) END;
            'D': einzeit(i);
            'G': zeitabhfil:='K';
        END; {of case}
    END;
END; {of case}
END ELSE
BEGIN
    il:=i-ir;
    kafil:=j; kufil:=k;
    write('Напр.источника '); read(qrfil);
    CASE progart OF 'S': BEGIN writeln;
                write('МНИМАЯ ЧАСТЬ:');
                read(qifil) END;
            'D': einzeit(i);
            'G': zeitabhfil:='K';
        END; {of case}
    END;
    writeln;
END; {einlin}

```

```

PROCEDURE aenderlin; {Изменение элементов лин.ветви}
BEGIN
writeln; writeln(' В скольких линейных ветвях' );
write(' должны быть изменены элементы ?') ;
read(me); writeln;
FOR m1:=1 TO me DO
BEGIN
write(' Ветвь' ) ;
REPEAT read(i) UNTIL (i>nz) AND (i<=iz);
writeln; einlin(i)
END;
END; {aenderlin}

PROCEDURE gleich; { Цель постоянного тока }
LABEL ende, finale, iter, start, weiter;
BEGIN
ClrScr;
writeln(' Анализ цепи постоянного тока ');
writeln; einglob;
writeln; ein_nlin;
FOR i:=nz+1 TO iz DO einlin(i);
festlaus; writeln;
iter: knotfeld; { Начало итерации метода Ньютона }
FOR k1:=1 TO 20 DO
BEGIN
dis_nlin;
dismod;
aufloes;
IF singul THEN knoten
ELSE BEGIN
writeln(' Решение невозможно' );
IF nz>0 THEN GOTO start ELSE GOTO ende
END;
zweig_nlin;
FOR i:=1 TO nz DO {Запрос,пригодно ли приближ. }
BEGIN
IF abs(sv[i])<1.e-30
THEN BEGIN IF abs(zs[i])>en THEN GOTO weiter END
ELSE BEGIN IF abs((zs[i]-sv[i])/sv[i])>en THEN
→GOTO weiter END
END;
GOTO finale;
weiter: FOR i:=1 TO nz DO
BEGIN IF k1>4 THEN sv[i]:=0.5*(zs[i]+sv[i])
ELSE sv[i]:=zs[i] END;
END; { 'Конец итерации метода Ньютона }
writeln('Прерывание итерации:');
writeln('Метод Ньютона не сходится');
start: start_nlin;
GOTO iter;

```

```

finale: zweig; pruef_nlin; {Выв.и выв.парам.ветвей}
IF pruefbed THEN ausfest ELSE GOTO start;
IF nz>0 THEN
BEGIN
  writeln;
  REPEAT
    write('Следующее начальное значение (J/N)?');
    read(Kbd,ch);
  UNTIL ch IN ['J','N'];
  writeln;
  IF ch='J' THEN GOTO start
END;
ende: aenderlin;
IF me>0 THEN GOTO iter
END;  (gleich)

```

```

PROCEDURE sinus; { Анализ лин.цепи синус.напряж. }
LABEL frequenz;
VAR f,fa,fe,fs: REAL;
BEGIN
  ClrScr;
  writeln('Анализ лин.цепи синусоид.напряжения' );
  writeln;
  einglob; writeln;
  FOR i:=1 TO iz DO einlin(i);
  knotfeld;
  kvorver;
  festlaus;
  frequenz;
  writeln;
  writeln('Частота: ');
  write('Нач.знач.   : '); read(fa); writeln;
  write('Знач.шага   : '); read(fs); writeln;
  write('Конеч.знач.  : '); read(fe); writeln;
  f:=fa-fs;
  REPEAT
    f:=f+fs;
    w:=2*pi*f;
    writeln;
    writeln('f =',f:7:3, ' Hz');
    kaufloes;
    IF singul THEN BEGIN kknoten; kzweig; kausfest;END
    ELSE writeln('Решение невозможно');
  UNTIL (f+fs)>fe;
  writeln; write('След. значение частоты (J/N)?');
  REPEAT read(kbd,ch) UNTIL ch IN ['J','N'];
  IF ch='J' THEN GOTO frequenz
    ELSE BEGIN aenderlin; IF me>0 THEN GOTO →
END;  (sinus)                → frequenz END

```

```

PROCEDURE dyna;
LABEL start, ziel;
BEGIN
  ClrScr;
  writeln('Анализ динамической цепи') ;
  writeln;
  einglob;
  writeln;
  ein_nlin;
  FOR i:=nz+1 TO iz DO einlin(i);
  start: knotfeld;
  festlaus;
  writeln;          ( Ввод управляющих параметров)
  write('Нач. значение шага      : ');
  read(ha);
  writeln;
  write(' Макс.значение шага    : ');
  read(he);
  writeln;
  REPEAT
    write('Максимальное значение порядка :') ;
    read(ke);
    writeln;
  UNTIL (ke>1) AND (ke<7);
  write(' Конечное значение времени te:');
  read(te);
  writeln;
  write(' Значение предела') ;
  read(eb);
  writeln;
  mt:=10; { При значении шага h =he знач.порядка}
  un(0):=0; { kb повышается от 1 до ke ;при }
  t:=0;    { каждом kb значении цепь анализируется}
  h:=ha;   { для mt моментов }
  jf:=false;
  FOR kb:=1 TO ke DO
    BEGIN
      koeff(kb);
      FOR m1:=1 TO mt DO
        BEGIN
          t:=t+h;
          anazeit
        END;
      END;
    jf:=true; { Для каждого значения вычисляется и}
    REPEAT { заданным образом удваивается}
      REPEAT { значение шага}
        dx:=0; vx:=0;
        FOR m1:=1 TO mt DO

```

