

MathSpice – аналитический PSpice-движок для OrCAD и MicroCAP

Часть 2. Директивы MSpice

Олег Петраков (Москва)

Пакет расширений Maple под названием MathSpice (MSpice) предназначен для аналитического решения электронных цепей и функциональных схем, но может быть использован как инструмент создания Spice-моделей сигналов и электронных приборов для различных симуляторов. Во второй статье цикла описаны основные директивы MSpice.

Для получения решения MSpice использует всего две основные директивы – ESolve() и Values(). Остальные директивы имеют характер Smart-утилит для быстрого выполнения некоторых типовых операций, например, построения графиков, вычисления нулей и полюсов, преобразований полученных решений. В свою очередь, директивы управляются опциями, которые модифицируют их действие в соответствии с характером решаемой задачи. В общем случае задача решается с применением всего арсенала средств Maple. MSpice и никаких ограничений не накладывает. И если вы хотите создать переносимый документ, то не следует применять Smart-утилиты массово, так как не все располагают MSpice. Либо надо принимать меры по консервации сгенерированных MSpice данных, например скопировать их через буфер в новую строку.

ОСНОВНЫЕ ДИРЕКТИВЫ MSPICE

К основным директивам относятся ESolve() с опциями Analysis и Devices, и Values(). Директива ESolve() читает NET-листинг, составляет уравнения Кирхгофа, решает их аналитически и вводит решения в рабочий лист Maple. Директива Values() вводит в рабочий лист Maple значения номиналов компонентов по схеме и присваивает их именам компонентов. После применения директивы Values() возможно получение численных решений.

ESolve() – директива чтения NET-листинга и получения решений

Это главная директива пакета. Она служит для вызова NET-листинга ис-

следуемой электрической цепи в рабочий лист Maple, составления системы уравнений Кирхгофа и вывода решения цепи в аналитическом виде. Директива ESolve() многорежимная и управляется опцией Analysis. Эта опция указывает директиве метод аналитического решения задачи, представленной NET-листингом. Опция Analysis указывается внутри директивы. Опция Devices внешняя. Она должна быть указана до применения ESolve(). Опция Devices указывает тип моделей (эквивалентных схем замещения) электронных приборов, которые используются вместо условных графических изображений (УГО) транзисторов, операционных усилителей, ламп. При отсутствии опции Devices используются модели по умолчанию, самые простые и самые употребительные. Кроме этого, опция Devices позволяет вывести схему модели прибора на экран в виде оформленного рисунка, что немало важно для пояснения хода расчётов и публикации в прессе.

Обобщённый формат ввода директивы ESolve():

```
> Devices:=[Одинаковые, [тип прибора, уровень сложности, номер рисунка], [x,x,x], ... [x,x,x] ];
> ESolve(Analysis, `Путь к NET-листингу`):
```

Где:

- Devices – необязательная опция управления моделями. При её отсутствии используются модели по умолчанию;
- Analysis – опция управления способом решения задачи;

- `Путь к NET-листингу` – строчная переменная, в которой указан путь к файлу NET-листинга, записанный по правилам Maple.

Путь к листингу, в случае использования в качестве графического редактора OrCAD Capture, должен выглядеть примерно так:

```
`P:/Samples/MOS/MOS-PSpiceFiles/SCHEMATIC1/SCHEMATIC1.net`
```

Особенностью указания пути для Maple является то, что слеш должен быть направлен в обратную сторону (/) по сравнению с тем, как это принято в MS DOS (\). Операцию копирования пути легко выполнить с помощью Total Comander, а затем вручную заменить слеш на обратный.

Опция Analysis – это строка, составленная из ключевых слов (или их комбинации). Ключевые слова (Laplace, Q, Phusics, Spice, Differential, Big, Small, Not print, V) можно писать целиком, но для сокращения записи лучше использовать только большие буквы, указанные ниже. Опции Q, P, S, B, M, N, V (Q, Phusics, Spice, Big, Small, Not print, V) управляют выводом системы уравнений Кирхгофа на решение и форматом переменных. Опции L, D (Laplace, Differential) управляют способом решения задачи (либо решаются операторные уравнения, либо дифференциальные).

Формат переменных (имена узловых напряжений) может быть:

- *электрический*. При этом все переменные начинаются с символа «V» (V1, VOUT, VINP, V34). Этот формат используется по умолчанию;
- *физический*. Убирается символ «V» в именах переменных, если следующий символ не цифра (V1, OUT, INP, V34 – опция «P»);
- *Spice*. Это электрический формат, но имена узлов после символа «V» находятся в скобках (V(1), V(OUT), V(INP), V(34) – опция «S»). Этот ре-

жим полезен разработчикам PSpice-моделей. Полученные выражения могут использоваться практически любым Spice-симулятором.

Вывод уравнений на экран можно отключить (N), сделать красивым (Q), увеличить степень подробности (V). Если в именах переменных встречаются имена символов греческого алфавита, то их можно вывести либо строчными буквами (B), либо прописными (M). Для предотвращения повторного решения систем уравнений, если схема не изменялась, служит опция «write» или «wr». При наличии опции «wr» перезапись уравнений производится только после генерации нового NET-листинга. При наличии опции «write» перезапись уравнений производится постоянно (это не рабочий режим, требуется крайне редко).

Способ решения задаётся комбинацией этих опций, которые пишутся одной строкой вместо слова Analysis. Порядок следования произвольный. Например: QS – решение в операторном виде с переменными в формате PSpice; DPM – решение дифференциальных уравнений с переменными физического формата, при этом греческие буквы вводятся как прописные; Q – решение в операторном виде электрических задач, это наиболее употребительная опция; Qwr – решение в операторном виде электрических задач с записью (чтением) решения в файл.

