

**Объявляется набор в магистратуру
в 2021г.**

По направлению:

12.04.01 Приборостроение

Направленность (профиль) программы:

(Технологии цифрового города)

Все вопросы по почте:

ankravets@mail.ru

Кафедра радиоприемных устройств и
телевидения

Методические указания к лабораторной работе

**Создание и добавление в библиотеку
макромоделей**

по курсу

**Основы компьютерного
проектирования и моделирования РЭС**

Для студентов радиотехнического факультета всех
форм обучения



УДК 621.396.6(07.07)

Кравец А.В., Клименко В.В. Методические указания к лабораторной работе «Создание и добавление в библиотеку макромоделей» по курсу «Основы компьютерного проектирования и моделирования РЭС». – Таганрог: Изд-во ТТИ ЮФУ, 2009. – 16 с.

В данном руководстве изложено краткое описание лабораторной работы и краткие сведения о системе схемотехнического моделирования Micro-Cap 8. Приведены домашние и лабораторные задания, методические указания по выполнению лабораторного задания.

Лабораторная работа соответствует программам курса «Основы компьютерного проектирования и моделирования РЭС» для студентов всех форм обучения по направлению «Радиотехника».

Табл. 2. Ил. 5. Библиогр.: 6 назв.

Рецензент В.Т. Лобач, канд. техн. наук, профессор кафедры РТС ТТИ ЮФУ.

Введение

Данное методическое пособие является кратким учебным курсом, призванным помочь студентам быстро освоить систему Micro-Cap 8.0 (далее MC8). Пособие можно использовать как практическое дополнение к другим имеющимся источникам справочной информации по системе MC8.

Программа MC8 является средой автоматического проектирования, предназначенной для синтеза и анализа электронных схем. Она позволяет моделировать аналоговые, цифровые и смешанные электронные устройства. Программа не требует загрузки других программ для выполнения анализа.

Цель работы

Целью данной лабораторной работы является изучение основных особенностей моделирования радиотехнических цепей и процессов, протекающих в них; приобретение практических навыков работы в среде автоматического проектирования Micro-Cap 8.0.

Краткие теоретические сведения

В программе MC8 имеются макромодели двух типов. Макромодели типа **Macro** задаются *схемой замещения*, **Subckt** – *текстовым описанием*.

Макромодели, заданные схемами замещения (Macro), представляют собой схемный эквивалент текстовых описаний подсхем на языке SPICE (subcircuits). Как правило, они представляют собой функционально законченные блоки электрических схем, созданные и сохраненные на диске для последующего использования в других схемах.

Атрибуты:

PART: <имя> – позиционное обозначение элемента

FILE: <имя макромодели> – задает имя файла совпадающее с именем макромодели, кроме того, в круглых скобках () можно задавать список вызываемых параметров через запятую.

Param:KA=<значение параметра> и

Param:KB=<значение параметра> – атрибуты, которые задают численное значение параметров, а также их имена, перечисленные в директиве **.PARAMETERS(KA,KB)**

Макромодели, заданные при помощи директивы **.subckt**

Атрибуты:

PART: <имя> – позиционное обозначение элемента.

NAME: <имя подсхемы> – задается при помощи директивы **.SUBCKT**

FILE: <имя файла> – задает имя текстового SPICE-файла, в котором находится определение нужной подсхемы, начинающееся с директивы **.SUBCKT**. Программа MC ищет указанную директиву для подсхемы под заданным именем **NAME** в следующих перечисленных местах в той же последовательности:

- 1) в текстовой области схемного файла;
- 2) в файле, заданном в поле атрибута **FILE**;
- 3) в одном или более файлах, перечисленных в директиве **.LIB**; в одном или нескольких файлах, перечисленных в директиве, выполняющейся каждый раз по умолчанию '**.LIB NOM.LIB**' (фактически в файле **NOM.LIB** и во всех файлах, которые в нем указаны с помощью соответствующих директив **.LIB**).

PARAMS: [<имя параметра>=<значение>] <Имя параметра> представляет собой имя параметра, <значение параметра> определяет величину, которая будет присвоена параметру при вызове подсхемы.

TEXT: [<имя текста>=<значение>] – этот параметр

определяет текстовый файл импортируемый в текстовое описание схемы.