```

BEGIN
  t:=t+h;
  IF t>te+he THEN GOTO ziel;
  snazeit;
  IF sx>epsilon THEN dx:=dx/sx;
  END;
  UNTIL dx<eb;
  h:=2*h;
  FOR j:=1 TO 6 DO
    FOR i:=1 TO ir DO zg[i,j]:=zg[i,2*j-1];
  UNTIL h>=he;
  jf:=false; { Дальнейшие вычисления выполняются }
  REPEAT { с максимальным значением шага до }
    t:=t+h; { конечного значения времени }
    anazeit
  UNTIL t>=te;
  ziel: writeln; aenderlin;
  IF me>0 THEN GOTO start
  END; {dyna}

```

{ Основная программа }

```

BEGIN
  Start: ClrScr;
  writeln('N W A');
  writeln(' Анализ цепей )
  gotoxy(33,1);
  writeln('Расчет электрических схем' );
  gotoxy(33,2);
  writeln('Проф. доктор техн. наук В.Нерретер');
  writeln;
  writeln('G цепь постоянного тока' );
  writeln('S линейная цепь синусоидального напряж. ');
  writeln('D динамическая цепь переходный процесс');
  REPEAT
    writeln;
    write('Пожалуйста, выберите ... ');
    read(Kbd,progart);
  UNTIL progart IN ['G','S','D'];
  CASE progart OF
    'G': gleich;
    'S': sinus;
    'D': dyna;
  END; {of case}
  writeln;
  write(' Дальнейшие вычисления (J/N)?');
  REPEAT read(Kbd, ch) UNTIL ch IN ['J','N'];
  IF ch='J' THEN GOTO Start
  END.

```


ПРИЛОЖЕНИЯ

П1. Указатель программ на языке БЕЙСИК

- DIFF-KAP — расчет дифференциальной емкости
DIFF-IND — расчет дифференциальной индуктивности магнитного контура
DYNA-LIN — анализ переходного процесса линейной динамической двухполюсной цепи
DYNA-NL — анализ переходного процесса динамической цепи с нелинейными двухполюсниками
GLEICH — узловой анализ цепи постоянного тока с управляемыми источниками тока
NEWTON — узловой анализ цепи постоянного тока с нелинейными двухполюсниками
SINUS — узловой анализ линейной цепи переменного тока с управляемыми источниками тока
SPLINE — интерполяция с помощью сплайн-функций

П2. Указатель подпрограмм на языке БЕЙСИК

- 2000 — решение системы линейных уравнений
2100 — запоминание массива узлов
2200 — решение системы линейных уравнений с симметричной матрицей коэффициентов
2300 — упорядочение системы уравнений
2400 — ввод управляемых источников тока
2500 — ввод элементов цепи для ветви I
2600 — подготовка перемещения идеальных источников напряжения
2700 — составление и решение системы линейных уравнений
2800 — расчет узловых напряжений
2900 — расчет параметров ветвей
3000 — решение системы линейных уравнений
3100 — преобразование $R \rightarrow P$ для комплексных чисел
3200 — решение системы линейных уравнений с симметричной матрицей коэффициентов
3300 — упорядочение системы уравнений
3400 — определение выводимых величин
3500 — ввод элементов цепи для ветви I
3600 — подготовка перемещения идеальных источников напряжения
3800 — расчет узловых потенциалов
3900 — расчет параметров ветвей
4000 — расчет дискретной токовой модели
4100 — расчет параметров ветвей
4300 — запоминание параметров состояния
4400 — расчет мгновенных значений параметров источников
4500 — ввод элементов цепи для ветви I
4800 — расчет коэффициентов
5000 — решение симметричной трехдиагональной системы уравнений
5100 — расчет сплайн-коэффициентов согласно граничному условию а)
5200 — расчет сплайн-коэффициентов согласно граничному условию б)
5300 — интерполяция
5400 — подготовка интерполяции
6000 — анализ в момент времени T

ПЗ. Переменные, используемые в программах

A(N, N)	— матрица узловых проводимостей, матрица коэффициентов
AL(6)	— коэффициенты α
AR	— реальная часть комплексной переменной
AS(NI, NZ)	— сплайн-коэффициенты a_j
B(N, N)	— мнимая часть матрицы узловых проводимостей
BI	— мнимая часть комплексной переменной
BS(NM, NZ)	— сплайн-коэффициенты b_j
C(IR)	— емкости
CS(NI, NZ)	— сплайн-коэффициенты c_j
DH(NT)	— элементы главной диагонали
DN(NT)	— элементы соседних диагоналей
DX	— D_{j+1} [см. уравнение (4.4-24)]
EB	— значение предела ϵ для метода обратного дифференцирования
EI; EU	— выходные величины, обычная точность
EK	— выходная величина, обычная точность
EN	— значение предела ϵ для метода Ньютона
EP	— значение предела при решении системы линейных уравнений
ER; EW	— выходные величины, обычная точность
F	— частота
FA	— начальное значение частоты
FE	— конечное значение частоты
FR\$	— алфавитно-цифровая запрашиваемая переменная
FS	— значение шага для частоты
G(RZ)	— проводимость
GA(6)	— коэффициенты γ
GD(IR)	— проводимость дискретной токовой модели
GI(IR)	— мнимая часть проводимости
H	— значение шага
HA	— начальное значение шага
HE	— конечное значение шага
I	— переменная цикла; номер ветви
IA(I)	— тип ветви
IG	— наличие управляемых источников
IL	— переменная цикла; номер идеального источника напряжения
IR	— количество нагрузочных ветвей
IS	— количество идеальных источников напряжения
IT(IZ)	— тип временной зависимости (см. п. 4.5.2)
IV	— номер интервала
IZ	— количество ветвей
J	— переменная цикла; номер узла
JD	— вспомогательное значение
JF	— коэффициент управления для расчета ошибки
JR	— номер строки, соответствующий узлу J
JS	— номер столбца (начальный узел управляющей ветви)
K	— переменная цикла; номер узла
KA(IS)	— зависимые узлы
KB	— значение порядка k для метода BDF
KE	— наивысшее значение порядка для метода BDF

KF(KN)	— массив узлов
KL	— переменная цикла для метода Ньютона
KN	— количество узлов
KR	— номер строки, соответствующий узлу K
KS	— номер столбца (конечный узел управляющей ветви)
KU(IS)	— независимые узлы
KV(IR)	— начальный узел нагрузочной ветви
KZ(IR)	— конечный узел нагрузочной ветви
L(IR)	— индуктивности
MA	— количество выдаваемых напряжений ветвей
MB	— количество выдаваемых токов ветвей
ME	— количество ветвей, в которых изменяются элементы
MG(NZ)	— управляющие ветви проводимостей, управляемых током
MI(IR)	— номера ветвей, выдаваемых токов ветвей
ML	— переменная цикла
MS	— номер управляющей ветви управляемого источника тока
MT	— конечное значение цикла
MU	— номера ветвей выдаваемых напряжений ветвей
MZ(IR)	— управляющие ветви управляемых источников тока
N	— количество уравнений системы линейных уравнений
NI	— количество интервалов
NM	— максимальное число опорных точек
NS(NS)	— число опорных точек нелинейных двухполюсников
NT	— число уравнений трехдиагональной системы уравнений
NZ	— количество нелинейных двухполюсников
PH	— угол комплексной переменной
PI	— число π
QI(IS)	— мнимая часть напряжений источников
QR(IS)	— напряжения идеальных источников
QT(IS)	— напряжения источников в момент t
RP	— значение комплексной переменной
RS(NT)	— правая часть трехдиагональной системы уравнений
SA(IR)	— токи в резистивных ветвях
SB(IR)	— мнимые части токов в ветвях
SD(IR)	— токи источников дискретной токовой модели
SE(NI)	— коэффициенты e_i (см п. 5.1.4)
SF(IR)	— коэффициенты управления S или усиления по току B
SH(NI)	— коэффициенты h_i (см. п. 5.1.4)
SI(IR)	— мнимые части токов источников
SK	— наклон m_k интерполирующего сплайна для x_k
SR(IR)	— токи источников нагрузочных ветвей