Используются следующие символы для сокращённого формирования опции Analysis:

- Analysis = "L"aplace: по умолчанию (метод Лапласа);
- Analysis = "Q": метод Лапласа с красивой печатью уравнений (можно использовать вместо L);
- Analysis = "P"ysics: представление переменных в физическом формате;
- Analysis = "S"pice: представление переменных в формате PSpice;
- Analysis = "D"ifferential: решение в дифференциальных уравнениях;
- Analysis = "B"ig: преобразование греческих символов в строчный шрифт;
- Analysis = s"M"all: преобразование греческих символов в прописной шрифт;
- Analysis = "N"ot print: подавление вывода уравнений на экран;
- Analysis = "V": вывод на экран первичных уравнений;

- Analysis = «write» («wr»): запись решения в файл с последующим чтением решения из файла. Это позволяет избежать повторного решения системы уравнений, если схема не изменялась.

Пример:

```
ESolve(Q, `OB.СКТ`):
```

Важно! При запуске Maple из меню Start Windows в рабочем листе Maple в директиве ESolve() надо указать полный путь к файлу NET-листинга с указанием метки диска, например:

```
ESolve(Q, `P:/Samples/MOS_1/MOS-PSpiceFiles/SCHEMATIC1/SCHEMATIC1.net`); # для OrCAD
ESolve(Q, `k:/ВидеоусилительOB/OB.СКТ`); # для MicroCAP
```

Можно использовать укороченные относительные пути (как в первом примере), но перед этим Windows надо указать, каким именно приложением открывать ваши проекты в виде рабочих листов Maple (указать надо примерно так c:\Maple 11\bin.win\cwmaple.exe). После этого открывать эти проекты надо щелчком мыши по имени файла.

Работу над большой задачей можно ускорить, используя запись решения системы уравнений в файл, например:

```
ESolve(Qwr, `k:/Видео усилительOB/OB.СКТ`),
```

где wr – режим однократной записи в файл и многократного чтения; write – режим принудительной записи в файл.

Это очень полезная возможность, если проект содержит сложную схему. В этом случае уравнения решаются долго. Однако задача в целом требует многократного перезапуска программы с целью анализа результатов. При этом схема может долго не изменяться. Чтобы повторно не составлять и не решать систему уравнений (это может быть и час), используйте эту опцию. Если вы измените схему и сгенерируете новый NET-листинг, то MSpice это заметит. Он обнаружит и считает новый NET-листинг, после чего выполнит решение задачи заново и опять запишет его в файл, ожидая следующей коррекции схемы.

Опция Devices присваивается значению списка, который содержит подопцию одинаковости параметров приборов в пределах одного типа и список

указателей моделей. В этом списке моделей надо указать только те модели, которые не являются используемыми по умолчанию.

Обобщённая форма записи опции Devices (Приборы) выглядит так:

```
Devices:=[Одинаковые, [тип прибора, уровень сложности модели, номер рисунка], [x,x,x], ... [x,x,x] ];
```

Важно! Опция Devices указывается до применения директивы ESolve().

Одинаковые – указатель на одинаковость параметров схемы замещения приборы. Синоним слова «Одинаковые»: O, o, Equal, equal, Equ, equ, E. При отсутствии слова «Одинаковые» (или синонима) параметры моделей всех приборов становятся разными, например:

```
Devices:=[ [BJT, AC1, 2] ] .
```

Так, если в схеме три транзистора, то в уравнениях появятся девять переменных: β1, Rэб1, Сэб1, β2, Rэб2, Сэб2, β3, Rэб3, Сэб3; в противном случае только три: β, Rэб, Сэб.

Перечень типов приборов:

- BJT, JFET, MOS, GaAs, IGBT – транзисторы;
- OP – стандартные операционные усилители;
- TOP – трансимпедансные операционные усилители;
- TRIODE – лампа-триод.

По умолчанию Mspice использует русские обозначения параметров схем замещения приборов. Если в указателе типа приборов присутствует сочетание символов «en», то обозначения параметров будут английские (BJTen, IGBTen, OPen, TRIODEen).

Перечень уровней сложности моделей:

- для транзисторов: DC1, DC2, DC3, DF, AC1, AC2, AC3;
- для ОУ: DC1, DC2, DC3, DC4, AC1, AC2, AC3, AC4.

Номер рисунка – это просто цифры, означающие номер рисунка, например: 3, 16, 3.23, 23.5. Если они указаны в списке, то в текст выводится пронумерованный рисунок со схемой замещения указанного в этом списке прибора, если не указаны, то рисунок не выводится.

Важно! По умолчанию, при отсутствии опции Devices модели всех приборов имеют уровень сложности DC1, параметры схем замещения разные, используются русские обозначения параметров. При рисовании электронных схем принято использовать услов-

ные графические обозначения (УГО), это делает их наглядными и простыми. При проведении аналитических расчётов схем, содержащих электронные приборы, вместо УГО в уравнения подставляются эквивалентные схемы замещения прибора. Общее число моделей, поддерживаемых MSpice, около сотни. Поэтому моделями надо управлять. Модели распределены на две большие группы – модели постоянного тока (DC) и модели переменного тока (AC). Внутри групп они различаются по уровню сложности – 1, 2, 3, 4.

По умолчанию, когда опция Devices не указана, используются простейшие модели (DC1). Более точные решения можно получить, используя модели более высокого уровня сложности DC(2...4), AC(1...4). Именно в этом случае используется опция Devices. В проекте можно сочетать разные приборы. Подопция одинаковости указывает на одинаковость параметров эквивалентной схемы в пределах одного типа. Например, в схеме используются три транзистора с одинаковым β , Rэб, Сэб или с разными β_1 , Rэб1, Сэб1, β_2 , Rэб2, Сэб2, β_3 , Rэб3, Сэб3. Естественно, в первом случае уравнения будут проще.

