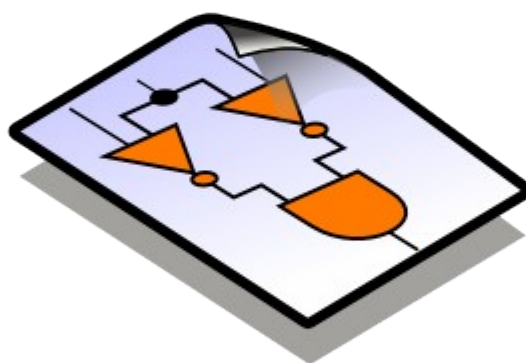


# KICAD

# EESCHEMA



# LINUX & WINDOWS

## ГРАФИЧЕСКИЙ РЕДАКТОР СХЕМ

**Программное обеспечение  
со свободной лицензией**

© Жан-Пьер Шарра (Франция)  
и KiCAD-сообщество  
программистов и  
пользователей

2010

## Содержание

1. Введение.....	4
1.1 Описание графического редактора EEschema.....	4
1.2 Технический обзор программы.....	4
2. Основные команды EEschema.....	5
2.1 Доступ к командам редактора схем.....	5
2.2 Команды, выполняемые мышкой.....	6
2.3 «Горячие» клавиши редактора схем.....	8
2.4 Выбор размера сетки.....	9
2.5 Выбор видового масштаба.....	9
2.6 Отображение координат курсора.....	9
2.7 Меню функций EEschema.....	9
2.8 Верхнее инструментальное меню.....	10
2.9 Иконки правого инструментального меню.....	11
2.9 Иконки левой инструментальной панели.....	13
2.10 Выпадающие меню для быстрого редактирования.....	13
3. Основное меню функций редактора схем EEschema.....	14
3.1 Меню функций File.....	14
3.2 Меню настроек редактора схем.....	15
3.3 Меню помощи Help.....	18
4. Основное инструментальное меню EEschema.....	20
4.1 Управление листами схемы.....	20
4.2 Инструменты поиска в схемах.....	20
4.3 Формирование списка цепей проекта (netlist).....	21
4.4 Инструмент формирования позиционных обозначений схемы.....	22
4.5 Применение инструмента ERC.....	23
4.6 Вывод перечня элементов в формате BOM.....	24
4.7 Импорт файла обратной аннотации от PCBNEW.....	25
5. Создание и редактирование схемы.....	26
5.1 Определения вариантов схем.....	26
5.2 Вводные наставления.....	26
5.3 Последовательность выполнения проекта.....	27
5.4 Размещение и редактирование компонент схемы.....	27
5.5 Проводники, шины, метки и порты питания.....	30
5.6 Элементы оформления схемы.....	36
6. Иерархические схемы.....	38
6.1 Применение.....	38
6.2 Навигация по иерархии.....	39
6.3 Локальные, иерархические и глобальные метки.....	39
6.4 Создание и заголовок иерархической схемы.....	39
6.5 Символ листа (Sheet symbol).....	40
6.6 Соединения: иерархические выводы.....	41
6.7 Соединения: иерархические метки.....	42
6.8 Комплексная иерархия схем.....	44
6.9 Плоская иерархия.....	45
7. Автоматическая классификация позиционных обозначений.....	47
7.1 Назначение функции аннотирования схемы.....	47
7.2 Пример применения.....	47
8. Проверка схемы типа ERC.....	48
8.1 Обзор функции ERC.....	48
8.2 Применение ERC-теста.....	49
8.3 Пример ERC-теста.....	49
8.4 Вывод диагностики для маркера.....	50

8.5 Выводы и флаги питания.....	51
8.3 Конфигурация для ERC-теста.....	52
8.4 Файл отчета ERC-теста.....	52
9. Создание Netlist-списка цепей схемы.....	53
9.1 Обзор функции.....	53
9.2 Формирование списка цепей Netlist для PCBNEW.....	53
9.3 Взаимодействие с программами spice-моделирования схем.....	55
9.4 Замечания по применению функции Netlist.....	57
9.5 Применение Netlist-плагинов внешних программ.....	57
10. Черчение и печать схем.....	58
10.1 Обзор функций Plot/Print.....	58
10.2 Общие команды печати.....	58
10.3 Черчение схемы в формате HPGL.....	58
10.4 Черчение схемы в формате Postscript.....	60
10.5 Черчение схемы в формате SVG.....	60
10.6 Черчение схемы в формате DXF.....	61
10.7 Печать схемы на принтере.....	61
11. Управление библиотеками компонентов схемы.....	63
11.1 Основная информация о библиотеках.....	63
11.2 Обзор состава компонентов.....	63
11.3 Загрузка компонента для редактирования.....	64
11.4 Создание компонент схемы.....	67
11.5 Разработка компонента.....	71
11.6 Создание и редактирование выводов.....	73
11.7 Редактирование полей.....	79
11.8 Создание символов порта питания.....	81
12. Дополнительные функции редактора компонент.....	82
12.1 Обзор функций.....	82
12.2 Позиционирование якоря.....	83
12.3 Псевдонимы компонентов (алиасы).....	84
12.4 Текстовые поля компонента.....	85
12.5 Документация на компонент.....	85
12.6 Файлы символов компонент.....	89
13. Визуализация библиотек.....	90
13.1 Основные функции Library Browser.....	90
13.2 Основное окно Library Browser.....	90
13.3 Инструментальная панель Library Browser.....	92
14. Ссылки.....	92
15. Послесловие к русской редакции перевода.....	93

## 1. Введение

### 1.1 Описание графического редактора EEschema

**EESchema** - это графический редактор для разработки принципиальных электрических схем, входящий в состав системы автоматизированного проектирования печатных плат KiCAD для операционных систем типа Linux и Windows. Независимо от используемой операционной системы создаваемые файлы схем полностью совместимы.

EEschema - интегрированная программа: из нее вызываются все функции управления, обслуживания библиотек, рисования и компоновки схемы и перехода к разработке печатной платы. Она допускает иерархическую разработку многостраничных схем.

EEschema работает совместно с графическим редактором проектов печатных плат и трассировки проводников PCBNEW, для которого она предоставляет файл Netlist, который описывает электрические соединения для разработки печатной платы.

EEschema также включает редактор изображений компонент (символов элементов схем) LibEdit, который позволяет создавать, редактировать и просматривать компоненты, и поддерживать библиотеки символов (операции импорта, экспорта, добавления и удаления компонент библиотеки).

EEschema дополнительно выполняет следующие, важные для современного программного обеспечения разработки схем, функции:

- Автоматизированная проверка правильности схемы (DRC) - контроль за правильностью ввода компонент и соединений, неподключенными компонентами и др.
- Генерация чертежей схем в форматах POSTSCRIPT, HPGL или DXF.
- Распечатка схем на локальном принтере.
- Формирование перечня материалов.
- Формирование списков цепей (netlist) для моделирования электрической схемы проекта и проектирования печатной платы

### 1.2 Технический обзор программы

EESchema – 32-битовая программа, ограниченная только ресурсами памяти. По этой причине нет ограничений на число компонент, выводов компонент, соединений и листов.

EESchema позволяет создавать простые или многостраничные схемы. В случае многостраничных схем, их представление иерархическое, с быстрым доступом к каждому листу.

Для многостраничных схем в редакторе EEschema может применяться:

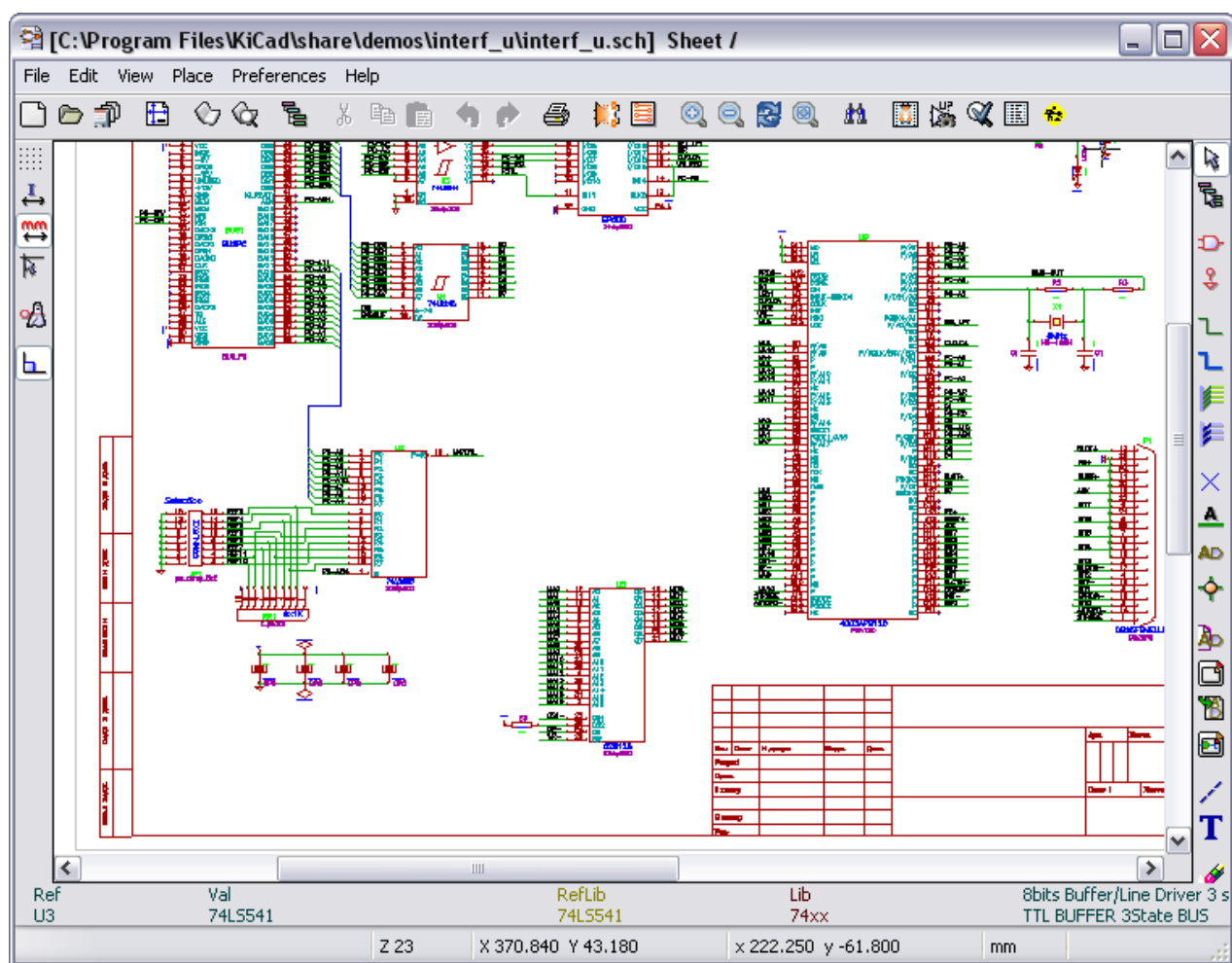
- простая иерархия (каждая схема используется единожды);

- сложная иерархия (некоторые схемы используются более одного раза);
- плоская иерархия (некоторые схемы неявно включены в основную схему).

Максимальный размер чертежа схемы возможен от формата А4 до А0, и от А до Е формата.

## 2. Основные команды EESchema

### 2.1 Доступ к командам редактора схем



Вы можете получить доступ к различным командам EESchema:

- Щелкнув по функциям основного меню (верхняя часть экрана).
- Щелкнув по иконке в верхней части экрана (основные команды).
- Щелкнув по иконке правой части экрана (специальные команды или инструменты - **tools**).
- Щелкнув по иконке левой части экрана (опции настройки отображения).
- Щелкнув по клавишам мышки (важные дополнительные команды). Как правило, правой клавишей открывается контекстное меню, зависящее от элемента под курсором (видимый размер, сетка и редактирование элементов).

- Функциональными клавишами клавиатуры (**F1, F2, F3, F4, Insert** и **space**).  
**В частности:** Клавиша **Escape** позволяет остановить выполнение команды. Клавиша **Insert** позволяет дублировать последний созданный элемент.

## 2.2 Команды, выполняемые мышкой

### 2.2.1 Основные команды

**Левая клавиша :**

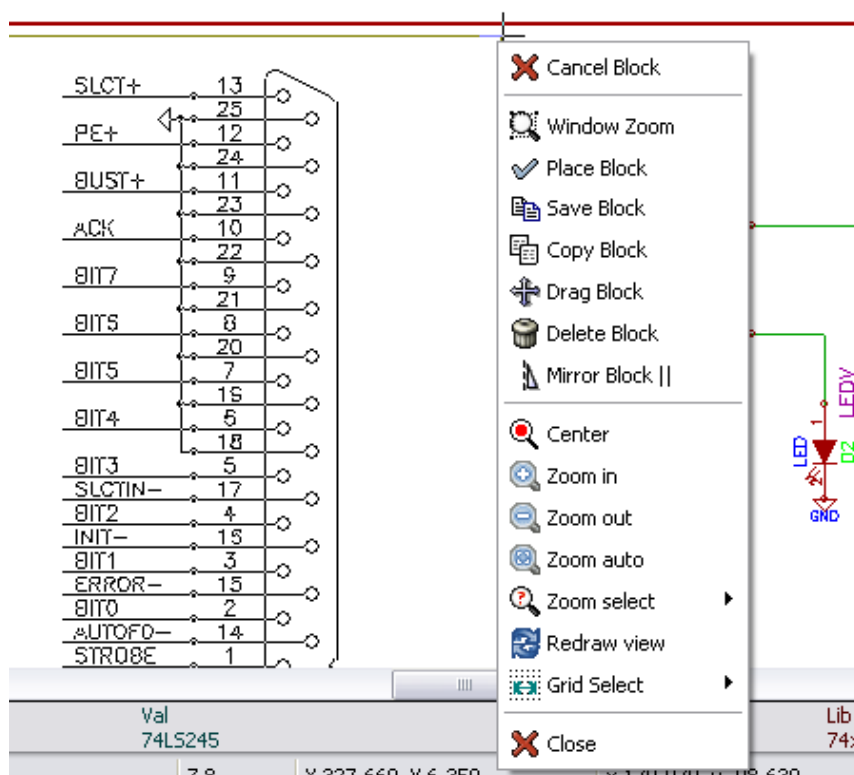
- Одинарный щелчок: отображает характеристики компонента или текст под курсором.
- Двойной щелчок: позволяет редактировать этот компонент (если элемент редактируемый) или текст.

**Правая клавиша :**

- Открывает выпадающее меню.

### 2.2.2 Операции с блоками

Вы можете перемещать, перетаскивать, копировать и удалять выбранные блоки (прямоугольные фрагменты схемы) во всех меню EESchema.



Выбираются они левой клавишей мышки. Команда выполняется при отпускании клавиши. Удерживая одну из клавиш **Shift**, **Ctrl**, или две клавиши **Shift** и **Ctrl**, при выборе в результате вы копируете, перетаскиваете или удаляете выбранный блок. Команда выполняется при отпускании клавиши.

Таблица команд:

Левая клавиша мышки	Переместить выбранное.
Shift + левая клавиша мышки	Копировать выбранное.
Ctrl + левая клавиша мышки	Перетащить выбранное.
Control + Shift + левая клавиша мышки	Удалить выбранное.

Во время перемещения выбранного :

- Щелкните вновь для возврата элементов на место.
- Щелкните правой клавишей для отмены перемещения.

Если запущена команда переместить блок, другая команда **Block** может быть выбрана из выпадающего меню (мышка, правая клавиша):

## 2.3 «Горячие» клавиши редактора схем

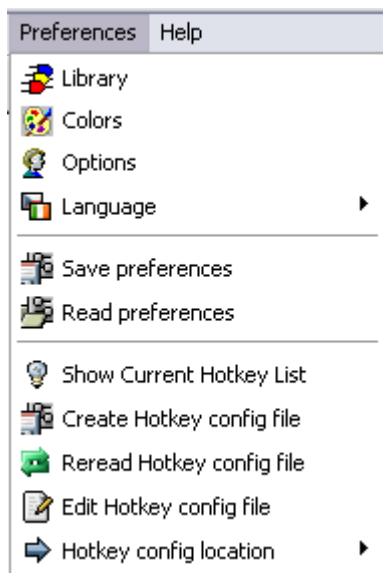
«Горячие» клавиши не являются зависимыми от ситуации.