ST	— параметр передачи (см. п. 4.5.2)
SX	— S_{j+1} [см. уравнение (4.4-23)]
SW(NZ)	— стартовое значение
T	— момент времени t

TE	— конечное значение времени
UA(IR)	— напряжения нагрузочных ветвей
UB(IR)	— мнимые части напряжений ветвей
UI(KN)	— мнимые части узловых потенциалов
UN(KN)	— узловые потенциалы
UV(KN)	— узловые потенциалы для предыдущего момента времени
VF(IZ)	— отрезки оси или начальные фазовые углы
VI(IR)	— мнимые части соответствующих напряжений источников
VR(IR)	— напряжения источников, соответствующие нагрузочным ветвям
W	— круговая частота
X(N)	— правая часть; вектор решения
XK	— координата x при интерполяции
XS(NM, NZ)	— опорные точки
Y(N)	— мнимые составляющие правой части или вектора решения
YK	— координата y при интерполяции
YS(NM, NZ)	— значение функции в опорной точке
ZE(IR, 11)	— значения параметров двухполюсного накопителя энергии
ZF(IZ)	— наклон или круговая частота (см. п. 4.5.2)
ZG(IR, 11)	— параметры состояния
ZS(NZ)	— параметры ветвей (см. § 5.4)

П4. Указатель функций и процедур программы NWA (версия 1.0)

aenderlin,	knoten,
anazeit,	knotfeld,
aufloes,	koeff(kb),
augwert (i, ein, aus),	kvorver,
ausfest,	kzweig,
betrage (a, b),	pruef_nlin,
dismod,	sgn(a),
dis_nlin,	sinus,
dyna,	splinea,
einglob,	splineb,
einlin(i),	start_nlin
einzeit(i),	tridiag,
ein_nlin	umspaus,
festlaus,	vorver,
gleich,	werteber,
interpol,	wingrad(a, b),
kaufloes,	zweig,
kausfast,	zweig_nlin,
kknoten,	

П5. Константы и переменные программы NWA (версия 1.0)

Все переменные, обозначаемые одной или двумя буквами, приведены в приложении ПЗ.

agq — наличие управляемых источников

ch	— алфавитно-цифровой символ
epsilon	— значение предела при решении системы линейных уравнений
ismax	— максимальное количество управляемых источников
izmax	— максимальное количество ветвей
knmax	— максимальное количество узлов
nsmax	— максимальное количество опорных точек
nzmax	— максимальное количество нелинейных двухполюсников
progart	— тип программы
pruefbed	— условие проверки (параметры ветвей нелинейных двухполюсников)
randbed [i]	— граничные условия <i>a</i> или <i>b</i>
singul	— решаемость системы линейных уравнений
zeitabh [i]	— временная зависимость идеальных источников
zweigart [i]	— тип ветви

П6. Версия 2.0 программы NWA на языке ПАСКАЛЬ

Персональная ЭВМ с операционной системой MS-DOS (PC DOS) имеет в своем распоряжении версию программы анализа цепей NWA 2.0, записанную на дискете. Дискета содержит исходный текст программы NWA.PAS на ТУРБО-ПАСКАЛЕ и оттранслированные программы NWA.COM и NWA-87.COM, а также несколько наборов данных для примеров

Запуск программы. Для эксплуатации программы NWA требуется лишь минимальная конфигурация ПЭВМ с объемом рабочей памяти 256 Кбайт. Перед считыванием программы дискета устанавливается в дисководе. После ввода команды NWA программа считывается и запускается.

Если в ПЭВМ встроен арифметический сопроцессор 8087, то скомпилированную для нее программу анализа цепей можно вызвать и запустить командой NWA=87.

После запуска программы необходимо ввести имя файла данных.

Заголовок файла данных. Файл данных может быть записан с помощью любого программногo редактора (например, WORDSTAR) и размещен на дискете. Каждый файл должен начинаться с четырех символов NWA, а в качестве пятого символа должна быть буква, означающая тип анализа. С помощью программы NWA можно анализировать три типа цепей: постоянного тока (G), синусоидального напряжения (S) и динамические цепи (D). Цепи типа G и S могут содержать нелинейные двухполюсники.

Первая строка файла, в которой находятся данные цепи, теперь записывается в виде NWA-G, NWA-S или NWA-D. В противном случае выдается сообщение об ошибке «файл не существует».

Идентификаторы. После строки заголовка следует до 150 строк описания двухполюсников. Поскольку каждая ветвь цепи может содержать один-единственный двухполюсник, число строк равно количеству двухполюсников. Ветви цепи нумеровать нет необходимости. Каждый двухполюсник многократно описывается последовательностью символов, которые в последующем интерпретируются как идентификаторы.

Первая буква идентификатора должна быть заглавной (прописной). Она определяет тип двухполюсника. Допускаются следующие

типы двухполюсников: C — емкость; G — проводимость; I — идеальный источник тока; L — индуктивность; R — сопротивление; S — управляемый источник тока; T — управляемая проводимость; U — идеальный источник напряжения.

Если проводимость G или сопротивление R , индуктивность L или емкость C нелинейны, то вторым символом идентификатора должна быть прописная буква N , любой другой символ, в том числе и строчная буква n , означает линейный двухполюсник.

Пример:

$Rnpgm$ — линейное (активное) сопротивление;

$Unenn$ — идеальный источник напряжения;

$RNa13$ — нелинейное сопротивление.

Идентификатор может состоять из любого количества символов, однако ЭВМ воспринимает только первые пять из них.

Номера узлов. В строке двухполюсника после идентификатора должен следовать по меньшей мере один пробел. После этого задается номер узла, в котором начинается данная ветвь, а затем номер узла, в котором она заканчивается. Эти параметры также отделяются друг от друга одним или несколькими пробелами. Базовому узлу цепи присваивается номер 0. В цепи может быть до 60 узлов. Если количество узлов превышает 60 или последовательность номеров «идет с разрывами», то выдается сообщение об ошибке.

Значение параметра ветви. После задания номера узла для каждого линейного двухполюсника указывается численное значение параметра ветви. В отличие от номеров узлов, которые должны быть целыми числами, значение параметра ветви может задаваться в виде десятичной дроби без пробелов, причем вместо десятичной запятой используется точка. Десятичная степень определяется буквой E .

Управляемые двухполюсники. Численное значение параметра ветви управляемого двухполюсника определяет его коэффициент управления. Для проводимости схемы замещения транзистора (см. рис. 5.13) коэффициент этот является константой Эрли. Затем задаются разделенные пробелом (пробелами) буквы SZ и идентификатор управляющей ветви. Для управляемых источников тока кроме букв SG необходимо задать тип управляющего параметра (буквы I или U).

Вывод. Наличие букв W (write) или P (print) после значения параметра ветви говорит о том, что значения напряжения или тока следует вывести на периферийное устройство.

В качестве примера ниже приведен набор данных для стабилизированного источника тока, изображенного на рис. 2.19, с проводимостью $G_{\partial a} = 1$ мСм. Данные размещаются на программной дискете в файле под именем $KSTROMA.DAT$, а на монитор выводятся значения напряжения и тока ветви с проводимостью $G_{\partial a}$. Файл $KSTROMB.DAT$ служит для расчета параметров ветви с проводимостью $G_{\partial b} = 2$ мСм.