Домашнее задание

Изучить теоретический материал по конспекту лекций и предложенному списку литературы.

Провести анализ предстоящей работы с использованием настоящего методического руководства.

Лабораторное задание

1. Создайте макромодель сумматора согласно методическим рекомендациям по выполнению лабораторной работы (для всех вариантов задание одинаковое). Проверьте работоспособность модели в режиме **transient analysis**.
2. Создайте следующие макромодели элементов с параметрами указанными в табл. 1. Проверьте работоспособность модели в режиме **transient analysis**.

Таблица 1

Номер варианта	Тип схемы	Количество входов	Входное сопротивление
1	Сумматор	4	1k
2	Перемножитель	3	2k
3	Делитель	2	∞
4	Сумматор	5	1k
5	Вычитающее устройство	4	∞
6	Перемножитель	3	10k
7	Ограничитель сверху	1	∞
8	Ограничитель снизу	1	3k

3. Создайте макромодель элемента И с инверсией (nand) согласно методическим рекомендациям по выполнению лабораторной работы (для всех вариантов задание

одинаковое). Проверьте работоспособность модели в режиме **transient analysis**.

4. Создайте следующие макромодели элементов с параметрами указанными в табл. 2. Проверьте работоспособность модели в режиме **transient analysis**.

Таблица 2

Номер варианта	Тип схемы	Количество входов
1	and (Вентиль И)	4
2	nand (Вентиль И-НЕ)	3
3	nor (Вентиль ИЛИ-НЕ)	2
4	and (Вентиль И)	5
5	or (Вентиль ИЛИ)	4
6	nor (Вентиль ИЛИ-НЕ)	5
7	nand (Вентиль И-НЕ)	4
8	nor (Вентиль ИЛИ-НЕ)	5

Методические рекомендации по выполнению лабораторной работы

Создание макромодели, заданной схемой (macro)

1. Создать схему макромодели (рис.1) (использование функциональных источников сигнала прил. 1).
2. По команде Options>Mode/Text (вызывается также нажатием **T**) присвоить имена узлам схемы, которые должны быть выводами макромодели. Если необходимо передавать в макромодель значения параметров, имена этих параметров декларируются в директиве .PARAMETERS. Записать схему в файл, имя которого должно совпадать с именем макромодели (расширение имени .mac).

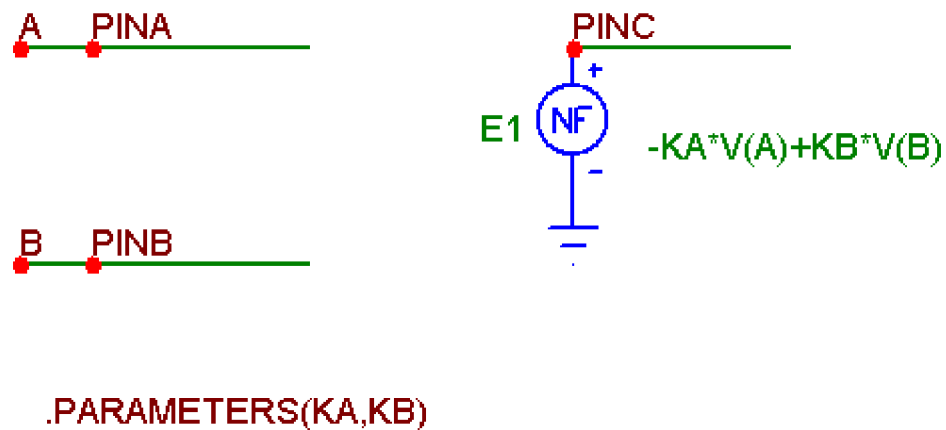



Рис. 1. Схема макроса

3. Ввести макромодель в библиотеку компонентов. Для этого:

а) создать обозначение элемента в редакторе Shape Editor для своего Макроса (прил. 2);

б) выбрать Component Editor в меню Windows;

в) выбрать группу (в правом окне Component Editor), в которую Вы хотите поместить компонент или создать новую (последнее предпочтительнее) ;

г) нажать кнопку добавления компонентов .