Действие подопции «Одинаковые» глобально для всех приборов (в том числе и для используемых по умолчанию, т.е. не указанных в опции Devices). Вместо опции Devices можно использовать опцию Приборы, это одно и то же.

Пример:

```
Приборы := [Одинаково-
вые, [VJTен, AC1, 2.13]] :
```

Расшифруем содержимое этой опции:

- Одинаковые – будут использоваться модели приборов с одинаковыми параметрами;
- VJTен – «en» указание на английские обозначения параметров модели VJT (биполярного) транзистора;
- AC1 – указание на инерционную модель прибора, уровень сложности – 1;
- 2.13 – цифры в конце списка. Если они есть, то на экран выводится пронумерованный рисунок со схемой модели прибора, на который можно давать ссылки в сопровождающем работу тексте. Это позволяет избежать неоднозначности толкования моделей; что вам нарисовала эта директива, то вы и увидите в уравнениях.

Values() – директива ввода в проект номиналов компонентов со схемы

Директива Esolve() вводит в рабочий лист Maple символьное решение. Для получения численных результатов вместо обозначений компонентов надо подставить их номиналы, которые указаны на схеме. Именно для этого служит директива Values(). Кроме этого, директива вводит параметры встроенных в Mspice моделей электронных приборов и некоторые физические константы, если это необходимо.

Обобщённый формат ввода директивы Values():

```
Values(x1, x2, x3) :
```

Где:

- x1, x2, x3 – указатели (модификаторы) формата вводимых директивой данных о номиналах;
- x1 – указатель типа расчётов: постоянный (DC), переменный (AC) ток, переходные процессы (laplace);
- x2 – указатель на разрешение ввода номиналов. Так, RLCVI – разрешает ввести все номиналы. Если убрать часть букв, то номиналы соответствующих групп компонентов не будут введены и останутся в символьном виде. Если в этой строчке присутствует символ «P», то номиналы компонентов будут введены в рабочий лист Maple, но печать на экран будет подавлена. Во многих случаях это оправдано, так как практически всё указано на схеме, которую через буфер Windows можно поместить в рабочий лист Maple;
- x3 – список указателей на компоненты, номиналы которых вводить запрещено. Например: x3= [«Rнагрузки»]. Тогда вы можете построить нагрузочную характеристику по переменной Rнагрузки. Примеры допустимых значений указателей:
- x1=DC – ввод DC/AC-источников с DC-значениями;
- x1=AC – ввод DC/AC-источников с AC-значениями;
- x1=laplace – выполняется преобразование Лапласа по переменной t;
- x1=laplace_T – выполняется преобразование Лапласа по переменной T;
- x1=laplace_eta – выполняется преобразование Лапласа по произвольной переменной, например eta;
- x2=RLC – вводятся элементы R, L, C, не выводятся V, I;

- x2=RLCVI – вводятся элементы R, L, C, V, I;
- x2=PRLCVI – вводятся элементы R, L, C, V, I, но без печати на дисплей;
- x3=[«R14», «Vinp»] – запрет вывода номиналов R14, Vinp, которые следует поместить в двойные кавычки. Это предохранит имена компонентов от модификации, если вы их ранее переопределили;
- x3=[] – отмена всех частных запретов вывода номиналов компонентов.

После вывода спецификации всем перечисленным в ней переменным присваиваются значения номиналов компонентов схемы. При этом все дальнейшие расчёты будут уже численными, если не принять специальных мер.

К специальным мерам относится запрет ввода номиналов некоторых компонентов, если вы собираетесь построить от них какую-либо зависимость. Пример использования директивы Values:

```
> Values(AC, RLCVI, []) ;
```

Ввод номиналов компонентов:

```
R5 := .100e6, "100K"
C4 := .47e-7, "47n"
R4 := .1e4, "1K"
R1 := 480, "480"
C1 := .51e-10, «51p»
R6 := 38, "38"
R2 := 480, "480"
R7 := 962, "962"
R3 := 240, "240"
C2 := .51e-10, "51p"
C3 := .102e-9, "102p"
beta := 100, "100"
`Fт` := .100e6, "100e3"
`Jэ` := .1e-2
`Rэб` := k*T/e*(beta+1)/`Jэ`
`Rэб` := k*T/e*(beta+1)/`Jэ`
e := .1602176462e-18
k := .1380650277e-22
T := 300, "300"
AC источник: DC: Vвх:=0 AC:
Vвх:=1 Pphase(degrees):=0
DC источник: DC: VB:=12 AC: VB:=0
H_Q2_КТ3102В := (V5-
VOUT)*beta/`Rэб`
H_Q1_КТ3102В := (V6-V4)*beta/`Rэб`
```

Ввод источников имеет особенности. Если установлен ключ DC, то выводятся только DC-составляющая всех источников. Если установлен ключ AC, то на экран выводятся DC- и AC-составляющие, а переменным присваиваются значения AC. Так, для чисто DC-ис-

точников составляющая АС всегда равна нулю.

При сравнении результатов с PSpice следует помнить, что в PSpice рабочая точка, в которой находится электронный прибор, зависит от величин DC-источников. Например, если вы хотите построить АЧХ транзисторного фильтра с помощью PSpice, то она будет несколько разной при разных питающих напряжениях.

При аналитических расчётах, если вы желаете добиться идеального совпадения результатов с PSpice, вы должны сами позаботиться о задании параметров схем замещения электронных приборов для рабочей точки. О том, как создаются такие модели, написано в разделе 3.2 Модели реальных приборов для аналитических расчётов.