```
NWA—G
Uquel 10 20
R1 14 68000
R2 40 68000
Rbas 45 1600
Uqbas 53 0.62
R3 30 1750
S4 23 280 SZ Rbas SG I
R5 23 42500
G6a 12 0.002 W
```

Нелинейные двухполюсники. Для идентификаторов нелинейных двухполюсников C, G, L, R второй буквой должна быть N . За номером узла следует начальное значение тока или напряжение для итерации. Затем через несколько пробелов задается идентификатор последовательности опорных точек для интерполяции нелинейных характеристик.

Опорные точки нелинейной характеристики могут находиться в любом месте набора данных. Каждая строка последовательности опорных точек начинается с идентификатора. За его первой буквой N может следовать несколько символов, однако машина воспринимает только первые четыре из них. Для нескольких однотипных линейных двухполюсников достаточно только один раз в наборе данных задать последовательность опорных точек.

После идентификатора задается список опорных точек: сначала значение x , а затем y . При этом максимально допустимым является задание 20 пар значений. Завершает список опорных точек граничное условие (символы RB), после чего следуют заглавные буквы A или B . В отличие от задания двухполюсников, которое описывается одной строкой, задание нелинейных характеристик может состоять из нескольких строк, каждая из которых начинается идентификатором. Заглавные буквы FZ в конце строки указывают на строку продолжения.

В качестве примера ниже приводится набор данных для схемы на рис. 5.22.