д) ввести имя файла (например, для сумматора) в поле Name=SUM2 – имя компонента (должно совпадать с именем файла схемы замещения);

е) выбрать созданное Вами графическое обозначение элемента в поле Shape, например Sum – имя символа количество выводов должно совпадать с количеством выводов схемы Макроса;

ж) выбрать Macro – тип макромодели в поле Definition;

и) обозначить имена выводов PINA, PINB и PINC в схеме замещения и символе макромодели должны совпадать;

к) закрыть Component Editor с сохранением всех изменений;

л) собрать схему эксперимента (рис. 2) с использованием созданного макроса и проверить его работоспособность.

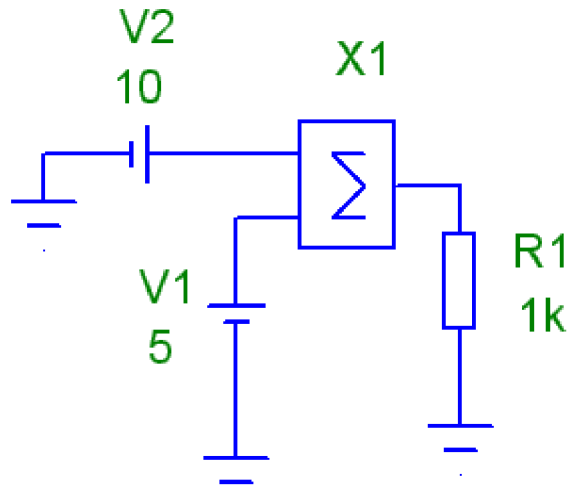



Рис. 2. Схема эксперимента

Создание макромодели, заданной текстовым описанием (subckt)

Рассмотрим на примере элемента И с инверсией (nand):

а) создать обозначение элемента в редакторе Shape Editor для своего Макроса (см. прил. 2);

б) выбрать Component Editor в меню Windows;

в) выбрать группу (в правом окне Component Editor), в которую Вы хотите поместить компонент или создайте новую (последнее предпочтительнее) ;

г) нажать кнопку добавления компонентов ;

д) ввести имя файла в поле Name= new_1 – имя компонента (например, new_1);

е) выбрать графическое обозначение элемента в поле Shape, количество выводов должно совпадать с количеством выводов в описании Макроса. Обозначение элемента можно создать произвольно в редакторе Shape Editor для своего Макроса;

ж) выбрать subckt – тип макромодели в поле Definition;

и) обозначить имена выводов, например А, В и Y в текстовом описании и символе макромодели должны совпадать;

к) выходим из редактора компонентов Component Editor сохранив результаты;

л) вызываем полученный элемент и располагаем на схеме.

В диалоговом окне создаем описание макромодели в текстовом формате:

```
.SUBCKT new_1 A B Y
```

* Звездочкой обозначаются примечания
описание входов: два входа А, В и выход Y

```
+ optional: DPWR=$G_DPWR DGND=$G_DGND
```

*Необязательные параметры: знак \$ обозначает абсолютное значение, установленное в главных настройках МС8. Формат описания параметров следующий: <буквенное название>=<абсолютная переменная> или значение.

```
+ params: MNTYMXDLY=0 IO_LEVEL=0
```

*Стандартные временные параметры, формат: <название переменной>=<значение>.

```
U1 LOGICEXP(2,1) DPWR DGND
```

*Описание логического устройства. Формат следующий: U<название> LOGICEXP (<количество входов, количество выходов>) <питание> <заземление>.

```
+A B
```

*Входы устройства.

```
+Y
```

*Выход устройства.

```
+DLY_00 IO_STD MNTYMXDLY={MNTYMXDLY}
IO_LEVEL={IO_LEVEL}
```

*Установка значений стандартных переменных.

```
+LOGIC:
```

```
+Y={~(A&B)}
```

*Логические выражения устройства. Указывается название переменной и в фигурных скобках значение. Можно указывать имена входов и выходов, как в этом примере.

Используемые знаки логических операций:

~ инверсия;

& и;

^ исключающее или;

| или;

```
.model DLY_00 ugate (tphTY=11ns tphMX=22ns
tphlTY=7ns tphlMX=15ns)
```

*Описание задержек при прохождении сигналов от входных контактов до выходных.

```
.ENDS new_1
```

*Конец программы работы подсхемы.