Текущий список горячих клавиш:

? – Help: это сообщение  
 F1 – увеличение изображения  
 F2 – уменьшение изображения  
 F3 – перерисовка экрана  
 F4 – центрирование изображения  
 Space – сброс локальных координат  
 Ctrl Z – откат  
 Ctrl Y – отмена отката  
 F5 – следующий поиск  
 Delete – удаление элемента  
 Insert – повторение последнего элемента  
 Tab – перевод блока перемещения в drag-блок  
 M – переместить компонент  
 M – переместить компонент  
 A – добавить компонент  
 R – вращать компонент или наименование  
 X – зеркально отобразить компонент относит. X  
 Y – зеркально отобразить компонент относит. Y  
 N – нормальная(нулевая) ориентация компонента  
 V – редактировать значение компонента  
 F – редактировать посадочное место компонента  
 W – начать проводник

Горячие клавиши могут быть запрограммированы пользователем. Для этого необходимо:



1. Создать или обновить hotkey-файл (**Create Hotkey config file**)
2. Отредактировать его, как нужно
3. Для применения новых клавиш перечитать hotkey-файл (**Reread Hotkey...**) или перезапустить редактор схем EEschema.



## 2.4 Выбор размера сетки

Курсор в рабочем поле редактора схем перемещается по сетке, которая может быть отображена или нет (кнопка сетки всегда отображается в меню управления слева). Вы можете изменить размер сетки через выпадающее меню или меню **Preferences/Options**.

По умолчанию размер сетки схемы 50 mil (0.050") или 1,27 миллиметра. Кто-то пожелает работать со средней (20 mil) или мелкой сеткой (10 mil). Это не рекомендуется для обычной работы. Так как средняя или мелкая сетка специально предназначены для разработки или сопровождения компонент с большим количеством выводов (несколько сотен).

## 2.5 Выбор видового масштаба

Для изменения видового масштаба (ZOOM):

1. Щелкните правой клавишей мышки, чтобы открылось выпадающее меню и выберите подходящий масштаб.
2. Используйте функциональные клавиши:
  - **F1**: Увеличить (Zoom in).
  - **F2**: Уменьшить (Zoom out).
  - **F3**: Перерисовать (Redraw).
  - **F4**: Центрировать возле курсора, или щелкните средней клавишей мышки (без ее перемещения).
  - **Window Zoom**: увеличение окном, вырезаемом средней клавишей мышки.
  - **Mouse wheel**: Zoom in/Zoom out - изменение масштаба колесиком.

## 2.6 Отображение координат курсора

Отображение координат рабочего пространства для разработки схемы производится в дюймах (inch или ") или в миллиметрах. Однако, Eeschema внутренне всегда работает с 1/1000 дюйма. Следующая информация отображается внизу в правой части окна (на строке состояния):

- Zoom-фактор масштаба.
- Абсолютная позиция курсора.
- Относительная позиция курсора.
- Текущая система единиц (mm/inch).

Относительные координаты (x, y) могут быть сброшены пробелом. Появляющиеся координаты будут относиться к этой точке.



## 2.7 Меню функций EEschema

Это меню базовых функций графического редактора EEschema, которое позволяет открыть и сохранить схему (опция File), настроить конфигурацию программы, вызвать функции редактирования (Edit),






















отображения (View) и размещения (Place) элементов схемы, вызвать обращение к файлу оперативной помощи (Help).

File Edit View Place Preferences Help

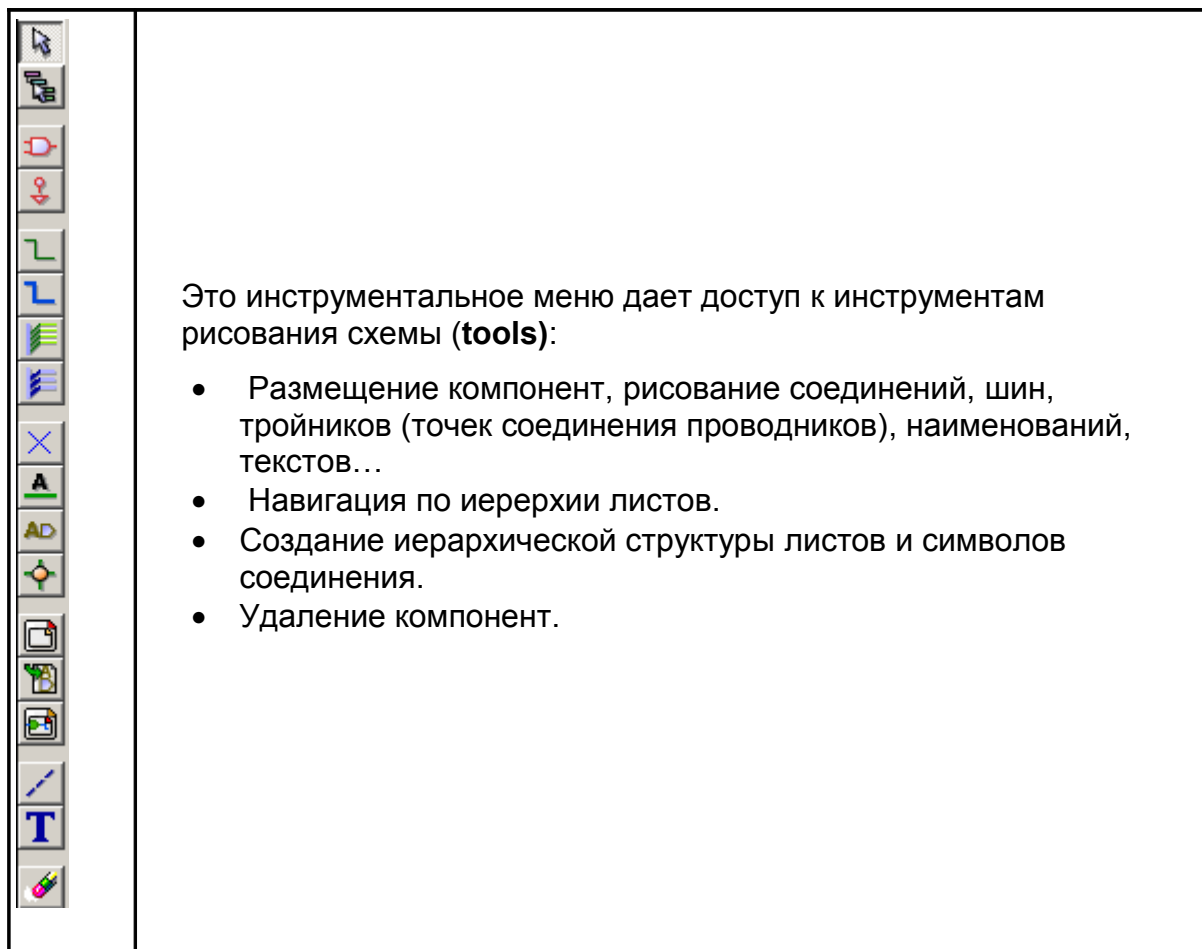
## 2.8 Верхнее инструментальное меню

Это головное меню редактора обеспечивает быстрый доступ к основным функциям EESchema.



	Создать новую схему.
	Открыть схему.
	Сохранить полную схему (со всей иерархией).
	Выбор размера листа и редактирование штампа чертежа.
	Вызов редактора компонент <b>Libedit</b> для проверки и редактирования библиотеки компонент.
	Показать компоненты библиотеки ( <b>Viewlib</b> ).
	Вызов «навигатора», чтобы отобразить древовидную структуру иерархии схемы (если она содержит добавочные листы) и быстро перейти к любому листу иерархии.
	Удалить выбранный элемент во время перемещения блока схемы ( <b>move block</b> ).
	Копировать выбранный элемент в буфер обмена во время перемещения блока ( <b>move block</b> ).
	Копировать последний выбранный элемент или блок с текущего листа.
	Отменить последнее удаление (до 10 уровней).
	Открыть меню печати схемы
	Вызвать программу <b>CVPCB</b> перехода к печатной плате
	Вызвать редактор печатных плат <b>PCBNEW</b> .
	Zoom in и out - увеличение и уменьшение вокруг центра экрана.
	Перерисовать экран с оптимальным Zoom (масштабом обзора).
	Вызвать меню поиска компонент и текстов.
	Создать netlist (список цепей в формате Pcbnew или Spice
	Формирование позиционных обозначений компонент схемы
	ERC (Electrical Rules Check): автоматическая проверка электрических соединений схемы
	Генерация перечня элементов схемы в формате BOM (Bill of materials) и/или иерархических меток.

## 2.9 Иконки правого инструментального меню



Детальное использование этих инструментов описано в разделе **Создание и редактирование схем - Diagram Creation/Editing**. Перечень использования дан ниже.



Приостановить порядок или действующий инструмент (текущую функцию редактора схем EESchema).



Навигация по иерархии схемы



Вызвать меню размещения компонент в поле схемы.



Меню размещения компонент питания (Powers).



Рисование проводников для соединений схемы.



Рисование шин.



Подсоединение к шине. Эти элементы играют только декоративную роль, не осуществляя соединения; таким образом, они не могут использоваться для соединения проводников.



Соединения шины с шиной, могут соединить только две шины между собой.



Символ “No connection – нет соединения” для размещения на выводах компонент, которые не должны быть присоединены. Символ полезен при ERC-проверке, чтобы показать умышленно ли выводы оставлены свободными или были пропущены.



Размещение локальной метки. Два проводника могут иметь идентичные метки на одном листе. Для соединения между разными листами вы должны использовать глобальные символы.



Размещение глобальных меток, которое дает возможность осуществить связь между суб-листами и основным листом (имеет символ структуры).



Размещение пересечения (тройника). Для соединения пересекающихся проводников или проводника и вывода, когда оно не однозначно. (то есть, если конец проводника или вывод не соединен с одним из концов другого проводника).



Расположение символа иерархии листов (масштабируемый прямоугольник). Вы должны назначить имя файла для сохранения данных этого суб-листа (subsheet).



Импорт глобальной метки с суб-листа с целью создать соединение с символом суб-листа. Глобальные метки предполагаются уже размещенными на этом суб-листе. Для этого иерархического символа созданные точки соединения эквивалентны традиционным выводам компонент и должны быть соединены.



Создание глобальной метки на суб-листе для создания точек соединения. Эта функция проще предыдущей, поскольку не требует уже определенных глобальных символов.



Нанесение линий для формирования чертежа схемы с форматной рамкой (обрамления схемы). Соединений не выполняют.



Размещение текста вспомогательных надписей..



Удалит выбранный элемент. Если несколько наложенных элементов выбраны, приоритет отдается наименьшему (в убывающей последовательности: пересечение, признак НетСоединения (NoConnect), соединение, шина, текст, компонент). Это также касается иерархии листов. Заметьте: функция “Undelete” (отменить) - основной инструментальной панели, который позволяет отменить последние удаления.

## 2.9 Иконки левой инструментальной панели



Эта панель управляет опциями отображения :

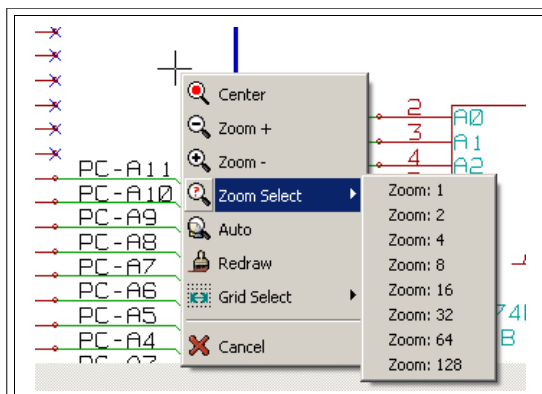
- Отображать или нет сетку
- Задать систему единиц измерения координат
- Выбрать форму графического курсора.
- Скрыть выводы.

## 2.10 Выпадающие меню для быстрого редактирования

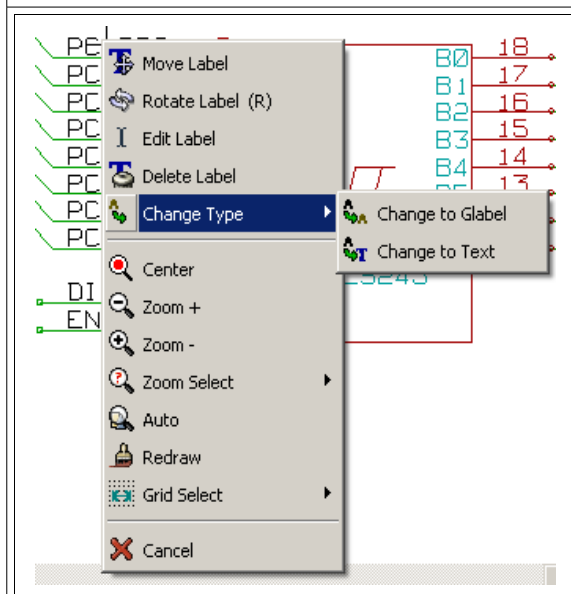
Щелчок правой клавишей мышки открывает выпадающее меню, чье содержание зависит от выбранного элемента (если есть хотя бы один).

Вы имеете немедленный доступ к:

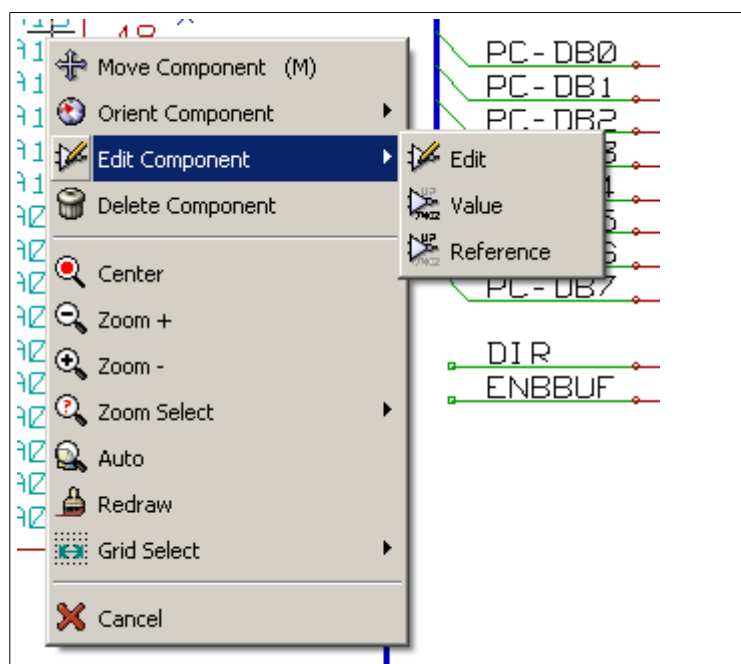
- Zoom-фактору масштаба схемы.
- Регулированию сетки рисунка.
- Заданию параметров элементов схемы (зависит от контекста)



Выпадающее меню  
без выбранного элемента.



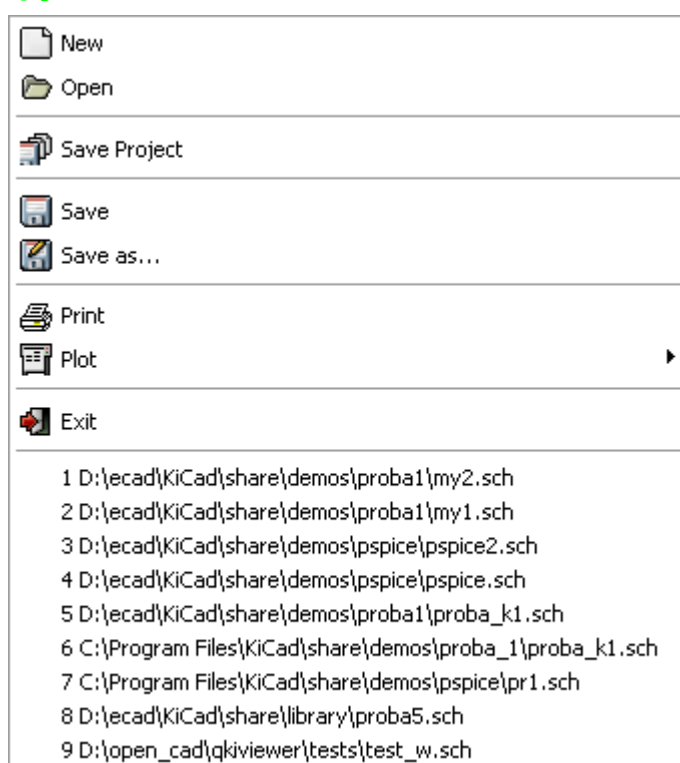
Редактирование метки



Редактирование компонента.