```

NWA-G
TrCE 30 1.25 SZ GNbas W
Uquel 40 24
STran 30 225 SZ GNbas SG I
N107 0 0 0.55 0 0.6 5.E-6 0.65 20.E-6 0.7 75.E-6 F2
N107 0.75 180.E-6 0.8 400.E-6 RB:B
B4 12 22.E3
R5 20 1000
GNbas 20 0.75 N107 W
Rcol 13 470 W
R7 41 100
```

Считывание данных. После запуска программы и считывания набора данных с дискеты определяются следующие величины: количество ветвей, количество идеальных источников напряжения, количество узлов. Далее проверяется нумерация узлов. Кроме того, выясняется, соответствует ли каждому зависимому узлу единственный идеальный источник напряжения, и в противном случае выдается сообщение об ошибке. После считывания данных программа сигнализирует об этом и выводит на монитор тип цепи.

Цепи постоянного тока. Здесь индуктивность представляет собой коротку, а емкость — разрыв цепи [12]. Программа обеспечивает необходимые преобразования цепи; емкость не рассматривается в качестве ветви, а индуктивность преобразуется в идеальный источник напряжения с $U_q=0$ (см. § 2.2).

После окончания расчета для нелинейных двухполюсников путем нажатия клавиши S задаются начальные значения. Только изменяя начальное значение функции, можно определить, имеет ли она несколько решений. В противном случае считается новый набор данных (клавиша D) или прерывается выполнение программы (клавиша A).

Линейные цепи синусоидального напряжения рассчитываются

с помощью комплексных чисел. Собственный параметр идеального источника задается в R -форме, причем его численное значение соответствует вещественной части. Мнимая часть должна быть задана дополнительно после букв IM. Результаты вычислений выводятся на периферийное устройство в R -форме. Частота, постоянная для всех двухполюсников, изменяется оператором цикла. Для этого с клавиатуры вводятся ее начальное и конечное значения, а также шаг.

После завершения расчета путем нажатия на клавишу F может быть получено следующее значение частоты. В противном случае считывается новый набор данных (клавиша D) или прерывается выполнение программы (клавиша A). В качестве примера ниже приводится набор данных для полосового фильтра (см. рис. 3.13). На каждой частоте выводятся значения напряжения и тока ветви с проводимостью G_4 .

```

NWA-S
Uq 8 10 1 IM 0
R1 12 22000
C7 42 0.1 E-6
R3 20 56
C6 32 0.1 E-6
R2 43 82000
G5 03 1.0E-6
S4 40 -100 SZ G5 SG U
G4 40 0.001 W

```

Динамические цепи. Кроме задания идентификатора, номеров узлов и значений параметра двухполюсника в наборе данных для динамической цепи требуется дополнительная информация об идеальных источниках, а также емкости C или индуктивности L . В последнем случае после букв AU должны быть заданы начальные значения (для емкости — напряжение ветви, для индуктивности — ток ветви) в момент времени $t=0$.

Для каждого идеального источника тока и каждого идеального источника напряжения задается тип временной зависимости с помощью определенной буквы и двоеточия в виде

K : — параметр источника постоянен во времени;
 S : — параметр источника синусоидально зависит от времени;
 L : — параметр источника линейно возрастает до амплитудного (пикового) значения и далее остается постоянным.

Если параметры источника синусоидально зависят от времени, то значением параметра является амплитуда. При этом после букв KR должна быть задана круговая частота в герцах, а после букв NU — начальный фазовый угол. В случае линейной временной зависимости после букв ST задается наклон характеристики (A/c или B/c), а после букв AC — отрезок оси, определяющий значение параметра двухполюсника в момент времени $t=0$.

В качестве примера ниже приводится набор данных для расчета динамической цепи. В каждый момент времени выдаются значения напряжения и тока ветви для сопротивления R_2 .