Проверку работы макроса можно провести на простейшей схеме (рис. 3):

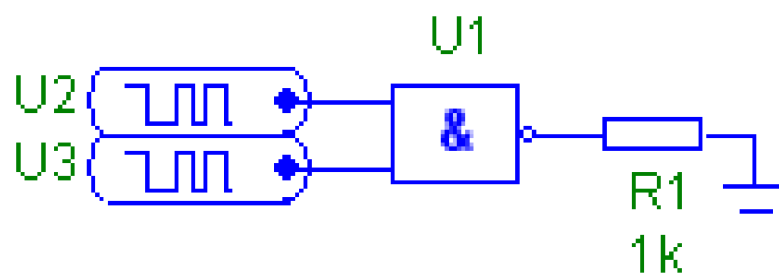


Рис. 3. Схема эксперимента


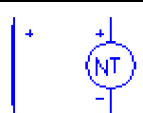
Контрольные вопросы

1. Алгоритм создания макросов (Macro).
2. Функциональные источники сигналов, их типы и способы задания параметров.
3. Алгоритм создания макросов (Subckt).
4. Команды задания логического выражения устройства.
5. Команды описания логического устройства.
6. Команды объявления необязательных параметров логического устройства.
7. Команды описания временных задержек логического устройства.
8. Команды начала и окончания описания логического устройства.

Библиографический список

1. Разевиг В.Д. Система схемотехнического моделирования MICRO-CAP V. – М.: Изд-во «Солон», 1997. – 273 с.
2. Разевиг В.Д. Система схемотехнического моделирования MICRO-CAP VI. – М.: Горячая линия – Телеком, 2001. – 344 с.
3. Разевиг В.Д. Система схемотехнического моделирования и проектирования печатных плат Design Center (PSpice). – М.: СК Пресс, 1996. – 272 с.
4. Амелина М.А., Амелин С.А. Программа схемотехнического моделирования Micro-Cap 8. – М.: Горячая линия – Телеком, 2007. – 464 с.
5. Касьянов А.Н. Micro-Cap в схемотехнике: Учебное пособие. – Тамбов: Изд-во Тамб. гос. техн. ун-та, 2004. – 112 с.
6. Вейс Л. Д. Схемотехника. Изучение и проектирование на компьютере: Учебное пособие. – Бишкек, 1999. (электронная версия).

Функциональные источники сигналов

Имя компонента	Графическое обозначение	Тип источника	Параметры
NFV		Источник напряжения, задаваемый функциональной зависимостью	Алгебраическая формула для расчета напряжения источника сигнала
NFI		Источник тока, задаваемый функциональной зависимостью	Например: $ka \cdot v(a) + kb \cdot v(b)$
NTIofI		Источник тока, управляемый током	Таблица зависимостей значений выходного сигнала от входного Например: (0,6)(1.5,8)(9,9)
NTVofI		Источник тока, управляемый напряжением	
NTIofV		Источник напряжения, управляемый током	
NTVofV		Источник напряжения, управляемый напряжением	

Пример использования источника напряжения, управляемого напряжением и задаваемого таблично, и его передаточной характеристики (рис 1.1).