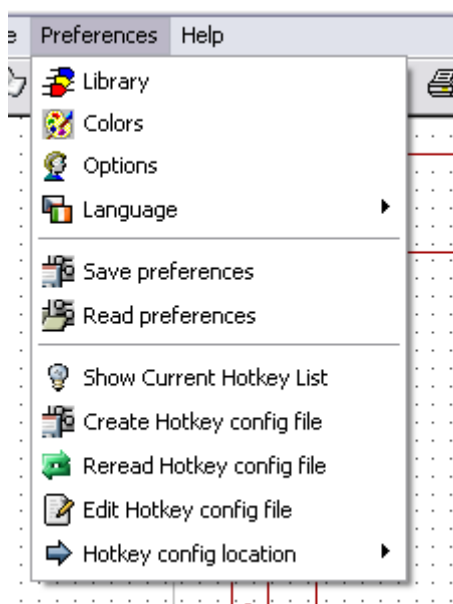
### 3. Основное меню функций редактора схем EESchema

#### 3.1 Меню функций File



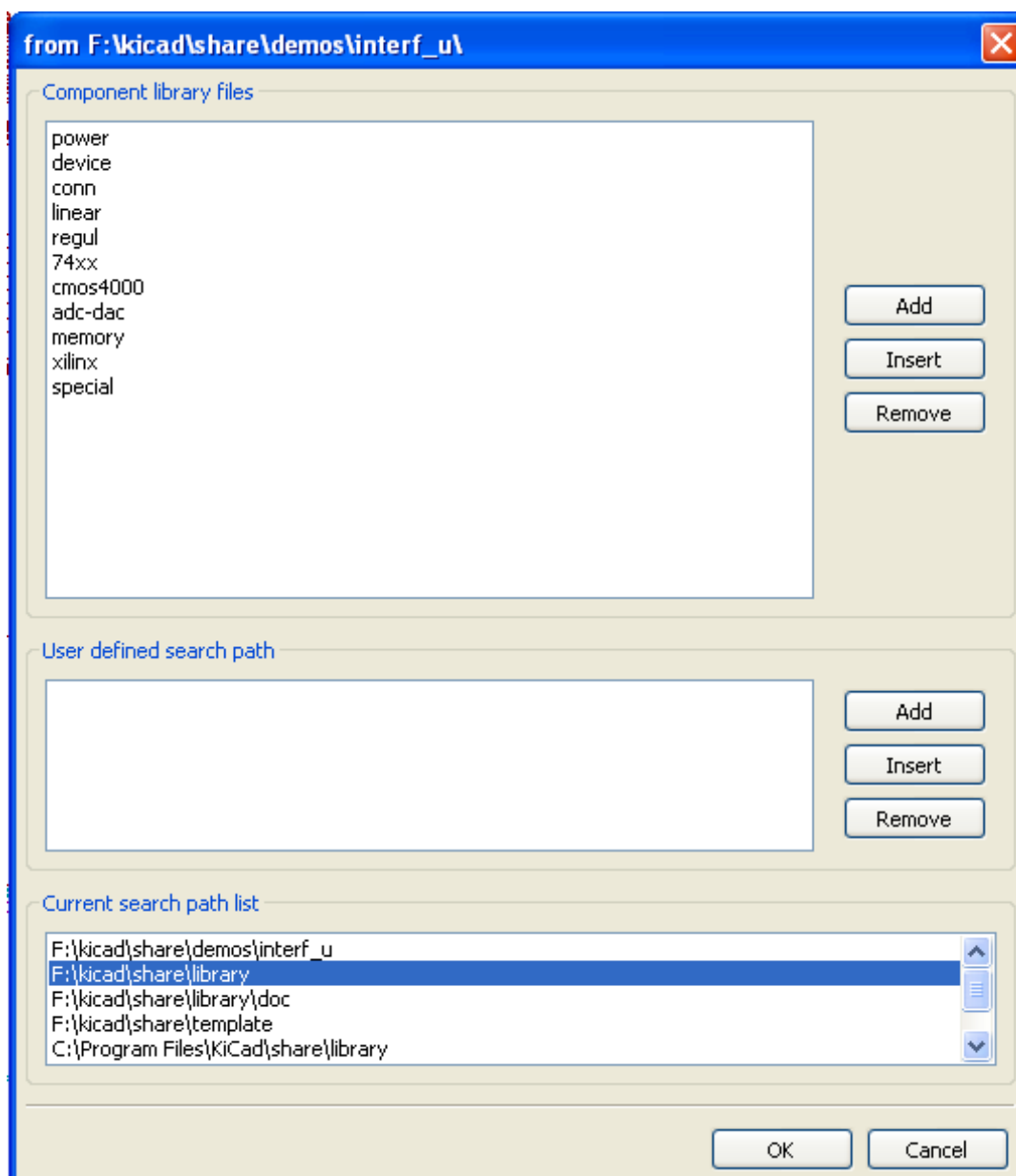
<b>New</b>	Начинает новую схему
<b>Open</b>	Загружает существующий файл схемы
<b>Save Project</b>	Сохраняет текущий проект схемы
<b>Save</b>	Сохраняет схему в текущем файле
<b>Save as...</b>	Сохраняет схему под новым именем.
<b>Print</b>	Доступ к меню печати (См. также "Print and Plot").
<b>Plot</b>	Черчение в Postscript, HPGL, DXF или SVF формате (См. также "Print and Plot").
<b>Exit</b>	Выход без сохранения.

### 3.2 Меню настроек редактора схем



<b>Library</b>	Выбор библиотек и путей к библиотекам.
<b>Colors</b>	Выбор цветов для оформления схемы
<b>Options</b>	Опции отображения (единицы схемы, сетка...)
<b>Language</b>	Выбор языка интерфейса редактора схем
<b>Save preferences</b>	Сохранить файл настроек
<b>Read preferences</b>	Прочитать файл настроек

### 3.2.1 Меню настроек - библиотеки (Preferences/Library)



Конфигурация Eeschema включает по существу:

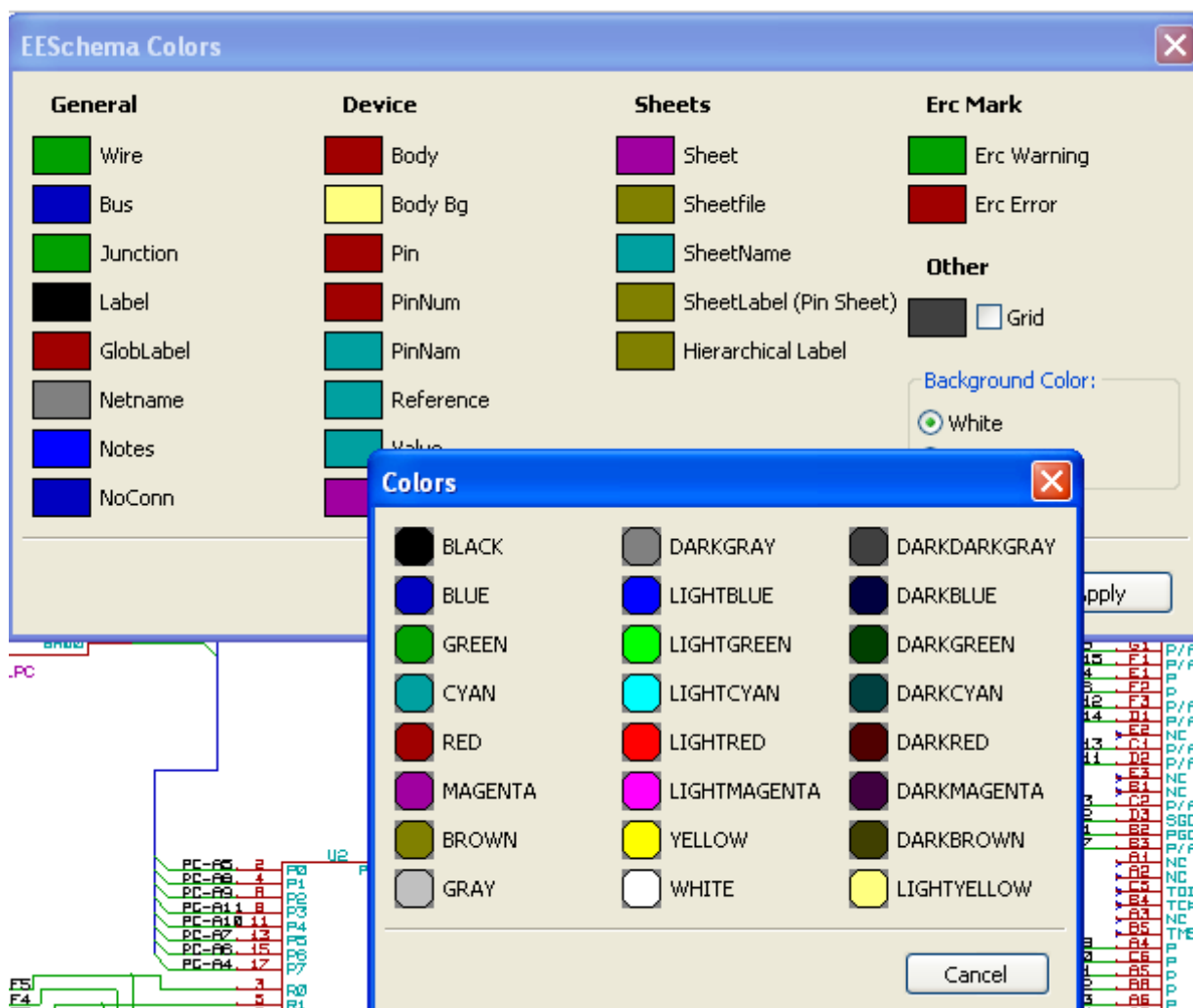
- Пути к библиотекам.
- Список библиотек.
- Netlist формат.

Параметры конфигурации сохраняются в файле типа **.pro**. В разных каталогах (папках) допустимы различные файлы конфигурации. EESchema отыскивает и использует их по убывающим приоритетам:

- Файл конфигурации **project>.pro** в текущей папке.
- Файл конфигурации **kicad.pro** в каталоге **kicad**. Этот файл может быть также предопределенной (по умолчанию) конфигурацией.
- Предопределенные значения, если файл не найден. Как минимум затем понадобится заполнить список библиотек для загрузки и сохранить конфигурацию.



### 3.2.2 Меню настроек - цвета

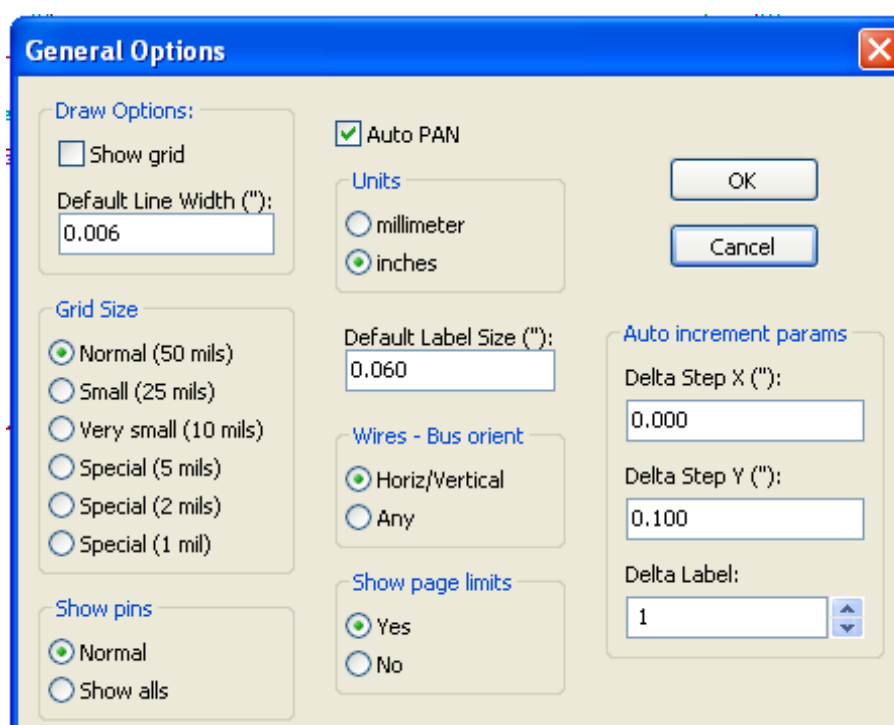


Выбор цвета для рисования различных элементов схем, выбор цвета фона основного окна редактора Eeschema (только черный или белый).

### 3.2.3 Меню настроек – опции

- Включение отображения сетки (**Show Grid**): если выбрано - сетка отображается.
- Размер сетки (**Grid Size**): работайте с нормальной сеткой (50mils = 0,050 дюйма или 1,27 мм). Меньшие сетки используются для построения компонент.
- Отображение выводов компонентов (**Show Pins**): отображает невидимые выводы, если опция включена (позволяет видеть выводы питания).
- Размеры (**Units**): выбор единиц для отображения размеров и координат курсора (дюймы или миллиметры).
- Ориентация выполняемых проводников и шин (**Wire/Bus orientation**): **Horiz/vertical** – допустимо проведение только горизонтальных или вертикальных проводников или шин; Если выбрано **Active Any**, то допустима любая ориентация соединений на схеме.

- Авто панорамирование (**Auto Pan**): если установлено, то автоматически увеличивает окно, если курсор выходит за его пределы во время выполнения соединения или перемещения компонента.
- Инкремент авто-размножения (**Auto increment params**): X шаг (**X Step**) - шаг по оси X для размножения (дублирования) элемента схемы (обычное значение 0); Y шаг (**Y Step**) – шаг по оси Y для дублирования элемента схемы (обычно значение 0,100 дюйма или 2,54 мм); Шаг метки (**Label Step**) - инкремент для позиции конца текста в числах (обозначении шин, например) при дублировании или редактировании (обычное значение 1 или -1).



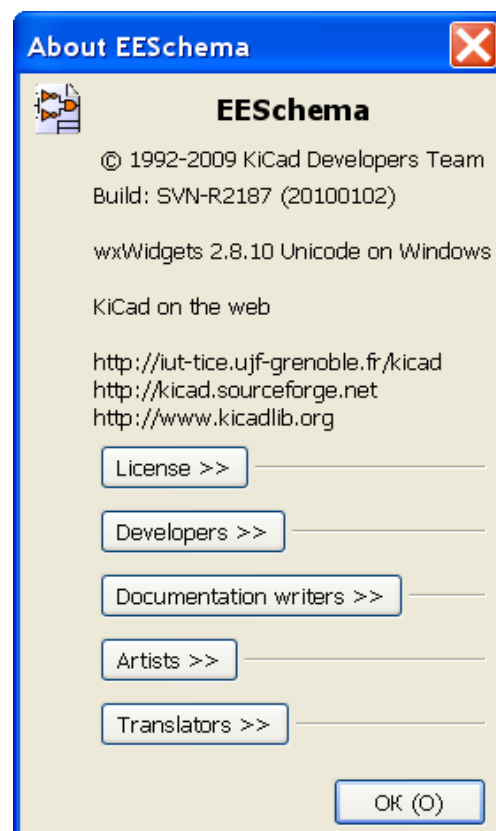
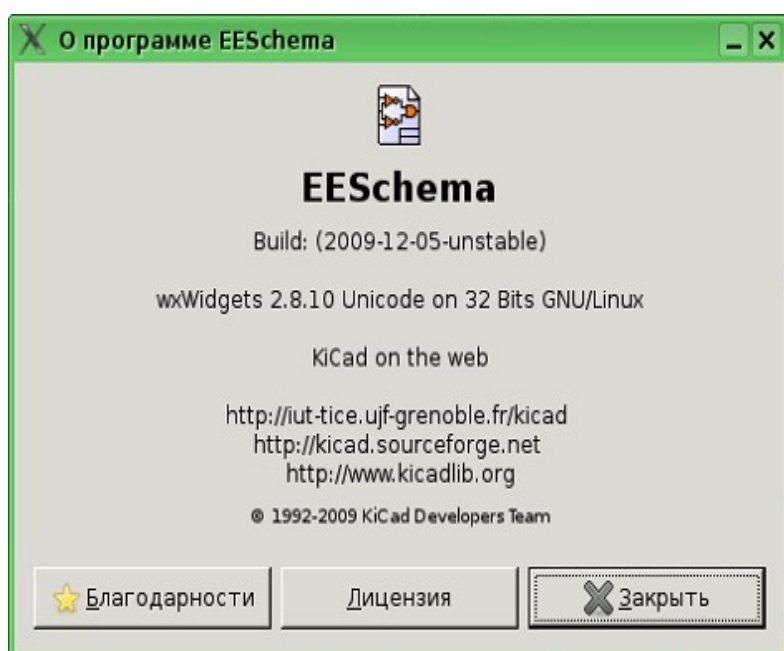
### 3.2.4 Меню настроек – выбор языка для интерфейса EESchema

Изначально используется предустановленный режим языка интерфейса графического редактора EESchema. Возможно переключение на другие языки в интерфейсе программы. Для запуска нового интерфейса EESchema нужно перезапустить.

## 3.3 Меню помощи Help

Функция Help служит для доступа к online-помощи пользователю (к тексту этого документа) и также для вывода окна с информацией о текущей версии редактора Eeschema (**About Eeschema...**). Ниже приведены изображения таких окон программы.