```

NWA-D
Uquelle 4 0 1 K:
R1 4 3 50
L3 3 2 0.009 AN 0
L4 2 0 0.0171 AN 0
L5 21 1 0.0009 AN 0
B2 1 0 50 W

```


После считывания набора данных с клавиатуры вводятся начальное приращение по времени, максимальный шаг приращения и конечное время. Расчет выполняется всегда для $k=4$ с пределом $EB=0.01$ согласно методу обратного дифференцирования.

После окончания анализа или при нажатии клавиши *Z* вводится следующее значение времени, или считывается новый файл данных (клавиша *D*), или прерывается выполнение программы (клавиша *A*). В заключение приводится распечатка набора данных для схемы на рис. 6.8:

NWA-D

Uquel 3 0 21 S: KR 314.15 NU 0

R4 3 2 12

GNdio 2 1 0.7 N1007

N1007 —1000 —1.E—5 .6 .6E—8 . 7 .04 .8.3 .91 1 2 RB: B

C2 1 0 470E—6 AN 0

R3 1 0 68 W

В первой строке набора данных после четырех символов должна задаваться буква, определяющая тип программы:

G—цепь постоянного тока;

S—линейная цепь синусоидального напряжения;

D—динамическая цепь.

В остальных строках располагаются либо последовательность опорных точек, либо описание двухполюсника. Каждая строка начинается с идентификатора, первой буквой которого для последовательности опорных точек является буква *N*. В случае двухполюсника первая буква определяет его тип. Для нелинейного двухполюсника второй буквой идентификатора должна быть буква *N*.

После номеров узлов и значений параметров двухполюсников задаются дополнительные данные:

AC — отрезок оси (тип программы *D*, тип двухполюсника *I* или *U* при параметре источника *L*);

AN — начальное значение (тип программы *D*, тип двухполюсника *C* или *L*);

FZ — указание на строку продолжения в последовательности опорных точек;

IM — мнимая часть (тип программы *S*, тип двухполюсника *I* или *U*);

K: — параметр источника постоянный во времени (тип программы *D*);

KR — круговая частота (тип программы *D*, тип двухполюсника *I* или *U* при параметре источника *S*);

L: — параметр источника, линейно зависящий от времени (тип программы *D*);

NU — начальный фазовый угол (тип программы *D*, тип двухполюсника *I* или *U* при *S*);

P — вывод параметров ветвей на печать;

RB — граничные условия (завершение последовательности опорных точек);

S: — параметр источника, синусоидально зависящий от времени (тип программы *D*);

SG — коэффициент управления (тип двухполюсника *S*);

ST — наклон (тип программы *D*, тип двухполюсника *I* или *U* при *L*);

SZ — управляющая цепь (тип двухполюсника *S* или *T*);

W — вывод параметров ветвей на монитор.

Список литературы

1. *Albrecht P.* Die numerische Bexandlung gewöhnlicher Differentialgleichungen, München, 1979.
2. *Brayton R., Gustavson F., Hachtel G.* A New Efficient Algorithm for Solving Differential-Algebraic Systems Using Implicit Backward Differentiation Formulas. Proceedings of the IEEE. Vol. 60, N 1, Jan. 1972. PP. 98—108.
3. *Bowers J. C., Sedore S. R.* SCEPTRE: A Computer Program Circuit and Sustems Analysis. Englewood Cliffs (NJ) 1971.
4. *Brauch W.* Programmierung mit BASIC. Stuttgart, 1982.
5. *Brown P., Senffleben D.* Über BASIC zu Pascal. Würzburg, 1984.
6. *Calahan D.* Computer—Aided Network Design. New York, 1972.
7. *Chua L., Lin P.* Computer — Aided Analysis of Electronic Circuits: Algorithms and Computational Techniques. Englewood Cliffs (NJ), 1975.
8. *Early J. M.* Effects of Space — Charge Layer Widening in Junction Transistors. Proceedings IRE, Vol. 40. P. 1401. Nov., 1952.
9. *Ebers J. J., Moll J. L.* Large — signal behavior of junction transistors. Proceedings IRE, Vol. 42, 1954. P. 1761—1772.
10. *Eckhardt H.* Numerische Verfahren in der Enèrgietechnik. Stuttgart, 1978.
11. *Effenberger E.* Beitrag zum «W—G—Kapazitätsverhalten» von Aluminium — Elektrolyt — Kondensatoren. Elektronikpraxis, Würzburg, 12, 1969. S. 9—21.
12. *Fuhrer A., Heidemann K., Nerreter W.* Grundgebiete der Elektrotechnik. 2. Bande. München, 1986.
13. *Finckenstein, Graf F.* von Einführung in die numerische Mathematik. 2 Bande. München, 1977, 1978.
14. *Funkhouser R.* BASIC auf dem IBM-PC. München, 1984.
15. *Germain C.* Das Programmierhandbuch für den IBM PC und XT. München, 1985.
16. *Goldstein L., Goldstein M.* Goldsteins IBM PC Buch. München, 1985.
17. *Gummel H. K., Poon H. C.* An integral charge control model of bipolar transistors. Bell Syst. Techn. Journal. Vol. 49, 1970. P. 827—852.
18. *Hainer K.* Numerik mit BASIC-Tischrechnern. Stuttgart, 1983.
19. *Hamming R. W.* Numerical Methods for Scientists and Engineers. New York, 1973.
20. *Heinhold J., Behringer F., Gaede K. W.* Einführung in die Höhere Mashematik. 4 Bane. München, 1976—1980.

21. *Heinzel W.* Arbeitsplatzrechner. München, 1984.
22. *Herrmann D.* Numerische Mathematik — 40 BASIC-Programme. Braunschweig, 1983.
23. *Herskovitz G. J., Schilling R. B.* Semiconductor device Modeling for Computer Aided Design. New York, 1972.
24. *Hoefer E. E., Nielinger H.* SPICE. Berlin, 1985.
25. *Heitzinger W., Troch I., Volentin G.* Praxis nichtlinearer Gleichungen. München, 1984.
26. *Heinze W.* Transient — Netzwerkanalyse mit dem Knotenverfahren bei Verwendung beliebiger Integrationsverfahren und — variablen. Nachrichtentechnische Zeitschrift 26, 1973. H. 2. S. 67—69.
27. *Jensen R., Watkins B.* Network Analysis. Englewood Cliffs (NJ), 1974.
28. *Joepgen H.* Turbo-Pascal. München, 1985.
29. *Jurgensmeier G.* Basic gestern, heute und morgen. PC Magazin Nr. 21, 15.05.1986, S. 55—61.
30. *Kleine Enzyklopadie Mathematik.* Frankfurt/M, 1985.
31. *Kremer H.* Numerische Berechnung linearer Netzwerke und Systeme. Berlin, 1978.