ВСПОМОГАТЕЛЬНЫЕ ДИРЕКТИВЫ MSpICE

Эти директивы носят характер Smart-утилит для быстрого выполнения некоторых типовых задач, например построения графиков. Чаще всего их достаточно, чтобы оформить работу, пригодную для публикации. Для получения более продвинутой визуализации

или анализа решения следует использовать функции Maple. Помимо ускорения работы, применение этих директив позволяет сделать листинги Maple с электрическими расчётами более короткими и приятными на глаз.

IMG() – директива построения графиков

Директива позволяет быстро построить график функции (или списка функций) с приемлемым для публикации качеством оформления рисунка. Для построения графиков используются следующие шкалы в терминах Maple: semilog, log10, logplot, loglogplot. Кроме этого формируется типовая надпись к рисунку с номером, легенда, имена переменных.

Обобщённый формат ввода директивы:

```
IMG([список функций], Имя переменной =начало..конец, «пояснительный текст к рисунку»):
```

Или более коротко:

```
IMG([F1, F2, ..., Fn], Var=Low..Hig, "txt"):
```

Где:

- [F1, F2, ..., Fn] – список функций;
- Var – имя переменной;
- Low – начальное значение переменной;
- Hig – конечное значение переменной;
- «txt» – надпись к графику.

Строка «txt» может содержать управляющие ключевые слова и структуры, которые на печать выводятся не всегда, но влияют на способ построения графика. Поддерживаются следующие ключевые слова и структуры:

- №) – номер рисунка;
- semi, log10, log, loglog – тип шкал по осям X и Y при выводе графика;
- del – подавление печати слова или структуры, написанных за del;
- \$ – указатель количества точек на графике (например, \$1000 – тысяча точек);
- [F1, F2, ..., Fn] – список функций для формирования легенды. В списке могут фигурировать любые разумные пояснения к графикам, позволяющие в них разобраться, но количество элементов в списке должно быть равно количеству функций, иначе сформируется легенда по умолчанию.

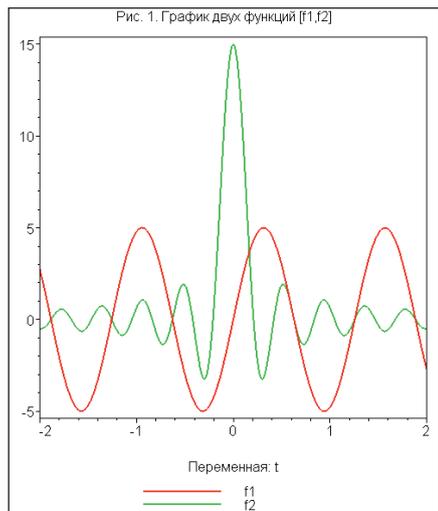


Рис. 1. График двух функций

Пример (рис. 1):

```
> f1:=5*sin(5*t):
> f2:=sin(15*t)/t:
> IMG([f1,f2],t=-2..2,"1) График
двух функций [f1,f2]");
```

HSF() – директива построения частотных характеристик операторных функций

Для линейных цепей построение частотных характеристик – типовая задача. Mspice решает такие задачи только в операторном виде. Для профессиональной работы иметь операторные функции вполне достаточно, так как переход в частотную область – задача тривиальная. Однако проект при этом засоряется громоздкими выражениями, но ничего нового не добавляется. Директива HSF() автоматически переводит указанные в ней операторные функции в частотную область. Если в директиву помещена частотная функция $K(f)$ или $K(\omega)$, её график тоже будет построен, но она должна соответствовать имени переменной Var. Это функция очень удобна, поскольку экономит время и делает рабочий лист Maple более коротким. Для построения АЧХ используются следующие шкалы в терминах Maple: semilog, log10, logplot, loglogplot. Кроме этого формируется типовая надпись к рисунку, легенда, имена переменных. Несмотря на скромный вид, она весьма универсальна и позволяет избегать громоздких записей при использовании стандартной команды plot(). Если не использовать в качестве переменных символы f и ω , то она будет работать как директива IMG().

Шаблон директивы:

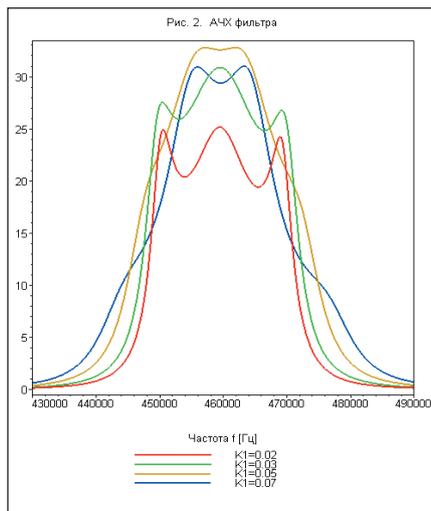


Рис. 2. АЧХ фильтра

```
HSF ([список функций], имя =начало..конец, «пояснительный текст»):
```

Или более коротко:

```
HSF ([H1, H2, ..., Hn], Var=Low..Hig, "txt"):
```

Где:

- [H1, H2 .. Hn] – список операторных функций;
- Var – имя переменной. Для построения АЧХ обязательно использовать имена f или ω ;
- Low – начальное значение переменной;
- Hig – конечное значение переменной;
- «txt» – надпись к графику.

Строка «txt» может содержать управляющие ключевые слова и структуры, которые на печать выводятся не всегда, но влияют на способ построения графика. Поддерживаются следующие ключевые слова и структуры:

- № – номер рисунка;
- semi, log10, log, loglog – тип шкал по осям X и Y при выводе графика;
- del – подавление печати слова, написанного за del;
- \$ – указатель количества точек на графике (например, \$1000 – тысяча точек);
- [H1, H2, ..., Hn] – список функций для формирования легенды. В списке могут фигурировать любые разумные пояснения к графикам, позволяющие в них разобраться, но количество элементов в списке должно быть равно количеству функций, иначе сформируется легенда по умолчанию.