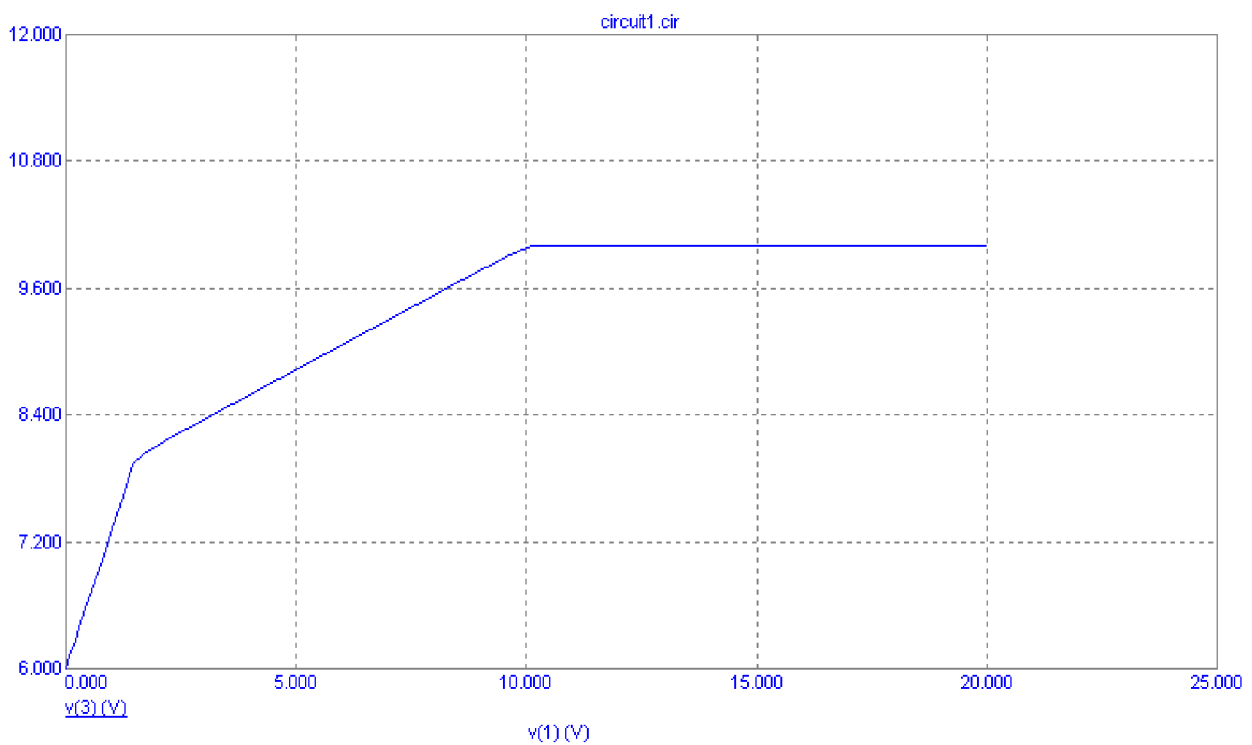
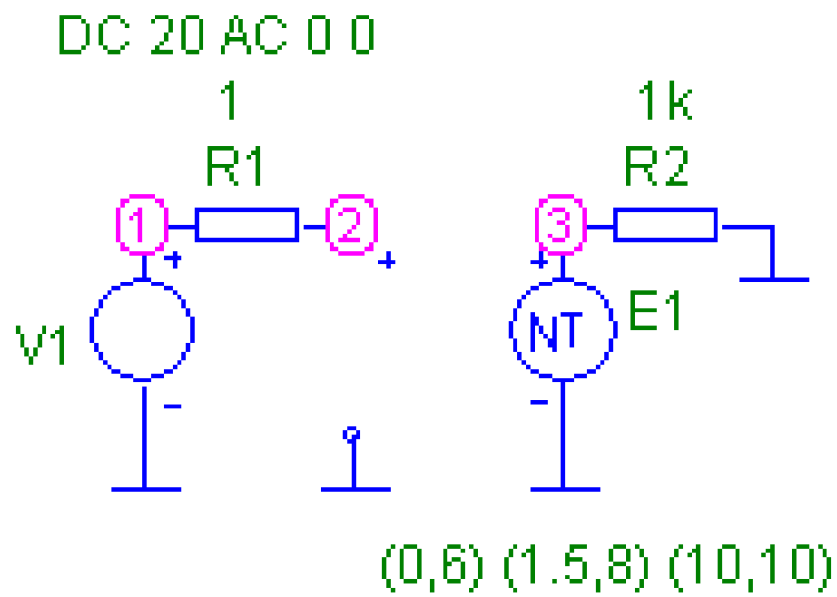


Рис. 1.1. Пример использования источника напряжения, управляемого напряжением и задаваемого таблично, и его передаточной характеристики

Редактирование графических символов компонентов

Создание и редактирование символов или условных графических обозначений компонентов принципиальных электрических схем осуществляется с помощью графического редактора, вызываемого в меню Windows по команде Shape Editor.

Структура экрана редактора графических символов Shape Editor.

После загрузки редактора Shape Editor на экране появляется окно редактирования, изображенное на рис. 2.1.