Установка EESchema выполнена на базе кросс-платформенной интерфейсной библиотеки wxWidgets 2.8.10 Unicode для операционных систем ALT Linux Desktop и Windows XP.



## 4. Основное инструментальное меню EESchema

### 4.1 Управление листами схемы



Доступ к установкам оформления страницы схемы:

Ввод размеры страницы и редактирование штампа (основной надписи) форматной рамки схемы (для оригинальной и русскоязычной версии EESchema рамка и штамп отличаются). Дата создания форматки, общее количество листов и нумерация листов схемы автоматически обновляются.


### 4.2 Инструменты поиска в схемах



Доступ к этому инструменту.

Вы можете искать компонент, значение или текстовую строку на текущем листе или во всей иерархии схемы. В результате поиска курсор будет позиционирован на найденном элементе, на соответствующем листе схемы.

### 4.3 Формирование списка цепей проекта (netlist)

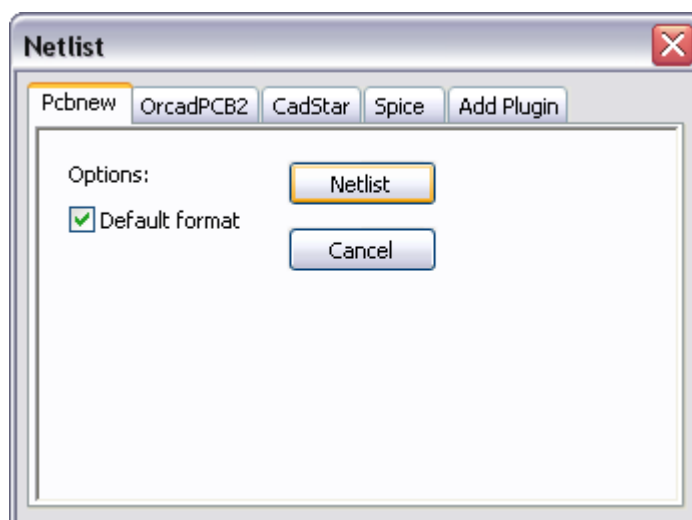
Инструментальная кнопка  обеспечивает доступ к инструменту получения из схемы списка электрических цепей - netlist. Файл netlist может быть получен со всей иерархии листов схемы (обычная опция), или только с текущего листа (в этом случае netlist частичный, но эта опция может быть полезна для некоторых программ).

В многостраничной иерархии схемы любые локальные метки опознаются только внутри листа, которому они принадлежат. Так метка ТОТО листа 3 иная, чем метка ТОТО листа 5 (если не было намерено сделано связи для их соединения). Это происходит благодаря тому, что номер листа (обновленный командой **annotate** - аннотировать) ассоциирован с локальной меткой. В предыдущем примере первая метка ТОТО в действительности ТОТО\_3, а вторая метка ТОТО оказывается ТОТО\_5.

Эта ассоциация может быть подавлена, если нужно, но необходимо осознавать возможность нежелательных соединений.

**Примечание 1:** Метки не имеют ограничений по длине в EESchema, но программа, использующая сгенерированный netlists, может иметь ограничения на этот счет.

**Примечание 2:** Избегайте пробелов в метках, поскольку они впоследствии появляются, как отдельные слова. В Eeschema нет ограничений, но многие форматы списка цепей (netlist'a) предполагают отсутствие пробелов в метках.




**Опция:** Формат по умолчанию (Default Format): устанавливается для выбора программой **Pcbnew** в качестве формата вывода списка цепей по умолчанию. Список может также быть сформирован и в других форматах :

- Orcad PCB2 (для редактора печатных плат САПР OrCAD);

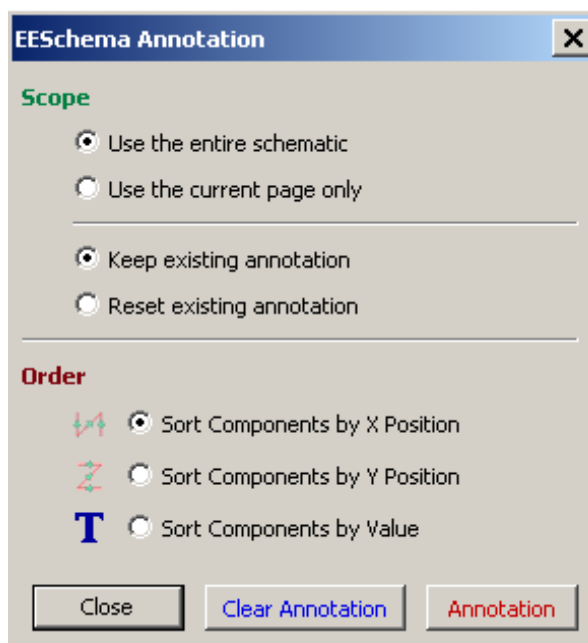
- CadStar (для редактора печатных плат САПР CADSTAR);
- PadsPCB (для редактора печатных плат САПР PADS);
- Spice (для Spice-симуляторов схем).

#### 4.4 Инструмент формирования позиционных обозначений схемы

 - доступ к функции аннотирования (назначения) позиционных обозначений компонент схемы. Этот инструмент выполняет автоматическую нумерацию обозначений введенных в схему компонент.

В обозначения многоэлементных компонент (таких, как 7400 TTL, который состоит из 4 логических элементов - вентилях) при этом добавляется символьный суффикс. Так, 7400 TTL, обозначенное как U3, будет поделено на U3A, U3B, U3C и U3D.

Вы можете аннотировать все компоненты или только новые компоненты, то есть только те, что прежде не были обозначены.



В приведенном окне необходимо выбрать предмет аннотирования (всю схему или текущую ее страницу) и назначить порядок сортировки позиционных обозначений согласно их координатам на схеме: X или Y, или согласно значениям Value, присвоенным компонентам.

Обычно аннотируется вся схема. Аннотирование текущего листа используется только в специальных случаях – например, для оценки количества резисторов на текущем листе.


#### **Клавиши:**

**Annotation:** Выполняет аннотирование схемы.

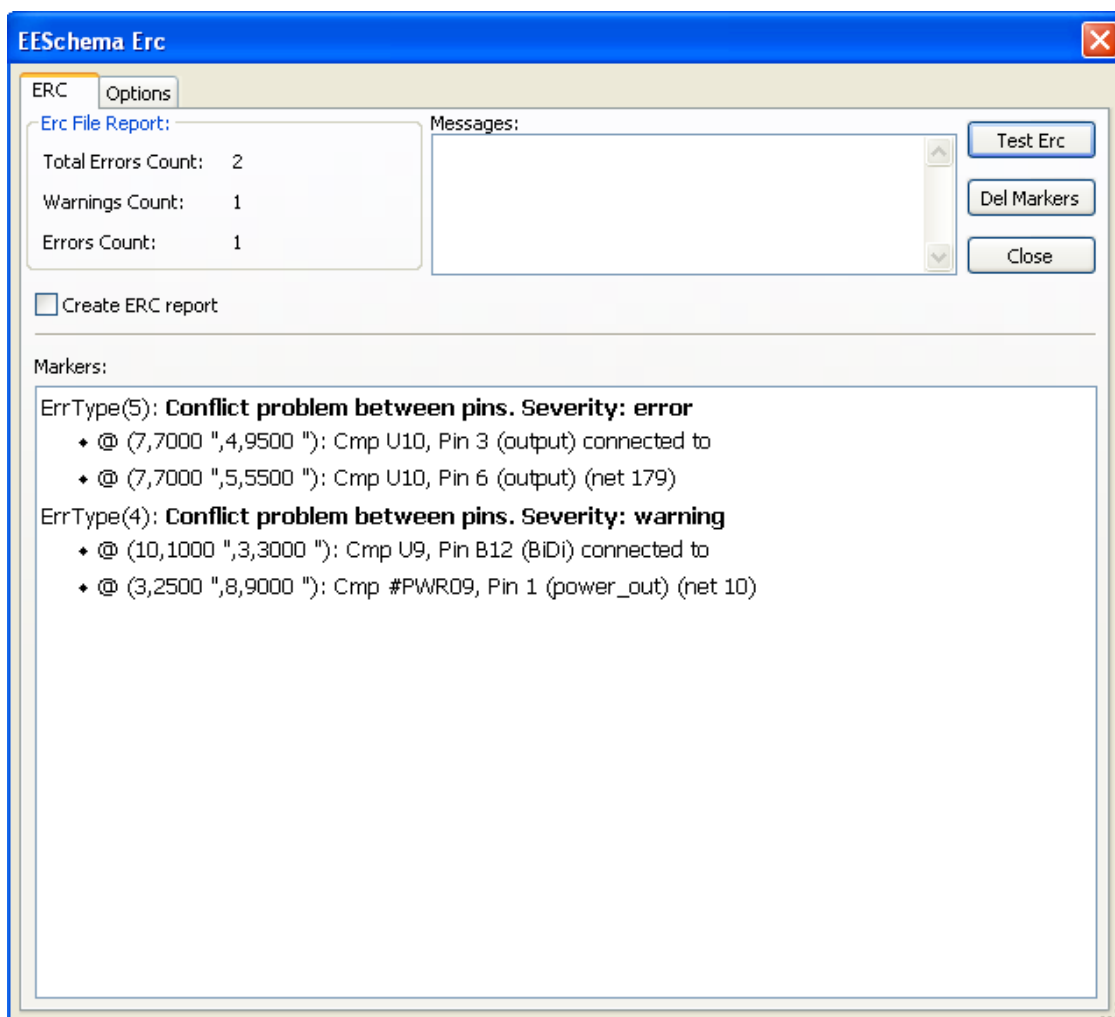
**Clear Annotation:** Удаляет аннотирование схемы.

**Close:** закрывает диалоговое окно.

## 4.5 Применение инструмента ERC

 Активирование инструмента ERC (**E**lectrical **R**ules **C**heck – проверка электрических правил) служит для контроля правильности электрических соединений схемы. Эта функция чрезвычайно полезна как для выявления соединений, которые забыли сделать, так и для определения несовместимостей схемы.

При контроле Eeschema размещает специальные маркеры на проблемных выводах или метках. Щелчком левой клавиши мышки по маркеру для него может быть выдана поясняющая диагностика. Также может быть сгенерирован отчет о сеансе ERC-теста. Диалоговое окно теста ERC

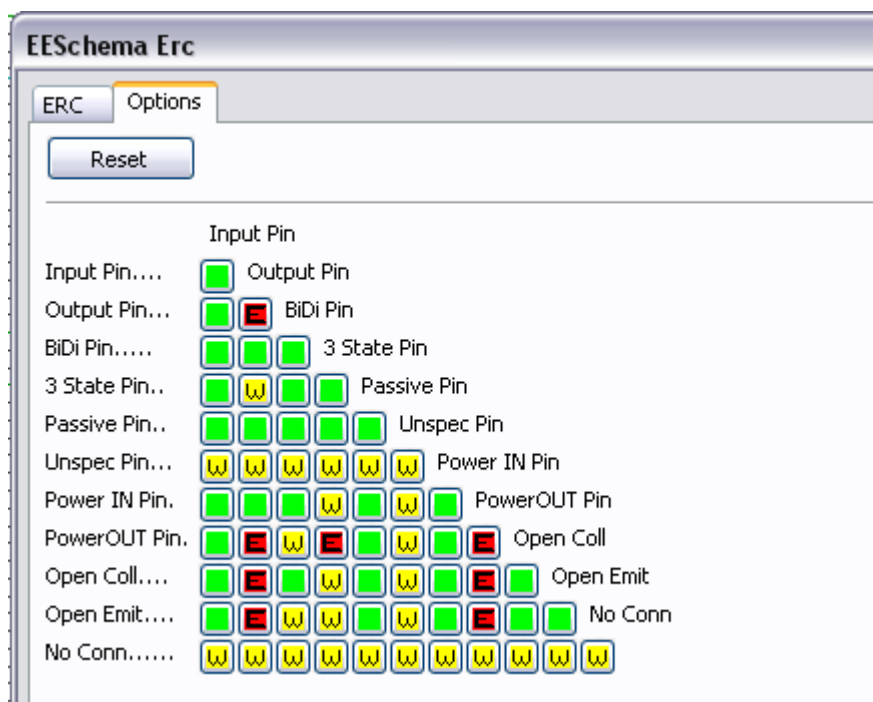


Данные, отображаемые на экране диалога **EEschema Erc**:

- общее количество ошибок и предупреждений для схемы;
- количество ошибок (Errors);
- количество предупреждений (Warnings).

Для записи отчета о проверке в файл используйте клавишу **Create Erc report**. Клавиша проверки **Test Erc** выполняет Erc-тест. Удалить маркеры проверки со схемы можно с помощью клавиши **Del Markers**. Выход из окна диалога выполняется с помощью клавиши **Close**.

### 4.5.1 Задание опций теста ERC




Это окно диалога предварительных установок ERC-теста схемы, которое позволяет вам создавать правила соединений между выводами. Можно сделать выбор между тремя опциями для каждого случая :

- Не обращать внимания на ошибки (No error, зеленый квадрат)
- Предупреждать (Warning, желтый квадрат)
- Сообщать об ошибках (Error, красный квадрат)

Каждый квадратик матрицы опций может быть модифицирован щелчком по нему.

### 4.6 Вывод перечня элементов в формате BOM

 Это меню позволяет генерировать файл со списком компонент и/или иерархических соединений (глобальных меток). Компоненты могут быть отсортированы по следующим критериям:

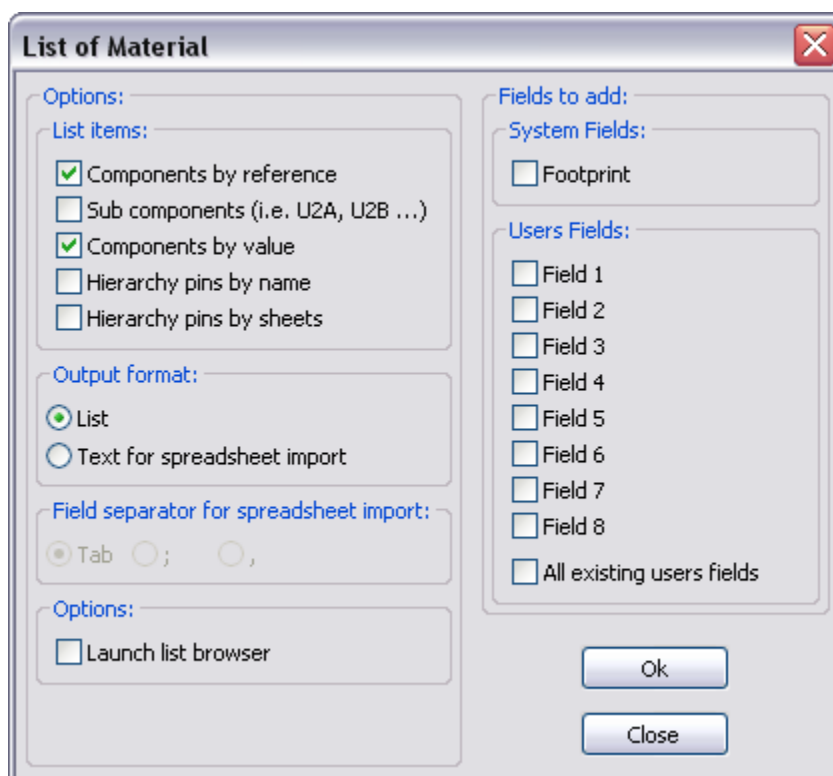
- по ссылкам (Reference).
- по значению (Value).

Многоэлементные компоненты могут при этом детализироваться. Глобальные метки могут быть отсортированы по :

- алфавиту.
- листам схемы.

Различные способы сортировки могут использоваться одновременно.






### Опции:

<b>Components by Reference</b>	Перечень сортируется по ссылкам (Reference).
<b>Component by Value</b>	Перечень сортируется по значению (Value).
<b>Sub components</b>	Перечень показывает каждый элемент многоэлементных компонент (из U2A, U2B...).
<b>Hierarchy Pins by name</b>	Иерархические соединения сортируются по алфавиту.
<b>Hierarchy Pins by Sheet</b>	Иерархические соединения сортируются по номерам листов.
<b>List</b>	Создание обычного текстового файла для печати
<b>Text for spreadsheet import</b>	Создание файла ASCII для легкого импорта в таблицу
<b>Launch list browser</b>	Пуск текстового редактора для просмотра созданного BOM-файла

## 4.7 Импорт файла обратной аннотации от PCBNEW

 Данная функция читает файл типа **.STF**, предварительно созданный программой CVPCB, и заполняет поле (F3) наименований посадочных мест компонентов. Это полезно не только для PCBNEW, но и добавляет соответствующее поле в файлы материалов BOM и списка цепей NET. Имея информацию о позиционных обозначениях/посадочных местах компонентов в одном файле схемы, можно сформировать полный список цепей (netlist), сделав файл типа .CMP необязательным.