В приводимых ниже примерах сами графики не показаны. Показано толь-

ко, как формируются надписи и легенды в зависимости от ключевых слов:

```
HSF ([H1, H2, K1], f=1..1e6, "1)
semi [H1, H2, K1] усилителя"):
Рис. 1. АЧХ усилителя
```

(При этом на графике будут указаны легенды H1, H2, K1.)

```
HSF ([H1, H2, K1], omega=1..10e8, "1)
semi [H1, H2, K1] усилителя на круговой частоте):
```

Рис. 1. АЧХ [H1, H2, K1] усилителя на круговой частоте

(При этом на графике будут указаны легенды H1, H2, K1.)

```
HSF ([H1, H2, K1], f=1..1e6, "1) semi
[H1, H2, K1] $1000 усилителя"):
Рис. 1. АЧХ [H1, H2, K1] усилителя
```

(При этом на графике будут указаны легенды H1, H2, K1. Будет выводиться тысяча точек, это необходимо для вывода острых резонансных кривых.)

```
HSF ([sin(100*t), VOUT], t=1..1e6, "2
.2.1) del [sin(100*t), VOUT] Выход
усилителя"):
Рис. 2. 2. 1. Выход усилителя
```

Рис. 2. 2. 1. Выход усилителя

(При этом на графике будут указаны легенды sit(100*t), VOUT. Если имя переменной не f или ω , то HSF работает как IMG. В принципе IMG можно не использовать.)

Приведём пример сложного графика (рис. 2), построенного с помощью HSF() для полосового радиопередатчика:

```
>
[eval (OUT, K1=0.02), eval (OUT, K1=0.03), eval (OUT, K1=0.05), eval (OUT, K1=0.07)]:
HSF (% , f=4.3e5..4.9e5, "2) АЧХ
фильтра
del ['K1=0.02', 'K1=0.03', 'K1=0.05', 'K1=0.07']");
```

Очевидно, что такого рисунка вполне достаточно для анализа АЧХ, причём многовариантного.

BodeAVdB(), BodePphase() – директивы построения диаграммы Боде

Эти директивы предназначены для вывода логарифмических частотных характеристик в децибелах одиночной операторной или частотной функции.

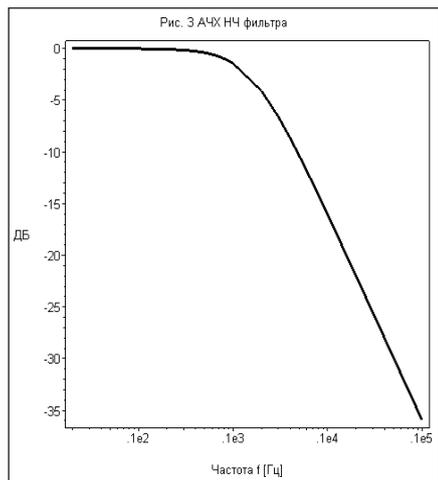


Рис. 3. АЧХ НЧ-фильтра

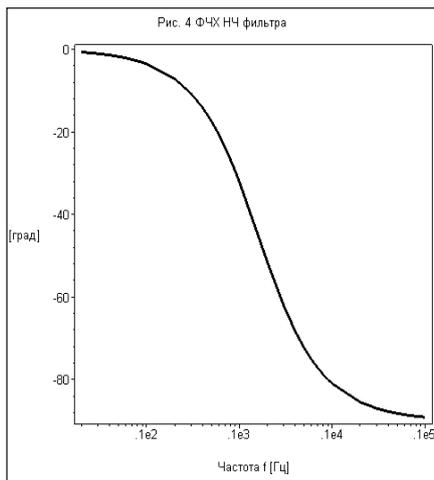


Рис. 4. ФЧХ НЧ-фильтра

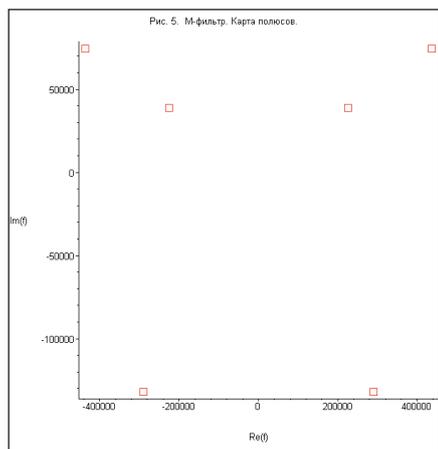


Рис. 5. Карта полюсов М-фильтра

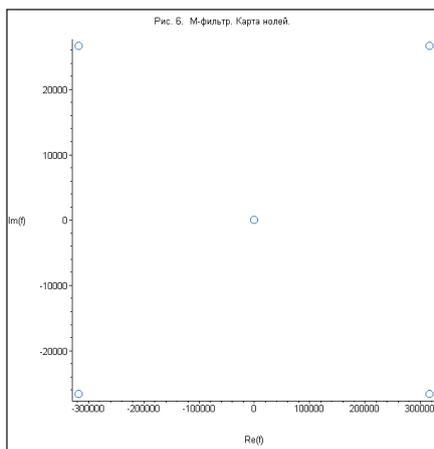


Рис. 6. Карта нулей М-фильтра

Директивы автоматически переводят указанные в ней операторные функции в частотную область. Эта функция очень удобна, поскольку экономит время и делает рабочий лист Maple более коротким. Кроме этого формируется типовая надпись к рисунку, легенда, имена переменных. В этом они сходны с директивой HSF, описанной выше.