32. *Kupfmüller K.* Einführung in die theoretische Elektrotechnik. Berlin, 1984.
33. *Kunz H., Kraft W.* Bipolartransistoren und ihre Anwendungen. Zürich, 1983.
34. *Lange D.* Standardprogramme der Netzwerkanalyse für BASIC-Taschencomputer. Braunschweig, 1982.
35. *May G.* Strukturiertes Programmieren mit FORTRAN. München, 1980.
36. *McCracken D.* Computerpraxis mit FORTRAN 77 in Naturwissenschaft und Technik, München, 1985.
37. *Mellert P.* Rechnergestützter Entwurf elektrischer Schaltungen. München, 1981.
38. *Meyer W., Schacht K.* Das große BASIC-Lernbuch. München, 1985.
39. *Mildenberger D.* Analyse elektronischer Schaltkreise. Zwei Bände. Heidelberg, 1975.
40. *Nagel L. W., Pederson D. O.* SPICE (Simulation Program with Integrated Circuit Emphasis). Berkeley, Cal.: University of California. Electronics Research Laboratory. Memorandum ERL—M382, Apr. 2, 1973.
41. *Naunin D.* Einführung in die Netzwerktheorie. Braunschweig, 1985.
42. *Norton P.* Die verborgenen Möglichkeiten des IBM PC. München, 1985.
43. *Norton P.* MS—DOS und PC—DOS. München, 1985.
44. *Partosch G.* PASCAL auf PC. München, 1986.
45. *Paul R.* Grundlagenlehrbuch Elektrotechnik, Band II: Netzwerke. Berlin, 1985.
46. *Platz G.* Methoden der Software—Entwicklung. München, 1983.
47. *Rollke K.* Das Turbo—Pascal Buch. Düsseldorf, 1985.
48. *Sauer R., Szabo I.* Mathematische Hilfsmittel des Ingenieurs. 4 Bände. Berlin, 1967—1970.
49. *Selder H.* Einführung in die Numerische Mathematik für Ingenieure. München, 1979.

50. *Shampine L., Gordon M.* Computer—Lösung gewöhnlicher Differentialgleichungen. Wiesbaden, 1984.

51. *Shoup T. E.* Numerische Verfahren für Arbeitsplatzrechner. München, 1984.

52. *Singer F.* Programmieren in der Praxis. Stuttgart, 1980.

53. *Spiro H.* Simulation integrierter Schaltungen. München, 1985.

54. *Tholl H.* Bauelemente der Halbleiterelektronik. 2. Bande. Stuttgart, 1976, 1978.

55. *Tietze U., Schenk C.* Halbleiter-Schaltungstechnik. Berlin, 1985.

56. *Unbehauen R.* Elektrische Netzwerke. Berlin, 1981.

57. *Vaske P.* Praktische Kennlinienapproximation in BASIC. Stuttgart, 1986.

58. *Vlach J., Singhal K.* Computer Methods for Circuit Analysis and Design. New York, 1983.

59. *Wilson I. R., Addyman A. M.* PASCAL. München, 1984.

60. *Wilkinson J. H., Reinsch C.* Linear Algebra. Berlin, 1971.

61. *Wolfe M., Koelling C. P.* BASIC—Programme aus Naturwissenschaft und Technik München, 1985.

62. *Wolf H.* Lineare Systeme und Netzwerke. Berlin, 1978.

63. *Zurmühl R.* Praktische Mathematik. Berlin, 1965.

Содержание

Предисловие	3
Предисловие к русскому изданию	5
Предисловие к немецкому изданию	7
Глава 1. Применение персональных ЭВМ для расчета электрических цепей	9
1.1. Персональные ЭВМ	9
1.2. Языки программирования	10
1.3. Типы цепей	16
Глава 2. Линейные цепи постоянного тока	18
2.1. Линейные двухполюсные цепи	18
2.2. Расширенный анализ узлов	23
2.3. Цепи с линейным четырехполюсником	28
2.4. Решение системы линейных уравнений	36
2.5. Программа GLEICH на языке БЕЙСИК	44
Глава 3. Линейные цепи синусоидального напряжения	58
3.1. Комплексные величины	58
3.2. Комплексные вычисления	63
3.3. Программа SINUS на языке БЕЙСИК	71
Глава 4. Линейные динамические цепи	87
4.1. Структура динамической цепи	87
4.2. Анализ во времени	88
4.3. Метод Эйлера	90
4.4. Метод обратного дифференцирования	97
4.5. Программа DYNA-LIN на языке БЕЙСИК	106
Глава 5. Нелинейные цепи постоянного напряжения	121
5.1. Интерполяция	121
5.2. Моделирование схемных элементов	137
5.3. Метод Ньютона	141
5.4. Программа NEWTON на языке БЕЙСИК	148
Глава 6. Нелинейные динамические цепи	156
6.1. Нелинейные двухполюсные накопители энергии	156
6.2. Начальное значение	161
6.3. Программа DYNA-NL на языке БЕЙСИК	167
Глава 7. Программы на языке ПАСКАЛЬ	178

7.1. Основная программа	178
7.2. Ввод исходных данных с клавиатуры	178
7.3. Цепь постоянного тока	180
7.4. Линейная цепь	180
7.5. Динамическая цепь	181
7.6. Распечатка программы	181

Приложения

П1. Указатель программ на языке БЕЙСИК	206
П2. Указатель подпрограмм на языке БЕЙСИК	206
П3. Переменные, используемые в программах	207
П4. Указатель функций и процедур программы NWA (версия 1.0)	209
П5. Константы и переменные программы NWA (версия 1.0)	209
П6. Версия 2.0 программы NWA на языке ПАСКАЛЬ	210
Список литературы	215

Производственно-практическое издание

НЕРРЕТЕР ВОЛЬФГАНГ

РАСЧЕТ ЭЛЕКТРИЧЕСКИХ ЦЕПЕЙ НА ПЕРСОНАЛЬНОЙ ЭВМ

Заведующий редакцией **А. Б. Желдыбин**

Редактор издательства **А. А. Устинов**

Художник переплета **Т. Н. Хромова**

Художественные редакторы **Б. Н. Тумин, Г. И. Панфилова**

Технический редактор **О. Д. Кузнецова**

Корректор **Л. С. Тимохова**

ИБ № 3143

Сдано в набор 05.04.90 Подписано в печать 14.01.91 Формат 84×108^{1/32}
Бумага типографская № 2 Гарнитура литературная Печать высокая Усл.
печ. л. 11,76 Усл. кр.-отт. 11,97 Уч.-изд. л. 13,81 Тираж 69 000 экз. Заказ
№ 537 Цена 1 р. 50 к.

Энергоатомиздат, 113114, Москва, М-114, Шлюзовая наб., 10
Владимирская типография Госкомитета СССР по печати, 600000, г. Влади-
мир, Октябрьский проспект, д. 7.

Выходят в свет в 1991 году

Нюрманн Д. Измерительные приборы для домашней лаборатории: Пер. с нем. — 5,5 л.

Основное назначение книги — помочь любителю электронной техники оснастить свою измерительную лабораторию для налаживания современных электронных устройств. Кратко изложены сведения, необходимые для проведения простых измерений в домашних условиях. Рассмотрены принципы действия, схемы и конструкции измерительного моста, индикатора сигналов, звукового диапазона частот, мультиметра, индикаторов напряжения и частоты для любительского сигнал-генератора, функционального генератора. Схемы приведенных устройств доступны для повторения самым широким кругам радиолюбителей.