## 5. Создание и редактирование схемы

### 5.1 Определения вариантов схем

Схема может быть разработана на одном листе, но часто для нее требуется несколько листов.

Схемы, представленные на нескольких листах, называются иерархическими, и все эти листы составляют проект EESchema (каждый лист при этом представлен своим собственным файлом). Таким образом, проект (project) состоит из основной схемы, называемой корневой (root schematic), и дополнительных суб-листов, образующих иерархию схемы.

Для того, чтобы можно было отыскать каждый из файлов проекта, вы должны выполнять определенные правила рисования схемы, которые описаны далее.

Ниже, когда мы говорим о проекте, мы будем иметь в виду как проект на одном листе, так и иерархический многостраничный проект. Специальные дополнительные символы поддерживают иерархические схемы и их характеристики.

### 5.2 Вводные наставления

Схемы, разработанные с помощью EESchema – это больше, чем просто графическое представление электронного устройства. Это, как правило, отправная точка или начальное звено в цепи конструирования, которое позволяет:

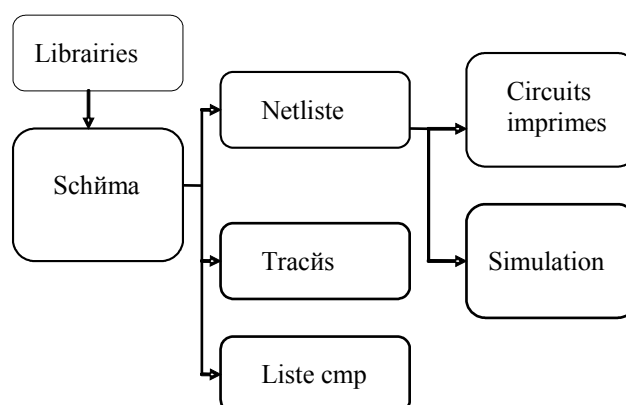
- Проверить правила электрического соединения (проверка ERC), что в свою очередь позволяет обнаружить ошибки или пропуски в схеме.
- Автоматическое формирование перечня материалов (bill of material).
- Генерацию netlist-списка соединений для моделирования схемы в такой, например, программе, как Pspice.
- Вывод netlist для программы проектирования печатной платы PCBNEW. Возможна последовательная автоматизированная сверка соединений схемы и разводки печатной платы.

Чтобы извлечь пользу из этих возможностей, вы должны придерживаться некоторых правил и соглашений. Это позволит избежать неприятных сюрпризов и ошибок.

Схема, главным образом, состоит из компонент, соединений (проводников), меток (labels), точек соединений (или пересечений), шин и точек подключения питания (power ports).

Для ясности схемы вы можете дополнительно размещать чисто графические элементы, такие как вводы шины, комментарии и штриховые линии для прорисовки ее структуры.

## 5.3 Последовательность выполнения проекта




Программа EESchema разработки схемы использует библиотеки компонент (component libraries).

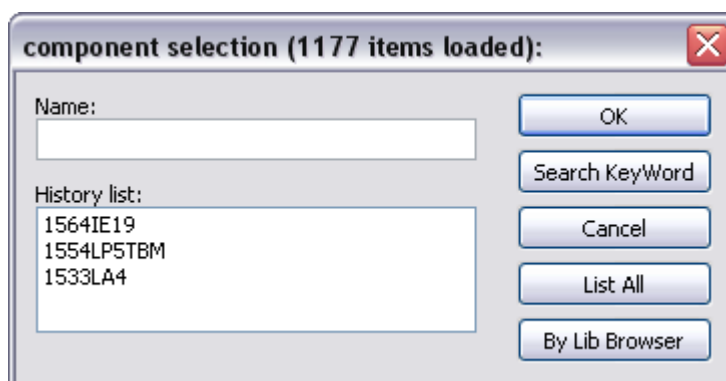
В дополнение к основному файлу разрабатываемой схемы чрезвычайно важен файл netlist-спецификации, поскольку он используется в других программах. Файл netlist передает список компонент и соединений непосредственно из схемы.

К сожалению для пользователя, существует множество форматов netlist-спецификаций, в том числе более известные, как в случае формата Spice, например.

## 5.4 Размещение и редактирование компонент схемы

### 5.4.1 Поиск и размещение компонента на схеме

Для загрузки компонент в схему используйте инструмент . Для размещения нового компонента щелкните по месту его предполагаемой установки. Появившееся диалоговое окно позволит вам впечатать имя модуля для загрузки.



В окне диалога отображаются два последних загруженных элемента. Если вы введете \*, или если выберете клавишу **List All** (весь список), EESchema отобразит список библиотек, а затем доступные их компоненты.

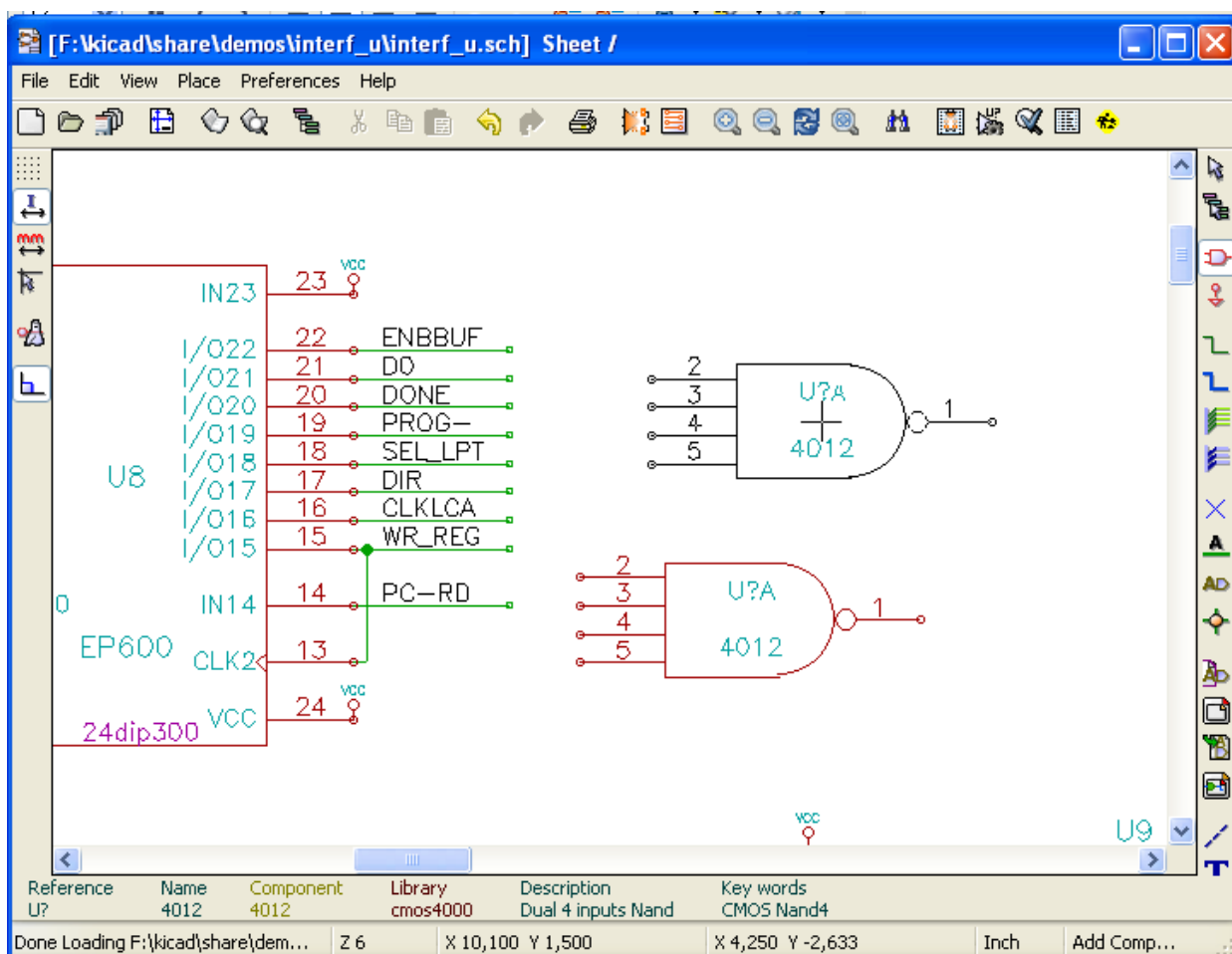
Если впечатать символ "=", сопровождаемый ключевыми словами, EESchema отобразит список компонент, соответствующих этим ключевым словам.

Вы можете также ограничить список выбора имен: например, если вы введете LM2\*, то все имена компонент, начинающиеся с LM2, будут представлены в списке.

Выбранный компонент появится на экране в режиме размещения. Перед размещением в выбранном месте (щелчком левой клавиши мышки) вы можете повернуть компонент (на 90 градусов за раз), осуществить отражение, соответственно, по оси X или Y, или выбрать его представление через выпадающее меню быстрого редактирования.


Это все можно легко сделать и после размещения. Если нужный компонент отсутствует, запомните, что вы можете загрузить схожий компонент, а затем модифицировать его: если нужен 54LS00, вы можете, естественно, загрузить 74LS00 и изменить его имя с 74LS00 на 54LS00.

Вот компонент в процессе размещения:



#### 5.4.2 Подключение питания (power ports)

Символ порта питания это компонент (символы сгруппированы в библиотеке *power*). Для размещения можете использовать предыдущую

команду. Но если это действие повторяется часто, то есть функциональная кнопка  на инструментальной панели.

Этот инструмент подобен предшественнику, за исключением того, что поиск осуществляется непосредственно в библиотеке *power library*, экономя время.

### 5.4.3 Редактирование компонент и модификация уже размещенных

Редактирование или модификацию различают двух видов.

- Модификация собственно компонента (позиция, ориентация, выбор элемента многоэлементного компонента).
- Модификация одного из полей компонента (ссылка, значение или другого).

Как только компонент размещен, вы можете изменить его значение (это актуально для резисторов, конденсаторов...), но бесполезно пытаться сделать это немедленно для присвоенного номера ссылки или для выбранного элемента многоэлементного компонента (наподобие 7400).

Присвоение позиционных обозначений элементам схемы может быть сделано автоматически через функцию аннотации (annotation function).

#### 5.4.3.1 Модификация компонента

Для выполнения этого действия позиционируйте курсор мышки на компоненте (но не на поле компонента). Затем можно:

- Дважды щелкнуть по компоненту, чтобы открыть диалоговое окно полного редактирования его свойств.
- Щелкнуть правой клавишей мышки, чтобы открыть выпадающее меню и воспользоваться одной из отображаемых в нем команд (**Move**-переместить, **Orientation**-ориентация, **Edit**-редактировать, **Delete**-удалить).

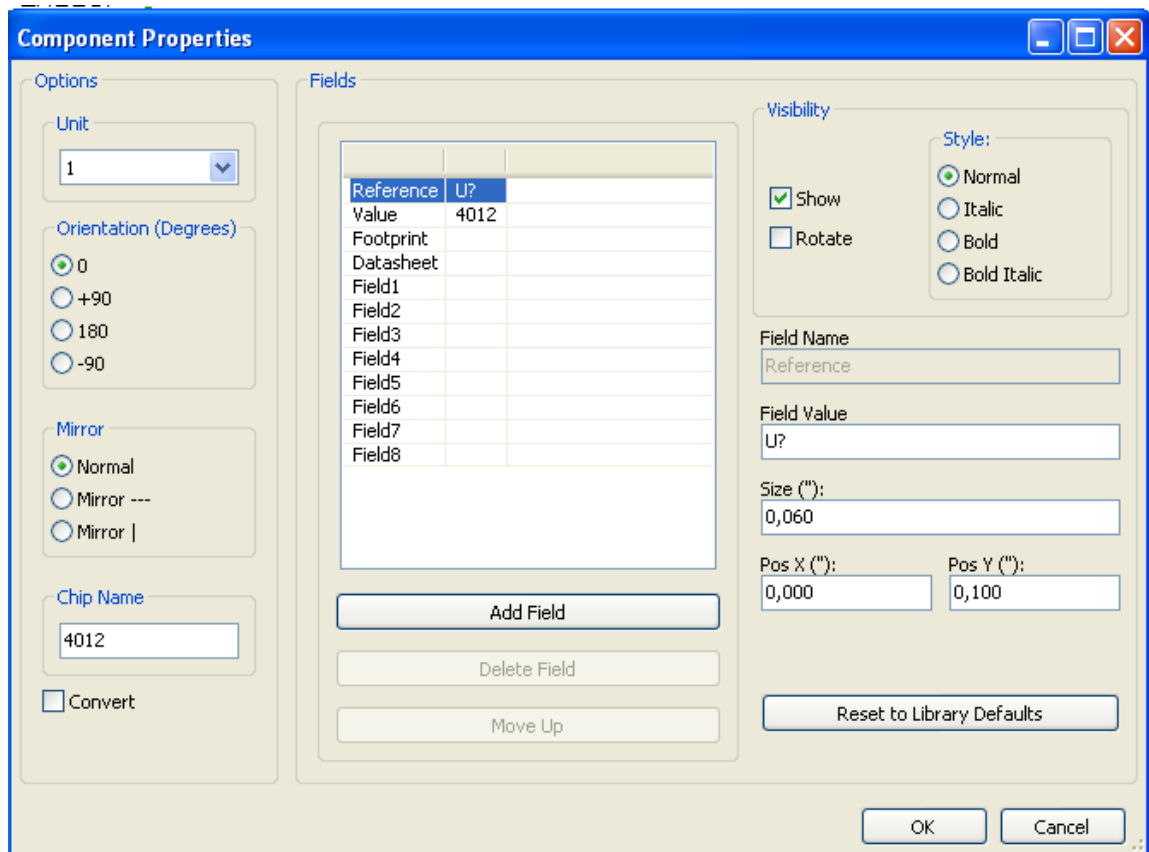
#### 5.4.3.2 Модификация текстовых полей

Вы можете модифицировать ссылку, значение, позицию, ориентацию, размер и видимость полей.

Для простого редактирования:

- Дважды щелкните по текстовому полю, чтобы модифицировать его.
- Щелкните правой клавишей и выполните одну из команд в выпадающем меню (**Move**-переместить, **Rotate**-повернуть, **Edit**-редактировать, **Delete**-удалить).

Для более полного редактирования или с целью создания новых полей дважды щелкните по компоненту. При этом открывается диалоговое окно для задания свойств компонента (Component Properties) :



Каждое поле может затем редактироваться. Его можно сделать видимым, задать зеркальное отражение по горизонтали или вертикали, отобразить в обычном, жирном или курсивном стиле.

Отображаемая (изменяемая) позиция всегда выводится для нормального отображения компонента (без поворота или отражения) и относится к точке привязки компонента на схеме.

## 5.5 Проводники, шины, метки и порты питания

### 5.5.1 Основные элементы схемы

Эти элементы схемы могут быть размещены с помощью инструментов правой вертикальной инструментальной панели:

- **Wires** - **проводники** для изображения обычных соединений схем
- **Buses** – **шины**, используемые только для соединения меток шин для улучшения восприятия схемы
- **Dotted lines** - **штриховые линии** для улучшения графического представления схем
- **Junctions** - **пересечения** для индикации точек обязательного соединения между пересекающимися проводниками или шинами
- **Bus entries** - **входы в шину** для присоединения проводников к шине или шины к шине, для эстетического восприятия чертежа схемы
- **Labels** - **метки** для обычных соединений листа схемы
- **Global labels** - **глобальные метки** для соединений между листами.
- **Texts** – **текстовые надписи** для комментирования схемы

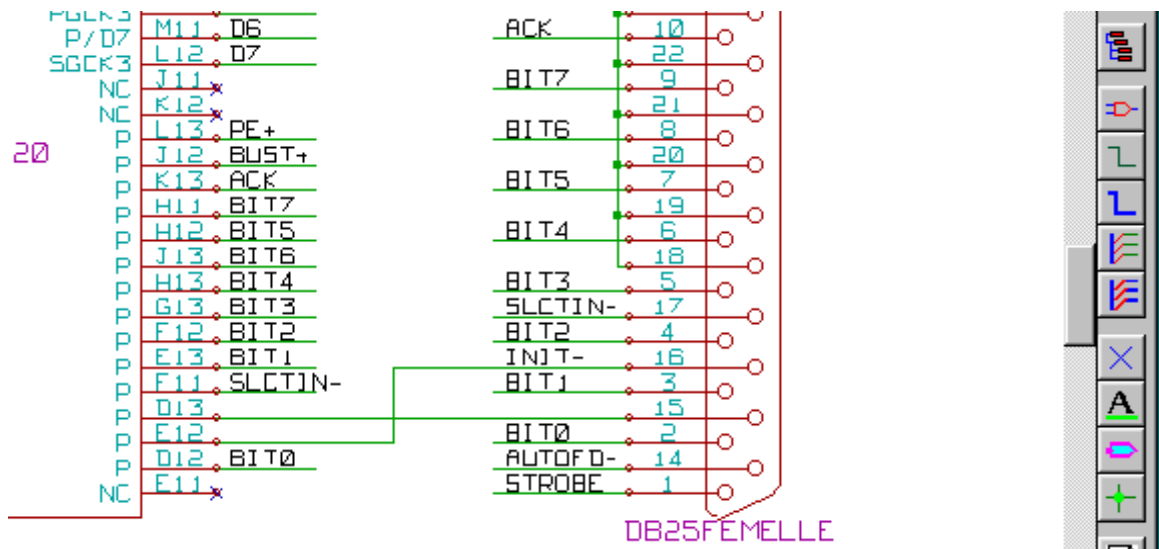
- **NoConnection** – символы для изображения отсутствующего соединения на выводе компонента
- **Hierarchy sheets** - иерархия листов и выводов схемы.