Формат ввода:

```

BodeAVdB(H,Var=Low..Hig,txt): -
Построение диаграмм Боде для амплитуды
BodePfase(H,Var=Low..Hig,txt): -
Построение диаграмм Боде для фазы
    
```

Где:

- H – операторная или частотная функция;
- Var – имя переменной (f или omega);
- Low – начальное значение переменной (в декадах);
- Hig – конечное значение переменной (в декадах);
- txt – надпись к рисунку.

Пример построения диаграммы Боде для простейшего НЧ-фильтра в виде RC-цепочки (рис. 3, 4):

```

> H:=1/(0.02(I*Pi*f+1);
BodeAVdB(H,f=0..5,"Рис. 3 АЧХ НЧ
фильтра");
BodePfase(H,f=0..5,"Рис. 4 ФЧХ НЧ
фильтра");
    
```

PoleZero(), PlotPZ() – директивы построения карт нулей и полюсов

Директива PoleZero() выводит полюсы и нули в аналитическом виде, а после использования директивы Values() вычисляет их численное значение.

Директивы PlotPZ() предназначены для графического вывода частотных или комплексных нулей и полюсов одиночной операторной функции. Директивы автоматизируют достаточно рутинные операции по поиску корней полиномов. Их применение экономит время и делает рабочий лист Maple более коротким. Кроме этого формируется типовая надпись к рисунку, легенда, имена переменных. В этом они сходны с директивой HSF, описанной выше.

Формат записи директив:

```

PoleZero(H,Var): - аналитический
поиск нулей и полюсов
    
```

PlotPZ(H,Var,"txt"): - карта нулей и полюсов в виде графика

Где:

- H – операторная или частотная функция;
- Var – имя переменной (s, f или omega);
- txt – надпись к рисунку (директива PoleZero() не нуждается в этой опции).

В приводимом ниже примере в качестве переменной указана частота f, поэтому для операторной функции H(s) будет выведена карта полюсов (рис. 5) и нулей (рис. 6) в частотной области. Для специалиста по фильтрам видеть такую карту радостнее, чем звёздное небо в самую тёмную ночь.

```
> PlotPZ(H,f,"5 М-фильтр.");
```

PlotModel() – директива печати моделей приборов

MSpice поддерживает около сотни моделей электронных приборов, базирующихся на общепринятых в ведущих учебниках для вузов моделях. Они различаются по уровню сложности. Сложные модели используются при изучении отдельных транзисторных каскадов усиления. Простые модели используются при исследовании устройств с массовым использованием транзисторов, когда применение сложных моделей не позволяет получить результат приемлемой сложности. Директива PlotModel() вводит в проект графическое изображение моделей, которые затем используется для пояснений расчётов. Этой директивой изображение модели можно вставить в произвольное место рабочего листа Maple. Можно распечатать всё сразу, например, для обзора моделей.

Обобщённый формат директивы:

```

PlotModel([тип прибора, уровень
сложности модели, номер рисунка],
... , [x,x,x]);
    
```

Указание для печати конкретных моделей в рабочий лист задаётся в виде списка списков, в котором можно перечислить всё, что вас интересует в данный момент. Все указатели в этой директиве аналогичны указателям опции Devises, описанной выше, за исключением поддержки подопции одинаковости. Здесь нет никакого смысла в одинаковости парамет-

ров схем замещения электронных приборов.

Примеры использования директивы для обзора моделей:

```
PlotModel ([MOS, DC, AC, AC2, AC3, DC3, DC2, 2]);
PlotModel ([DMOS, DC, AC, AC2, AC3, DC3, DC2, 2]);
PlotModel ([JFET, DC, AC, AC2, AC3, DC3, DC2, 2]);
PlotModel ([BJT, DC, AC, AC2, AC3, DC3, DC2, DF, 2]);
PlotModel ([GaAs, DC, AC, AC2, AC3, DC3, DC2, 1]);
PlotModel ([IGBT, DC, AC, AC2, AC3, DC3, DC2, 1]);
PlotModel ([OP, DC, AC, AC2, AC3, DC3, DC2, 1]);
PlotModel ([TOP, DC, AC, AC2, AC3, DC3, DC2, 1]);
PlotModel ([TRIODE, DC1, DC2, DC3, AC1, AC2, AC3, 1]);
PlotModel ([NOT], "Список наиболее употребительных нотаций");
```

Пример вывода информации о AC3 модели IGBT транзистора (рис. 7):

```
> PlotModel ([IGBT, AC3, 7]);
```

AV(), AVM(), AVP() AVPdeg() – директивы перевода операторных функций в частотную область

Это наиболее востребованные операции при исследовании линейных цепей. Ничего интересного в этих преобразованиях нет, но в явном виде они загромождают рабочий лист. Это интересно разве что новичку в ТОЭ. К использованию этих функций быстро привыкаешь, поскольку они позволяют коротко и наглядно записывать уравнения для поиска различных характерных точек частотных функций. Имя переменной в этой директиве играет роль модификатора. В зависимости от имени используется радиальная или круговая частота, децибелы для модуля АЧХ.

Формат ввода директив:

```
AV(H, Var)
AVM(H, Var)
AVP(H, Var)
AVPdeg(H, Var)
```

Где: H – операторная функция, Var – имя переменной.