Для начинающих и квалифицированных любителей-конструкторов, интересующихся практической электроникой.

Бриндли К. Измерительные преобразователи: Справочное пособие. Пер. с англ. — 10 л.

Описаны принципы действия, конструкции и параметры первичных измерительных преобразователей электрических, механических, химических и оптических величин, а также датчиков давления и температуры. Автор ряда популярных книг по измерительной технике знакомит читателя с простейшими преобразователями, используемыми в измерительных и управляющих системах производственного назначения. Рассмотрены условия их применения и особенности монтажа.

Для широкого круга читателей, не обладающих специальной подготовкой в области из-

мерительной техники, конструкторов-любителей.

Крибель Х. Схемы любительских электронных устройств: Пер. с нем. — 13 л.

Подробно описаны схемы 30 электронных устройств: бытовой и автомобильной электроники, радио- и звукотехники, самодвижущихся игрушек и моделей, фото- и кинотехники, измерительных приборов (в том числе детектора ионизирующих излучений), источников электропитания (в том числе на солнечных элементах). Схемы выполнены на стандартных элементах и опробованы на практике. Приведена таблица используемых в схемах полупроводниковых приборов и их отечественных аналогов.

Для начинающих радиолюбителей, конструкторов электронной техники, а также лиц, занимающихся самостоятельным техническим творчеством.

Практическое руководство по расчетам схем в электронике: Справочник: В 2 т. Т. 1: Пер. с англ. — 45 л.

В первом томе приведены основы анализа электрических цепей, характеристики элементов, правила выбора полупроводниковых приборов. Включены главы по усилителям звуковой частоты, резонансным усилителям, цепям обратной связи, генераторам сигналов, источникам питания. Даны сведения по операционным усилителям. Изложены основы цифровой и видеотехники.

Для широкого круга специалистов в области электротехники, электроники и радиотехники, а также для радиолюбителей.

Практическое руководство по расчетам схем в электронике: Справочник: В 2 т. Т. 2: Пер с англ. — 40 л.

Во втором томе рассмотрены принципы построения микропроцессорных систем, передачи информации в аналоговой и цифровой фор-

мах, основы электроизмерений. Включены главы по линиям связи, фильтрам, основным типам направленных антенн и СВЧ-линий передачи. Приведены сведения по толстопленочной технологии и волоконно-оптическим системам.

Для широкого круга специалистов в области электротехники, электроники и радиотехники, а также для радиолюбителей.

Йоргенсен Ф. Справочная книга по магнитной записи: В 2 т. Т. 1: Пер. с англ. — 40 л.

В первом томе справочника методически удачно изложены основы теории и техники намагничивания. В разделах «Магнитные головки» и «Магнитные ленты и диски» охвачены все представляющие практический интерес аспекты теории, свойства материалов, изготовления и использования. Большое число иллюстраций, графиков и таблиц позволяет читателю быстро усвоить принципы технологии, причины брака и повреждений магнитного носителя и головок.

Для широкого круга специалистов, любителей и пользователей, желающих углубить свои познания в области магнитной записи и памяти на магнитном носителе.

Йоргенсен Ф. Справочная книга по магнитной записи: В 2 т. Т. 2: Пер. с англ. — 40 л.

Во втором томе справочника рассмотрены типы лентопротяжных механизмов, дисководов, взаимодействие головок с магнитным покрытием. Описаны процессы записи и воспроизведения, технические характеристики и электрические параметры устройств записи/чтения. Приведены сведения о видеомагнитофонах, использующих магнитные ленты и диски, компьютерных запоминающих устройствах, принципах параллельной записи, форматирования дискет. Даны рекомендации по эксплуа-

тации и анализу причин повреждений техники магнитной записи/чтения.

Для широкого круга специалистов, любителей и пользователей, желающих углубить свои познания в области магнитной записи и памяти на магнитном носителе.

Холленд Р. Микропроцессоры и операционные системы: Краткое справочное пособие: Пер. с англ. — 15 л.

В сжатой форме даны сведения об архитектуре микропроцессоров, наборах используемых команд, интегральных схемах электронного обрамления, применяемого для управления вводом/выводом, периферийными устройствами либо играющего роль запоминающих устройств. Описаны три класса широкоиспользуемых семейств микропроцессоров (8-, 16- и 32-разрядные) и три класса операционных систем для микроЭВМ.

Для широкого круга читателей — от студентов до инженеров-программистов и разработчиков.

Уильямс В. Мощная электроника: приборы, управление, применение: Справочное пособие: Пер. с англ. — 28 л.

Приведены основные сведения о мощных полупроводниковых приборах — диодах, биполярных и полевых транзисторах, тиристорах. Рассмотрены условия работы этих приборов на нагрузку, их параллельные и последовательные схемы соединения. Дан анализ применения мощных полупроводниковых приборов в устройствах преобразования электрической энергии: регуляторах напряжения, тиристорных прерывателях, статических преобразователях, ключевых источниках электропитания.

Для инженерно-технических работников, занимающихся разработкой и применением мощных полупроводниковых приборов в системах преобразования электрической энергии.

1р.50к.