### 5.5.2 Выполнение соединений проводниками и метками

Есть два пути установить соединение на схеме:

- Проводником от вывода к выводу (Pin to pin wires).
- Через метки (Labels).

Следующий рисунок показывает оба метода :



#### Примечание 1:

Точка контакта (или закрепления) метки – это левый нижний угол первой буквы метки. Эта точка должна, следовательно, контактировать с проводником, или быть наложена на точку контакта вывода так, чтобы метка бралась в расчет.

#### Примечание 2:

Для создания соединения часть проводника должна соединяться концом с другим проводником или выводом. Если есть перекрытие (если проводник проходит поверх вывода, но без соединения с концом этого вывода), соединения не будет. Метка, однако, будет соединяться с проводником, какая бы часть точки привязки метки не оказалась на этом проводнике.

#### Примечание 3:

Если проводник должен соединяться с другим проводником, иначе, чем концами, необходимо разместить символ пересечения (junction symbol) – жирная точка “тройника” в точке пересечения.

Предыдущий рисунок (с проводниками, соединенными с DB25FEMALE выводами 22, 21, 20, 19) показывает подобный случай соединения с использованием символа пересечения.

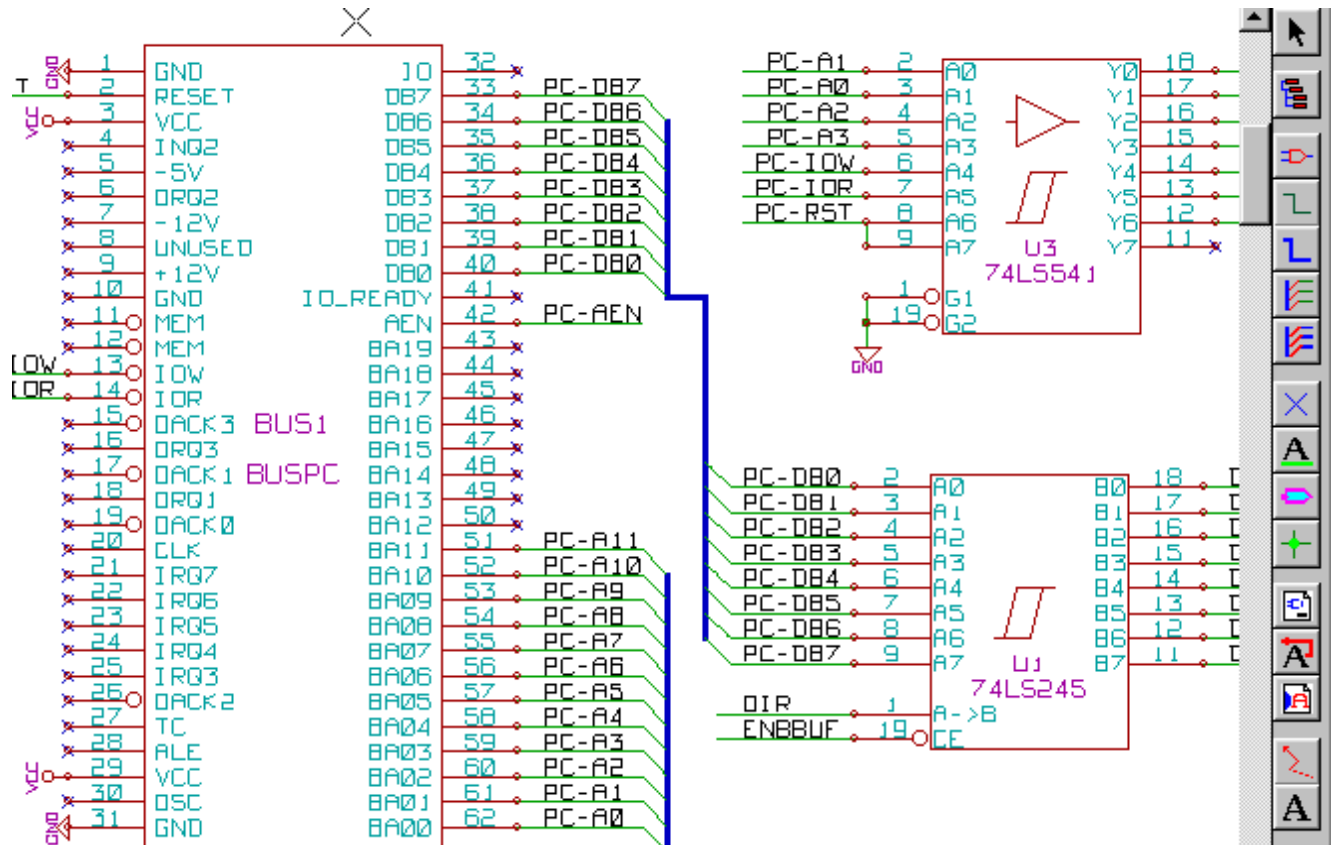
#### Примечание 4:



Если две разные метки помещены на одном проводнике, то они соединены между собой и становятся эквивалентны: все другие элементы контактирующие с одной или другой меткой становятся также соединенными между собой.

### 5.5.3 Соединения шин (Buses)

Давайте рассмотрим следующую схему:



Множество выводов схемы (особенно это выводы компонент U1 и BUS1) здесь соединены с шинами.

#### 5.5.3.1 Элементы шины (Bus members)

С точки зрения схемы шина – это подборка сигналов, начинающихся с общего префикса, а заканчивающаяся числом.

Это такая же концепция, как для микропроцессорных шин. Каждый сигнал - member-элемент шины. PCA0, PCA1, PCA2 – это элементы шины PCA.

Полная шина названа PCA[n...m], где n и m – это первый и последний номер проводника этой шины. Если PCA имеет 20 элементов от 0 до 19, то полная шина обозначается, как PCA[0..19]. Но подборка сигналов подобная PCA0, PCA1, PCA2, WRITE, READ не может содержаться в шине.

#### 5.5.3.2 Соединения между элементами шины

Выводы, соединяющие элементы шины, должны быть соединены с помощью меток (labels). Действительно, прямое соединение вывода и шины



не ощутимо, поскольку шина – это набор сигналов, и подобное соединение будет проигнорировано программой EESchema.

В примере выше соединение выполнено метками, размещенными на проводниках, присоединенных к выводам.

Соединения через ввод в шину (сегменты проводников под 45 градусов) к проводникам шины имеет только эстетическое значение, и не является необходимым на чисто схемном уровне.

Фактически, благодаря команде повтора - **repetition command** (клавиша **Insert**), соединение может быть очень быстро установлено, если выводы компонента выстроены по возрастанию (общепринятый случай для таких компонент, как память, микропроцессоры...):

- Разместите первую метку (например, PCA0)
- Используйте команду повтора столько раз, сколько необходимо для размещения всех элементов. EESchema будет автоматически создавать следующие метки (PCA1, PCA2...), вертикально выстроенные в позиции других выводов.
- Нарисуйте проводник под первой меткой. Затем воспользуйтесь командой повтора для размещения других проводников под метками.
- Если нужно, разместите входы шины тем же путем (поместите первый вход, затем используйте команду повтора).

**Примечание:**

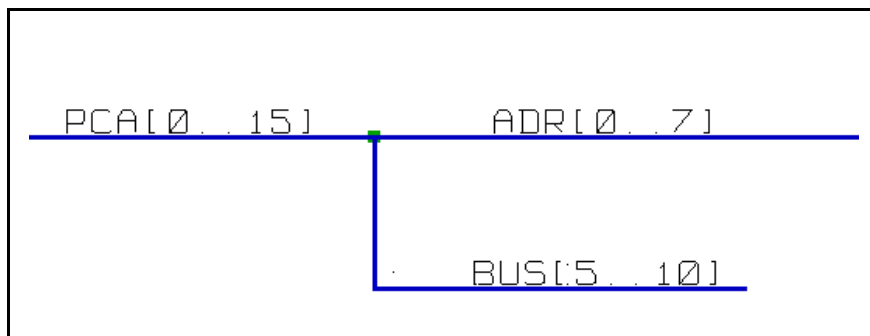
В меню **Preferences/Options** (Свойства/Опции) вы можете задать параметры повтора:

- Вертикальный шаг
- Горизонтальный шаг
- Инкремент обозначения метки (шаг увеличения, который может быть, соответственно, 2, 3 или меньше).

### 5.5.3.3 Глобальные соединения между шинами

Вам может понадобиться соединение между шинами с целью связать две шины, имеющие разные имена, или в случае иерархии - для создания соединения между разными листами.

Вы можете выполнить эти соединения следующим путем.



Шины PCA[0..15], ADR[0..7] и BUS[5..10] соединены вместе (заметьте, что пересечение здесь имеет место поскольку проводник вертикальной шины присоединяется в середине сегмента горизонтальной шины).

Точнее, соответствующие элементы соединены вместе: PCA0, ADR0 соединены, так же, как PCA1 и ADR1...PCA7 и ADR7. Более того, PCA5, BUS5 и ADR5 соединены (точно, как PCA6, BUS6 и ADR6 подобно PCA7, BUS7 и ADR7). PCA8 и BUS8 также соединены (точно так, как PCA9 и BUS9, PCA10 и BUS10).

С другой стороны, вы не можете соединить элементы разных «весовых категорий» подобным же образом. Если вы хотите соединить такие элементы от разных шин, вы должны сделать это элемент за элементом подобно двум обычным меткам, располагая их на тех же соединениях.

#### 5.5.4 Соединение питания


Когда выводы питания компонент видимы, они должны быть соединены, как и любые другие сигналы.

Трудности приходят со стороны компонент (таких как логические элементы и триггеры-защелки), для которых выводы питания обычно невидимы (invisible power pins). Трудность двойная, поскольку :

- Вы не можете присоединить проводники, из-за их невидимости.
- Вы не знаете их имен.

И более того, плохая идея сделать их видимыми и соединить подобно другим выводам, поскольку схема станет совсем не читабельной и не будет соответствовать принятым соглашениям.

#### **Примечание:**

EESchema соединяет автоматически невидимые выводы питания: если вы хотите принудительно отобразить эти невидимые выводы питания, вы должны установить опцию **Show invisible power pins** в диалоговом окне **Option** основного меню, или через иконку  левой инструментальной панели (панель опций).

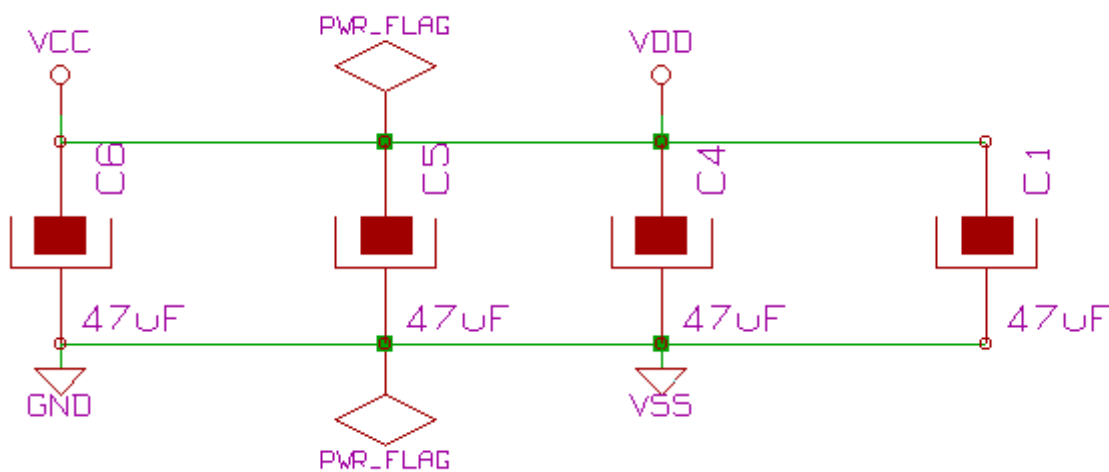
Все невидимые выводы питания с одинаковыми именами автоматически соединяются между собой без дополнительных оповещений. Однако эти автоматические соединения должны быть дополнены:

- Соединениями с другими видимыми выводами, соединенными с этим портом питания.
- Возможно, соединениями между группами невидимых выводов с другими именами. Например, общесхемные (земляные) выводы, обычно называются “GND” у TTL-компонент и “VSS” у MOS, и они должны быть соединены между собой.

Для таких соединений вы должны использовать символы портов питания - компонент, специально разработанных для подобного применения, которые вы можете создавать и модифицировать с помощью редактора схемной библиотеки. Эти символы состоят из невидимых выводов питания, ассоциируемых с необходимым рисунком.

Не используйте метки, которые имеют возможности только локального соединения (local) и не будут соединять невидимые выводы питания. Смотрите раздел о иерархической концепции схем, где этот вопрос рассматривается более детально.

Рисунок ниже показывает пример соединения портов питания.



В этом примере земля GND соединена с портом питания VSS, а порт питания VCC соединен с VDD.


Два символа PWR\_FLAG видимы. Они показывают, что два порта питания VCC и GND в действительности присоединены к источнику питания.

Без этих двух флагов инструмент ERC диагностирует : **Warning: power port not powered** (*Предупреждение: порт питания не запитан*).

Все эти символы – компоненты схемной библиотеки *power*.

#### 5.5.5 NoConnection-символы



Такие символы схемы нужны для ERC-контроля схемы (для проверки электрических соединений), чтобы избежать нежелательных предупреждений (электрические правила проверяются, чтобы гарантировать, что нет случайно оставленных не присоединенными выводов).

Если выводы специально остаются свободными (оборванными), необходимо разместить на этих выводах символ *NoConnection* (инструмент ) . Эти символы, однако, не имеют никакого влияния на генерируемые netlists-списки соединений.

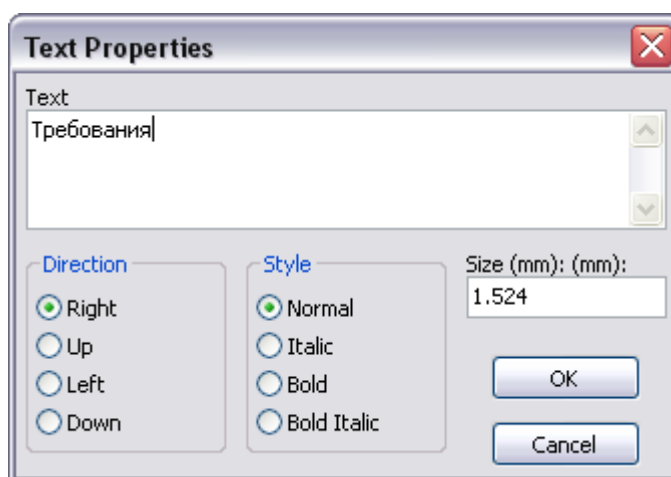
## 5.6 Элементы оформления схемы

### 5.6.1 Текстовые надписи и структурные рамки

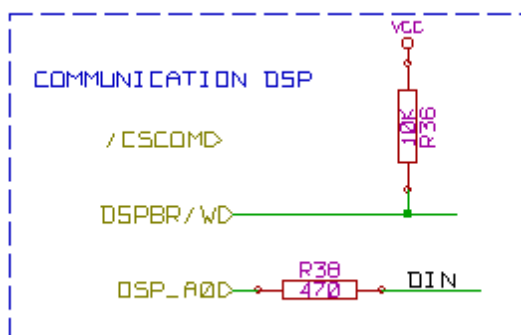
Для улучшения читабельности схемы может быть полезным размещение на схеме таких индикаторов, как текстовые надписи и структурные рамки (frames).

С этой целью используются такие инструменты оформления схемы, как текстовые надписи (инструментальная кнопка ) и штриховые линии (инструментальная кнопка ) , не имеющие электрического смысла, в противоположность меткам и проводникам, являющимся элементами схемных соединений.