Допустимые имена переменных:

- Var=f – вывод модуля (аргумента) функции на круговой частоте;

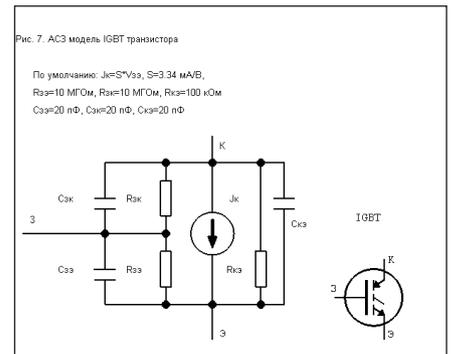


Рис. 7. AC3-модель IGBT-транзистора

- Var=omega – вывод модуля (аргумента) функции на радиальной частоте;
- Var=dBf – вывод модуля функции на круговой частоте в децибелах;
- Var=dBomega – вывод модуля функции на радиальной частоте в децибелах.

Директива AV() переводит операторную функцию, указанную в ней, в частотную область. При этом выполняется замена операторной переменной. Если Var=f, то выполняется замена $s = \Gamma^2 \cdot \Pi \cdot f$. Если Var=omega, то выполняется замена $s = \Gamma \cdot \omega$. Все остальные директивы, описанные ниже, производят аналогичные действия.

Директива AVM() переводит операторную функцию в частотную область

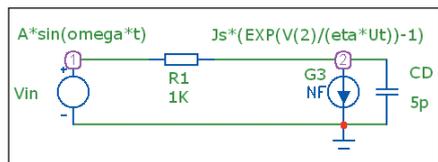


Рис. 8. Диодная схема

и извлекает модуль комплексной частотной функции. При этом если в имени переменной есть символы dB, модуль будет в децибелах.

Директива AVP() переводит операторную функцию в частотную область и извлекает аргумент комплексной частотной функции в радианах.

Директива AVPdeg() переводит операторную функцию в частотную область и извлекает аргумент комплексной частотной функции в градусах.

convNET(), pspice() – директивы-конверторы форматов PSpice

Эти директивы позволяют вернуть результаты аналитических расчетов в PSpice. Такая задача возникает у разработчиков моделей электронных приборов для симуляторов различных САПР. Maple позволяет эффективно решить любую физическую задачу, являясь профессиональным инструментом для математических исследований. Но численные методы, реализованные в нём, не так быстры и приспособлены для электрических расчётов, как у PSpice. Тогда систему уравнений, соответствующую новому прибору, можно оформить в виде макромодели, присвоить условное графическое изображение и использовать в PSpice-проектах. Такое взаимодействие Maple и PSpice может быть чрезвычайно плодотворным, если вы отличный физик, математик и электронщик в одном лице.

Формат ввода директив:

```
convNET(H, option): - транслятор форматов PSpice.
pspice(H): - преобразование формул в формат PSpice
```

Где:

- H – произвольная функция (в рамках поддержки MSpice);
 - option – модификатор действия директивы. Допустимые значения модификаторов для option: pspice, maple, IFmaple, model.
- Назначение модификаторов:
- pspice – трансляция в формат PSpice;

- maple – трансляция в формат Maple;
 - IFmaple – трансляция в формат Maple без преобразования условного оператора IF в `if`;
 - model – генерация Pspice-модели.
- Пример преобразования форматов PSpice и Maple:

```
> F := `if`(5 <
-15.00*sin(2*Pi*t), 5, `if`(-
15.00*sin(2*Pi*t) < -5, -5, -
15.00*sin(2*Pi*t)));
> pspice(F);
"{IF(5<-
15.00*SIN(2*Pi*time), 5, IF(-
15.00*SIN(2*Pi*time)<-5.-5.-16.00
*SIN(2*Pi*time)))}"
> convNET(F, pspice);
"{IF(5<-
15.00*SIN(2*Pi*time), 5, IF(-
15.00*SIN(2*Pi*time)<-5.-5.-16.00
*SIN(2*Pi*time)))}"
```

Пример генерации PSpice-модели:

```
> P1:=0.0015*V(1,2)^3-
0.007*V(1,2)^2+0.0085*V(1,2):
> P2:=subs(Vd=V(1,2), P1);
> convNET(P2, model_DTUNEL);
".SUBCKT 1 2 DTUNEL"
"G_GDTUNEL 1 2 VALUE { .150e-
2*V(1,2)^3-.700e-
2*V(1,2)^2+.850e-2*V(1,2) }"
".ENDS"
```

Важно! Имя модели указывается после символа подчёркивания в опции model. Например, мы хотим создать модель PSpice-компонента с именем DTUNEL, тогда опция будет model_DTUNEL.

FSource() – директива генерации файловых сигналов для PSpice

PSpice-моделирование невозможно без источников сигналов. В сложных технических системах используются сложные сигналы, например с шумом. Набор сигналов в PSpice велик, но не безграничен. Можно изготовить какой-то особенный сигнал с помощью богатейшего набора функций Maple и в виде файла передать его в PSpice.

Формат директивы:

```
FSource(VALUE, t=Low..Hig,
Step):
```

Где:

- VALUE – произвольная функция зависимости напряжения от времени;
- t – имя переменной (время);

- Low – начальное значение переменной;
- Hig – конечное значение переменной;
- Step – шаг дискретизации сигнала по времени.

Пример:

```
> restart:with(MSpice):with(RandomTools):
> s:=GetState():
SetState(state=s):
>
VOUT:=`if`(sin(100*t)>0,sin(100*t)
)+0.707e-12*Generate(integer,0);
> FSource(VOUT,t=0.001..0.2,1e-4);
```

После выполнения директивы Fsource() в текущем каталоге появится файл с именем FSource.txt, который можно использовать в PSpice.

convQma(), convMma() – трансляторы Mathematica

Выход в свет Mathematica-7.0 ясно показал, что она явный претендент в лидеры среди СКМ. На мой взгляд, Maple более прост в использовании и для электрических расчётов его более чем достаточно. Однако надо использовать и наработки для Mathematica, которые в изобилии есть в Интернете. Это позволяет решить задачу быстро и с минимальными усилиями, если вам подвернулся готовый шаблон решения, пусть даже в другой среде. Профессиональная жизнь коротка, и таким способом можно несколько увеличить плотность событий в ней.