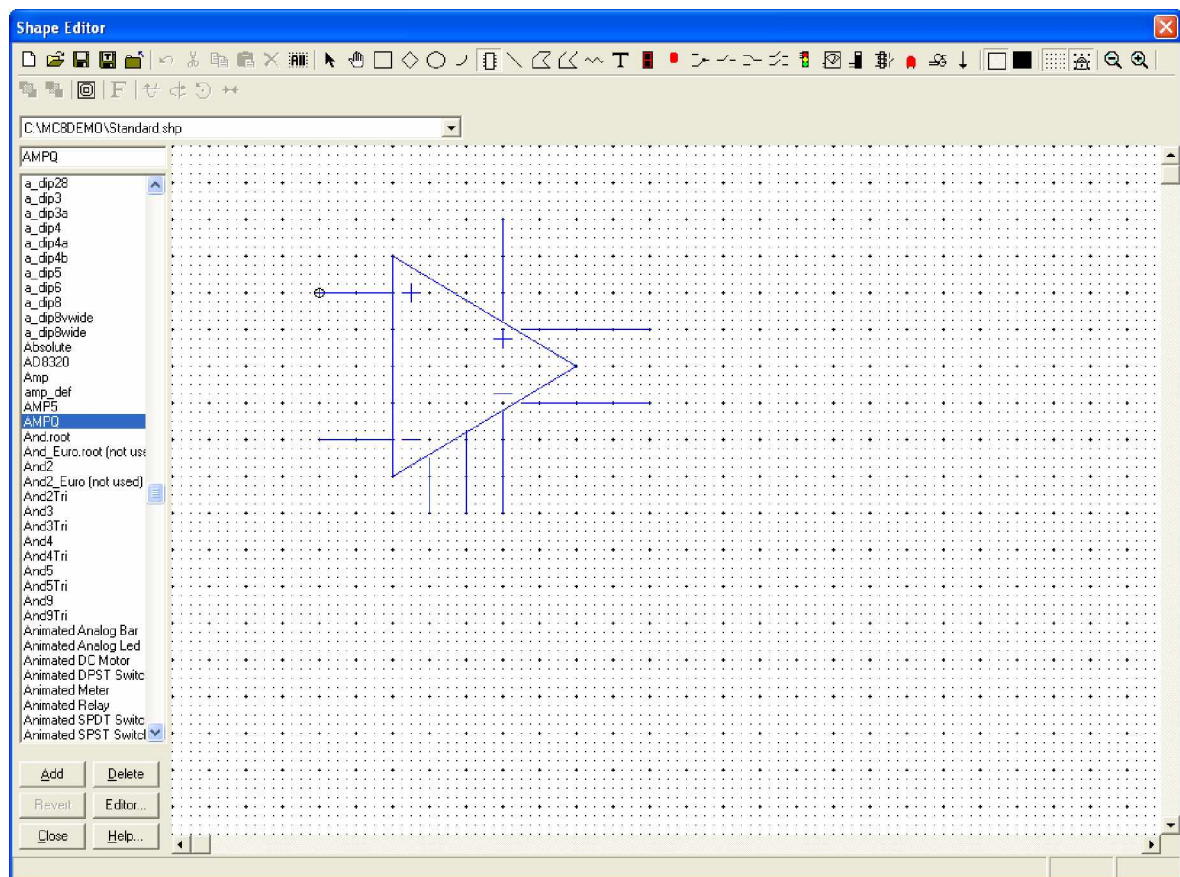


Рис. 2.1. Экран графического редактора символов Shape Editor

На нем имеются следующие панели управления, графы и окна.

Список символов. В средней левой части экрана приведен список графических символов, содержащихся в текущей библиотеке символов. Выбранный символ высвечивается и изображается в графическом окне.

Кнопки команд:

Add – добавление в библиотеку нового символа, имя которого указывается по дополнительному запросу. Графика символа задается с помощью команд меню инструментов и редактора графических объектов Object Editor (вызывается нажатием на кнопку *Editor*).

Delete – удаление выбранного символа.

Revert – восстановление первоначальной графики символа, которую он имел до вызова редактора Shape Editor. Восстановление первоначальной графики производится только для текущего выбранного символа. Изменения, произведенные при редактировании предыдущих символов, остаются в силе. Однако имеется возможность отменить все изменения, выполненные в текущем сеансе работы с редактором Shape Editor, прекратить с ним работу по команде *Close* по дополнительному запросу без сохранения изменений.

Editor – вызов редактора Object Editor, предназначенного для редактирования числовых параметров графических объектов.

Close – завершение работы с редактором символов и сохранение всех внесенных изменений в файл библиотеки.

Help – вызов раздела Shape Editor системы помощи.