Пример текстового ввода:



Пример структурной рамки :



### 5.6.2 Штамп форматной рамки схемы

Штамп с основной надписью форматной рамки редактируется инструментом , вызывающим диалоговое окно для настройки страницы схемы, которое в русскоязычном варианте выглядит так:

**Настройки страницы**

Размер страницы:

☐ Размер A4  
☒ Размер A3  
☐ Размер A2  
☐ Размер A1  
☐ Размер A0  
☐ Размер A  
☐ Размер B  
☐ Размер C  
☐ Размер D  
☐ Размер E  
☐ Пользовательский размер

Пользов. размер страницы по X: 430.00

Пользов. размер страницы по Y: 276.00

Число листов: 1    Номер страницы: 1

Изменение №: 1 ☐ Экспорт в другие листы

Наименование: Сборочный чертеж ☐ Экспорт в другие листы

Организация: Гидромонтаж ☐ Экспорт в другие листы

Децимальный номер: АБСЕ-1234.67-СБ ☐ Экспорт в другие листы

Разработал: Петров ☐ Экспорт в другие листы

Проверил: Иванов ☐ Экспорт в другие листы

Утвердил: Сидоров ☐ Экспорт в другие листы

OK (O)    Отмена (C)

Полный штамп в итоге смотрится так:

					АБСЕ – 1234.67 – СБ				
					Сборочный чертеж				
Изм.	Лист	№ докум.	Подп.	Дата					
Разраб.		Петров							
Проя.		Иванов							
Т.контр.									
					Гидромонтаж				
Н.контр.									
Утв.		Сидоров							

Номер листа, полезный в иерархических схемах, автоматически обновляется функцией формирования аннотации схемы.

## 6. Иерархические схемы


### 6.1 Применение

Иерархическое изображение схемы используется, главным образом, для больших проектов. Если вы хотите управлять такого рода проектом, необходимо :

- Использовать большие листы, которые могут создать проблемы при печати и сопровождении.
- Использовать несколько листов, которые выльются для вас в иерархическую структуру.

Полная схема в последнем случае состоит из основного листа схемы, называемого корневым листом, и суб-листов, составляющих иерархию.

Более того, искусное разбиение проекта на отдельные листы зачастую улучшает его восприятие. От корневого листа вы должны иметь возможность найти все суб-листы.

Управление иерархическими схемами в EESchema упрощается благодаря интегрированному навигатору (hierarchy navigator – навигатор по иерархии схемы), который активируется клавишей  верхней и правой инструментальных панелей.

Фактически есть два типа иерархии схем, которые могут существовать одновременно:

Первый только что был рассмотрен и широко используется.

Второй состоит в создании компонент в библиотеке, которые появляются подобно традиционным компонентам в схеме, но фактически соответствует схеме, описывающей внутреннюю структуру. Второй тип скорее используется при проектировании интегральных микросхем, поскольку в этом случае вам приходится использовать функциональные библиотеки элементов схемы, которую вы разрабатываете. EESchema в настоящий момент не обрабатывает этот второй тип иерархии схемы.

В схемах поддерживается иерархия трех видов:


- простая – заданный лист используется один раз;
- сложная – заданный лист используется более одного раза;
- плоская – это простая иерархия, но соединения между листами не прорисовываются.

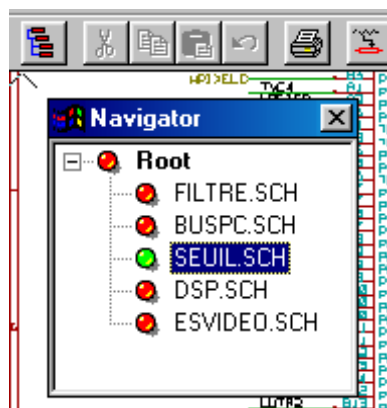
При создании иерархической схемы вся иерархия поддерживается, начиная с корневой схемы, как если бы это была только единственная схема.

Два момента, о которых следует знать :

- Как создать суб-лист схемы.
- Как строить электрические соединения между суб-листами.


## 6.2 Навигация по иерархии

Просмотр иерархической схемы легко выполняется благодаря инструменту навигатора (инструмент  горизонтальной инструментальной панели), показанному ниже:



Каждый лист схемы доступен щелчком по его имени.


### **Быстрая навигация:**


Щелкните правой клавишей по имени листа и выберите **enter sheet** – ввести лист. Вы также быстро можете добраться до корневого листа или суб-листа, благодаря инструменту  правой вертикальной инструментальной панели.

После выбора инструмента :

- Щелкните по имени листа для выбора этого листа.
- Щелкните где-нибудь еще для выбора основного листа.

## 6.3 Локальные, иерархические и глобальные метки

Локальные метки (инструмент ) связывают сигналы только одного листа схемы.

Иерархические метки (инструмент ) связывают сигналы только внутри листа и иерархические выводы, принадлежащие родственному листу.

Глобальные метки (инструмент ) обеспечивают сквозную связь сигналов по всей иерархии схемы.

Выводы питания (типа **power in** и **power out**) подобны глобальным меткам, так как также обеспечивают сквозные соединения в схемной иерархии.

В одной иерархии (простой или сложной) могут использоваться как иерархические, так и глобальные метки.

## 6.4 Создание и заголовок иерархической схемы


Вы должны :

- Разместить на корневом листе символ иерархии, названный **sheet symbol** – символом листа.
- Войти в новую схему (суб-листа) с помощью навигатора и нарисовать ее, подобно другим схемам.
- Отрисовать электрические соединения между двумя схемами, располагая глобальные метки – **Global Labels (Glables)** на новой схеме (суб-листе), и метки, имеющие те же имена на корневом листе, названные **SheetLabels** (метки листов). Эти метки листов будут соединены с символами листов корневого листа, с другими элементами схемы, подобно стандартным выводам компонент.

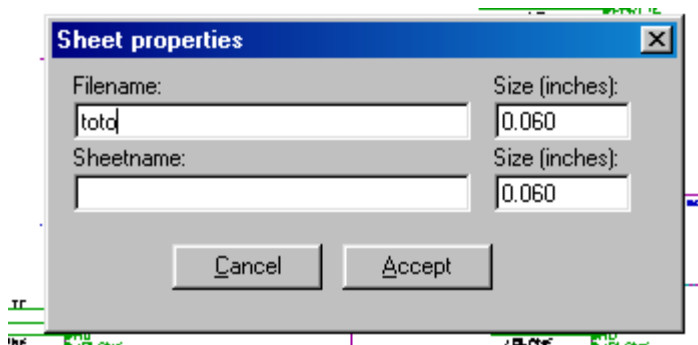
## 6.5 Символ листа (Sheet symbol)

Нарисуйте прямоугольник, обозначенный двумя диагональными точками, символизирующими суб-лист.

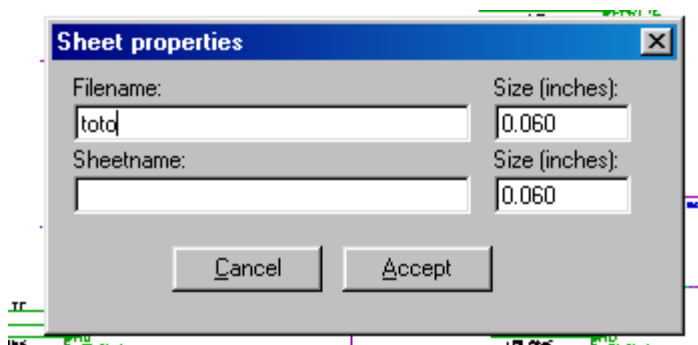
Размер этого прямоугольника должен позволить вам позже разместить специальные метки, иерархические выводы, соответствующие глобальным меткам (Glables) на суб-листе.

Эти метки похожи на обычные выводы компонента. Выберите инструмент . Щелкните для размещения верхний левый угол прямоугольника. Щелкните еще раз для размещения нижний правый угол, чтобы достаточно увеличить прямоугольник.

Пример :



Затем вам будет предложено ввести имя файла и имя листа для этого суб-листа (для поиска соответствующей схемы, используйте навигатор по иерархии).





Вы должны задать хотя бы имя файла. Если нет имени листа, имя файла будет использовано, как имя листа (обычный способ выполнения).

## 6.6 Соединения: иерархические выводы

Вы создадите здесь точки соединений (иерархические выводы) для символа, который был только что создан.


Эти точки соединения похожи на обычные выводы компонента, однако с возможностью присоединения к полной шине через единственную точку соединения.

Есть два способа сделать это:

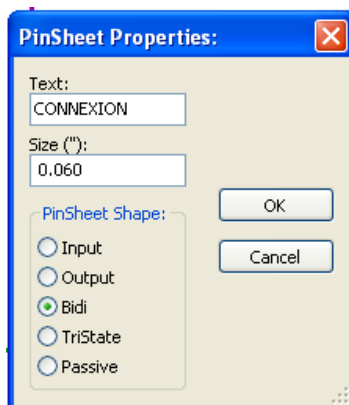
- Разместите разные выводы **перед** прорисовкой суб-листа (ручное размещение).
- Разместите разные выводы и глобальные метки **после** прорисовки суб-листа (полу-автоматическое размещение).

Второе решение предпочтительнее, если оно возможно.

### Ручное размещение:

- Выберите инструмент .
- Щелкните по символу иерархии, где хотели бы разместить этот вывод.

Посмотрите ниже пример создания иерархического вывода, названного “CONNEXION”.

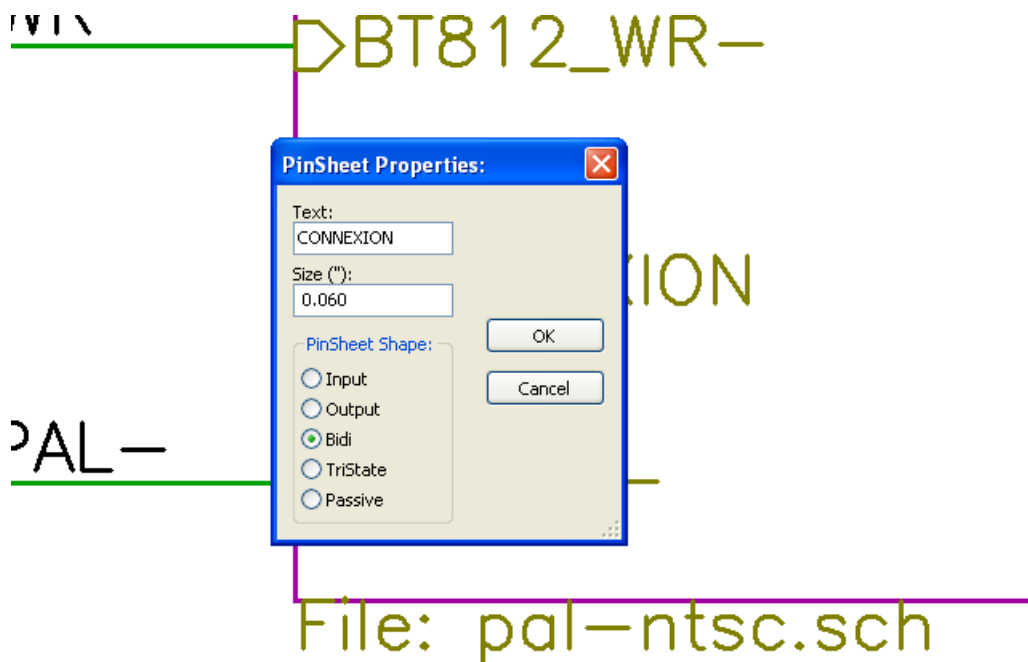


Вы можете затем определить его графические атрибуты и размер - щелкните правой клавишей и выберите функцию **Edit** в выпадающем меню ниже.


Доступны разные символы выводов :

- **Input** (ввод)
- **Output** (вывод)
- **BiDir** (двунаправленный)
- **TriState** (с тремя состояниями)
- **NotSpecified** (не определен)

Такие символы выводов - только графические элементы и не играют другой роли.




### Автоматическое размещение:

- Выберите инструмент .
- Щелкните по символу иерархии там, откуда вы хотите импортировать вывод, соответствующий размещению глобальной метки на соответствующей схеме. Появится иерархический вывод, если новая глобальная метка существует, то есть, не соответствует уже размещенному выводу.
- Щелкните там, где вы хотите разместить этот вывод.

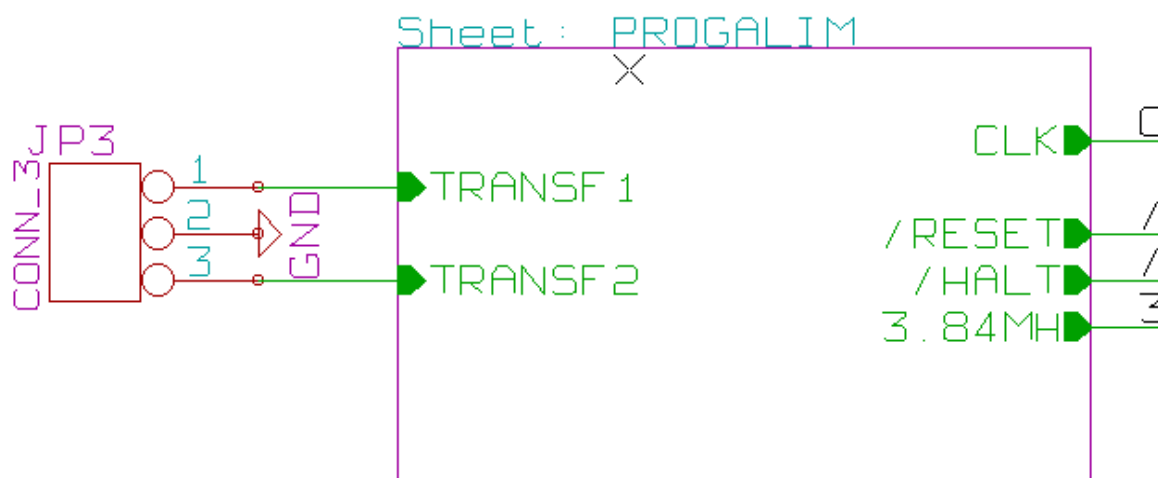
Все необходимые выводы могут таки образом размещаться быстро и без ошибок. Их вид соответствует соответствующим глобальным меткам.

## 6.7 Соединения: иерархические метки

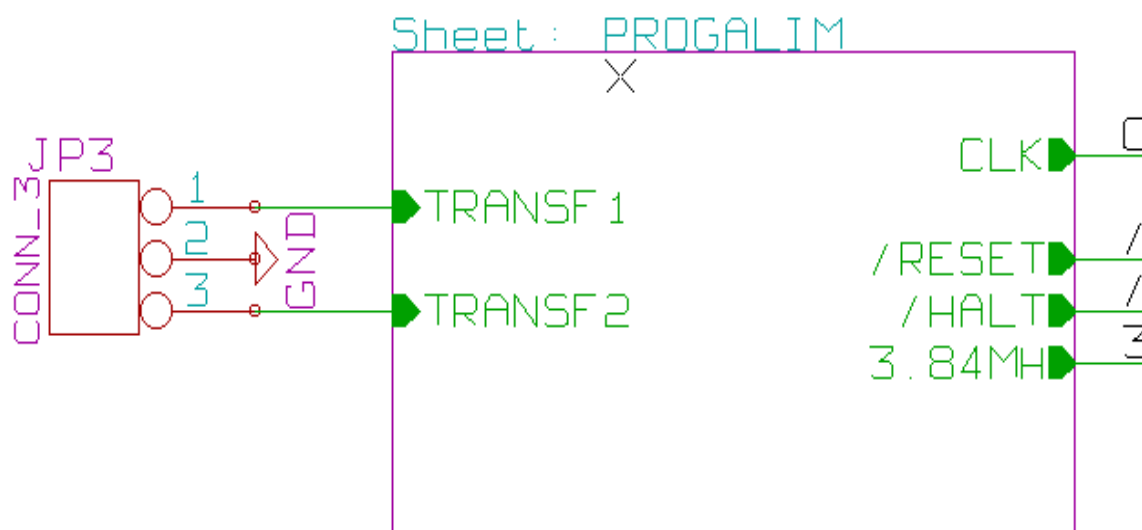
Каждый только созданный вывод символа листа должен соответствовать метке, называемой иерархической меткой на суб-листе, или **HLabel**.

Hlabels похожи на обычные метки, но они обеспечивают соединение между суб-листами и корневым листом. Графическое представление этих двух дополнительных меток (вывода и HLabel) схоже. Создание HLabel выполняется инструментом .

Ниже приведен пример корневого листа:



Заметьте, что выходы TRANSF1 и TRANSF2 соединены с коннектором JP3. Вот так выглядят соответствующие соединения на суб-листе :



Вы вновь найдете две соответствующие метки типа Hlabels, обеспечивающие соединение между двумя иерархическими листами.