Формат записи директив:

```
convQma(ODE, ISc):
```

– транслятор систем уравнений Кирхгофа из формата Maple в формат Mathematica:

```
convMma(H, option):
```

– двунаправленный транслятор выражений в формат Mathematica или Maple; где:

- ODE – система уравнений Кирхгофа в формате set;
- ISc – список переменных в формате set;
- H – операторная или произвольная функция;
- Option – опция указания направления преобразования. Если option=M, то производится преобразование из формата Mathematica в формат Maple. Если вместо option записан любой

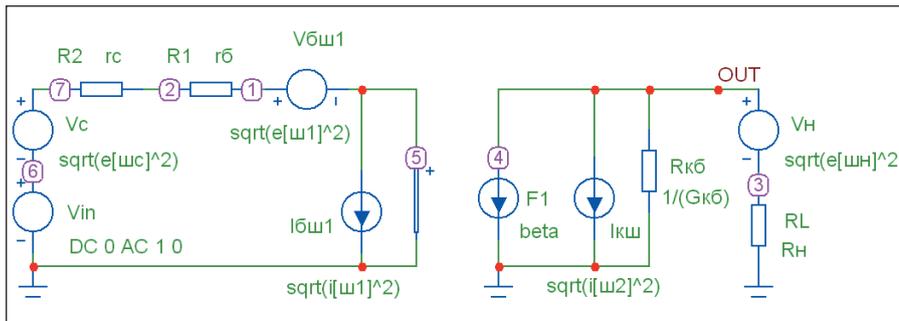


Рис. 9. Шумовая модель биполярного транзистора

другой символ (например, «m»), то производится преобразование из формата Maple в формат Mathematica.

Пример – преобразование системы уравнений из формата Maple в формат Mathematica:

```
> restart: with(MSpice):
> ODE:={ (V2-V4)/R2-(V4-V1)*s*C2,
-V5/R5-(V5-A2*(V4-A2*V4/(1+A2)))/R4};
> ISc:={V1, V2, V4, V5};
ISc:={V2, V1, V4, V5}
> convQma(ODE, ISc);
```

Уравнения в формате Mathematica:

```
"Solve[{(V2-V4)/R2-(V4-V1)*s*C2
== 0, -V5/R5-(V5-A2*(V4-A2*V4/(1+A2)))/R4 == 0}, {V2, V4,
V1, V5}]"
```

Формулы в схемах MicroCAP и OrCAD

MSpice допускает ввод формул в схему во многих случаях, даже в тех, где это неочевидно. Просто надо пробовать. Наиболее актуален и эффективен ввод формул для пассивных компонентов R, L, C и источников I и V. Поэтому для них предусмотрен некоторый сервис.

Возможность ввода формул вместо позиционных обозначений и номиналов компонентов придаёт E Solver и MSpice невероятную гибкость и позволяет установить самые замысловатые связи между элементами схемы. При этом очень часто задача становится более простой и выразительной для глаз («Всё написано на бумаге»), чем использование библиотечных функциональных блоков.

В зависимости от того, как введена формула, она проходит разный путь для получения решения цепи. В первом случае формула сразу подставляется в систему уравнений Кирхгофа, а потом ищется решение. Во втором сначала решается система уравнений Кирхгофа, а затем в решения подставляются формулы. Второй путь не всегда срабатыва-

ет, если через формулы образуются обратные. Тогда может возникнуть закливание при перекомпоновке решений.

Если вводить формулы вместо позиционного обозначения компонента, то это исключает возможность закливания программы, поскольку они ассимилированы в систему. Однако прямой ввод формул таким способом исключает ввод чисел с плавающей точкой. Это ограничение накладывает транслятор MicroCAP (OrCAD). Тогда следует использовать приставки «eq» и «Q» при обозначении компонентов. При этом формулы, записанные в позиции «номинал», интерпретируются как формулы позиционных обозначений. Приставка «Q» используется только для токов, поскольку они часто используются для моделирования транзисторов. В этом случае ограничения на вид формул снимаются. Такой способ ввода наиболее предпочтителен, так как, добавляя или убирая приставку, мы можем выбирать способ решения задачи, не переставляя формулы из одного места в другое. Показаны диодная (рис. 8) и транзисторная (рис. 9) схемы с формулами; это весьма удобный способ описания задачи. Источник G3 – это диод, представленный с помощью зависимого источника тока.

Следует отметить, что графические редакторы OrCAD и MicroCAP не очень приспособлены для ввода формул. Средствами Maple некоторые недостатки удалось несколько компенсировать. И если вы пользуетесь вводом сложных формул на схему, то после запуска MSpice внимательно просматривайте уравнения Кирхгофа и контролируйте их прохождение. В качестве имён переменных надо выбирать такие, которые не искажаются при трансляции в NET-листинг, либо надо это учитывать в дальнейших расчётах, главное – не проглядеть. Отметим, что при наличии формул наиболее корректно работает OrCAD.

Отметим следующие особенности трансляции формул со схемы в NET-листинг:

- прописные буквы всегда транслируются в NET-листинг как заглавные. Это во многом обедняет формулы;
- если формула имеет первым символом знак, то её следует заключить в скобки;
- в именах узлов недопустимы формулы и конструкции типа W[x], lambda(vx), lambda[vx]. В этом случае MicroCAP подставляет номер узла по умолчанию. Такое имя не появится в NET-листинге MicroCAP и OrCAD.