**Заметьте:** Вы можете использовать метки HLabels и иерархические выходы для соединения двух шин, соответственно с синтаксисом (Bus[n...m]), описанным ранее.

#### 6.7.1 Метки простые, Hlabels и невидимые выходы питания

Вот несколько пояснений по разным способам обеспечения соединений отличных от соединений проводниками.

##### 6.7.1.1 Простые метки

Простые метки имеют локальную емкость соединения, то есть, ограничены листом схемы, на котором расположены. Это происходит потому, что :

- Каждый лист имеет номер листа.
- Этот номер листа связан с меткой.

Итак, если вы размещаете метку “TOTO” на листе № 3, фактически, настоящая метка это “TOTO\_3”.

Если вы также размещаете метку “TOTO” на листе № 1 (корневой лист), вы фактически размещаете метку, названную “TOTO\_1”, отличную от “TOTO\_3”. Это всегда верно, даже если есть всего один лист.

#### 6.7.1.2 Иерархические метки HLabels

То, что было сказано относительно простых меток, истинно и для глобальных меток или HLabels.

Таким образом, на одном и том же листе HLabel “TOTO” считается соединенной с локальной меткой “TOTO”, но не соединенной с HLabel или меткой с именем “TOTO” на другом листе.

Однако, HLabel считается, как бы соединенной с соответствующим символом SheetLabel в иерархических символах, помещенных на корневой лист.

#### 6.7.1.3 Невидимые выводы питания

Было показано, что такие выводы соединены вместе, если они имели одинаковые имена. Таким образом, все выводы питания, объявленные **Invisible Power Pins** и названные VCC, соединены и формируют эквипотенциал VCC, где бы на листе они ни располагались.

Этим объясняется то, что если вы располагаете метку VCC на суб-листе, она не будет соединена с выводом VCC, поскольку эта метка фактически VCC\_n, где n – номер листа.

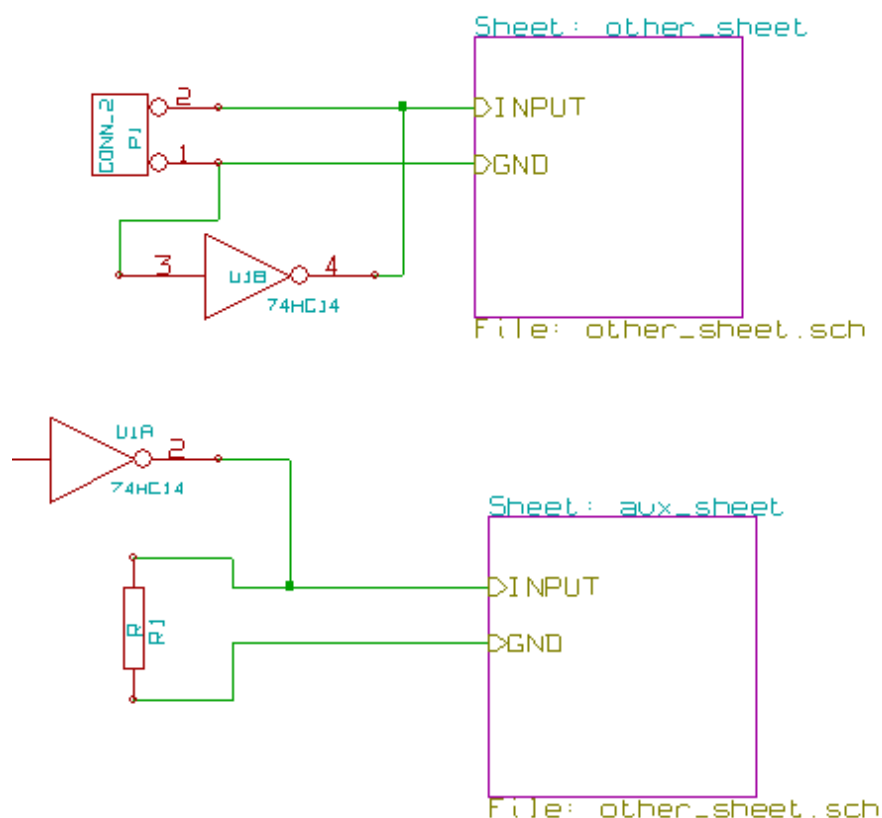
Если вы хотите, чтобы эта метка VCC была реально соединена с эквипотенциалом VCC, она должна быть явно присоединена к невидимому выводу питания при помощи VCC-порта питания.

#### 6.7.2 Глобальные метки GLabels

Глобальные метки с идентичными именами обеспечивают сквозные соединения в схеме (метки питания подобные VCC являются глобальными метками).

### 6.8 Комплексная иерархия схем

На примере ниже некоторая схема используется дважды (два вхождения). Два листа разделяют одну и ту же схему, поэтому имя файла схемы одинаково для обоих листов (“other\_sheet\_sch”). Но имена листов должны быть разными.

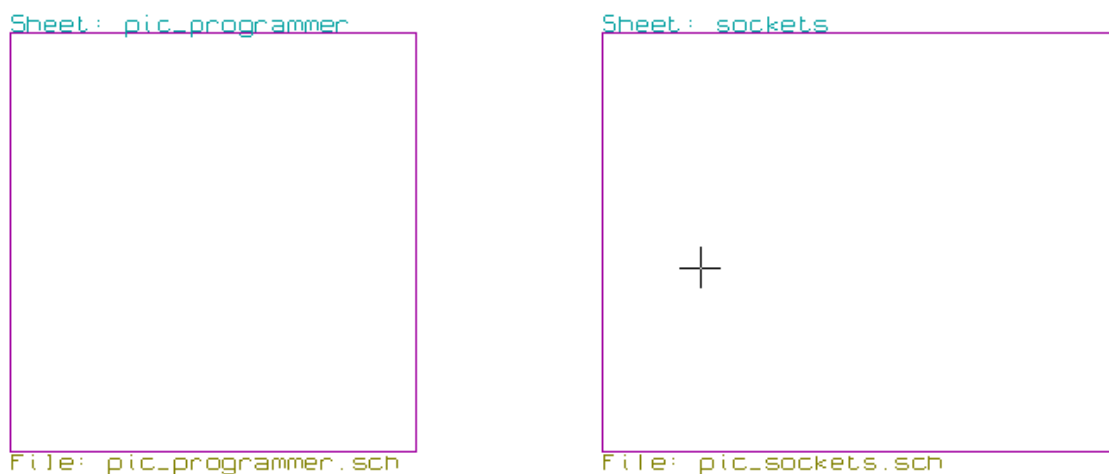


## 6.9 Плоская иерархия

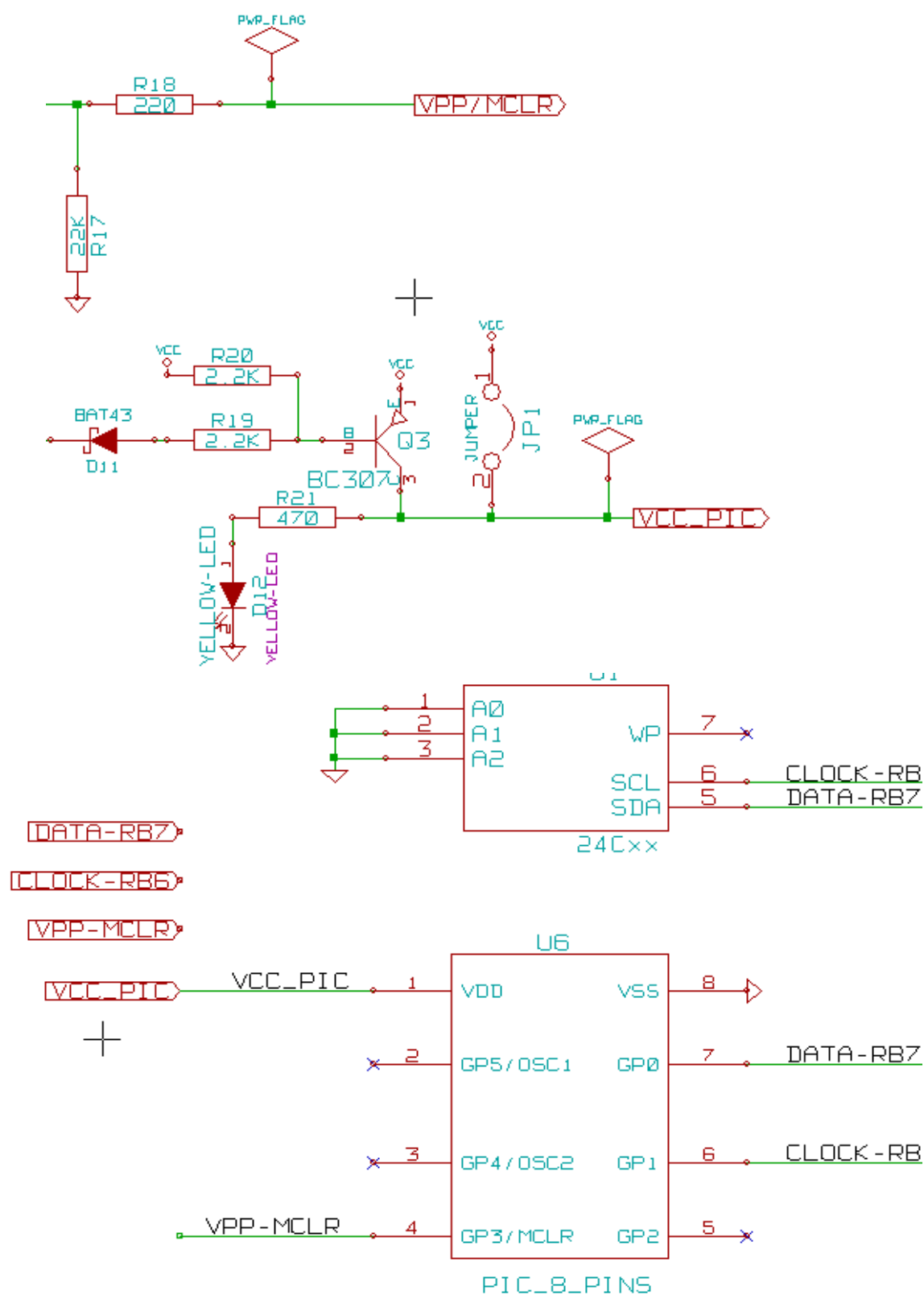
Возможно создать проект на нескольких листах, не создавая соединений между этими листами, если следовать таким правилам:

- Нужно создать корневой лист, включающий другие листы, который обеспечит связь между этими листами;
- Не выполнять явные соединения;
- Все соединения между листами строятся на глобальных метках вместо иерархических.

Здесь показан корневой лист:



Ниже приведены две страницы, связанные глобальными метками:



**Глобальные метки:**


DATA-RB7

CLOCK-RB6

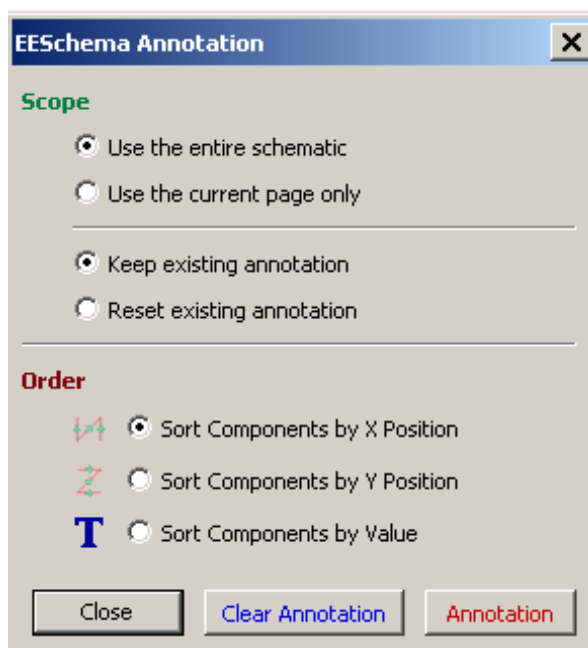
VPP-MCLR

## 7. Автоматическая классификация позиционных обозначений

### 7.1 Назначение функции аннотирования схемы

Эта функция, вызываемая с помощью инструмента , позволяет вам автоматически назначить позиционные обозначения для компонентов схемы, а для многоэлементных компонентов назначить общий суффикс, чтобы минимизировать количество корпусов на плате.

Меню функции:

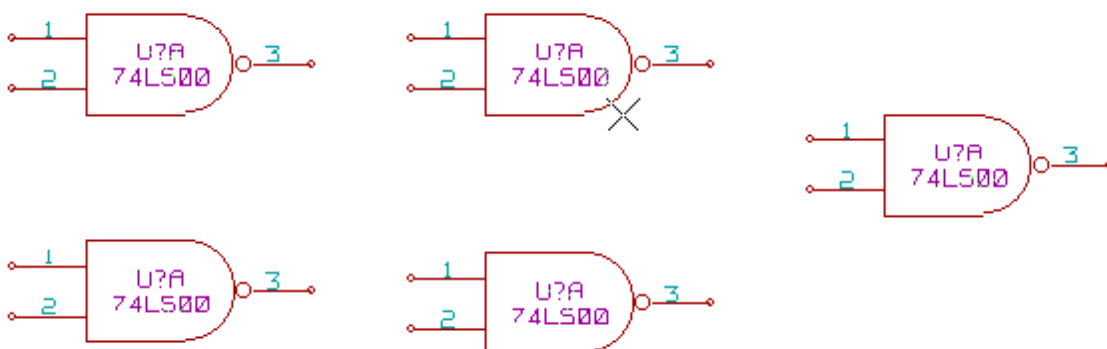


Доступны разные варианты аннотирования:

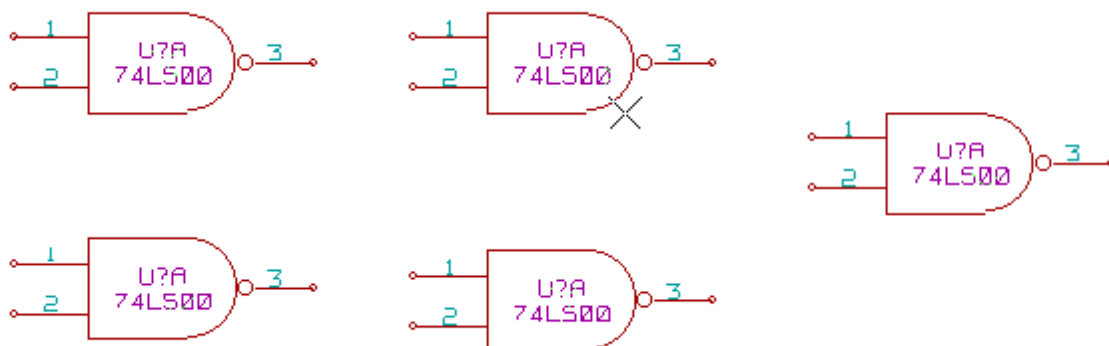
- Аннотировать всю схему (**Use the entire schematic**).
- Аннотировать только текущий лист (**Use the current page only**).

Автоматическое аннотирование обращается ко всему проекту (все листы) или только к новым компонентам, если вы не хотите модифицировать предыдущие аннотации (включена опция **Reset existing annotation**).

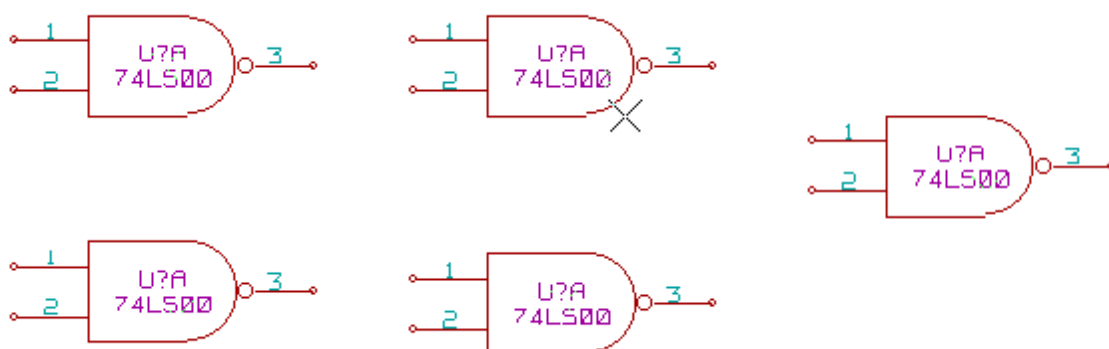
### 7.2 Пример применения



Пример выше показывает размещение 5 логических элементов, но без аннотации. После аннотирования получим:



Сортировка по X



Сортировка по Y

Можно увидеть, что четыре вентиля 74LS00 были обозначены по корпусу U1, и что пятый 74LS00 был обозначен по следующему, U2.

## 8. Проверка схемы типа ERC

### 8.1 Обзор функции ERC

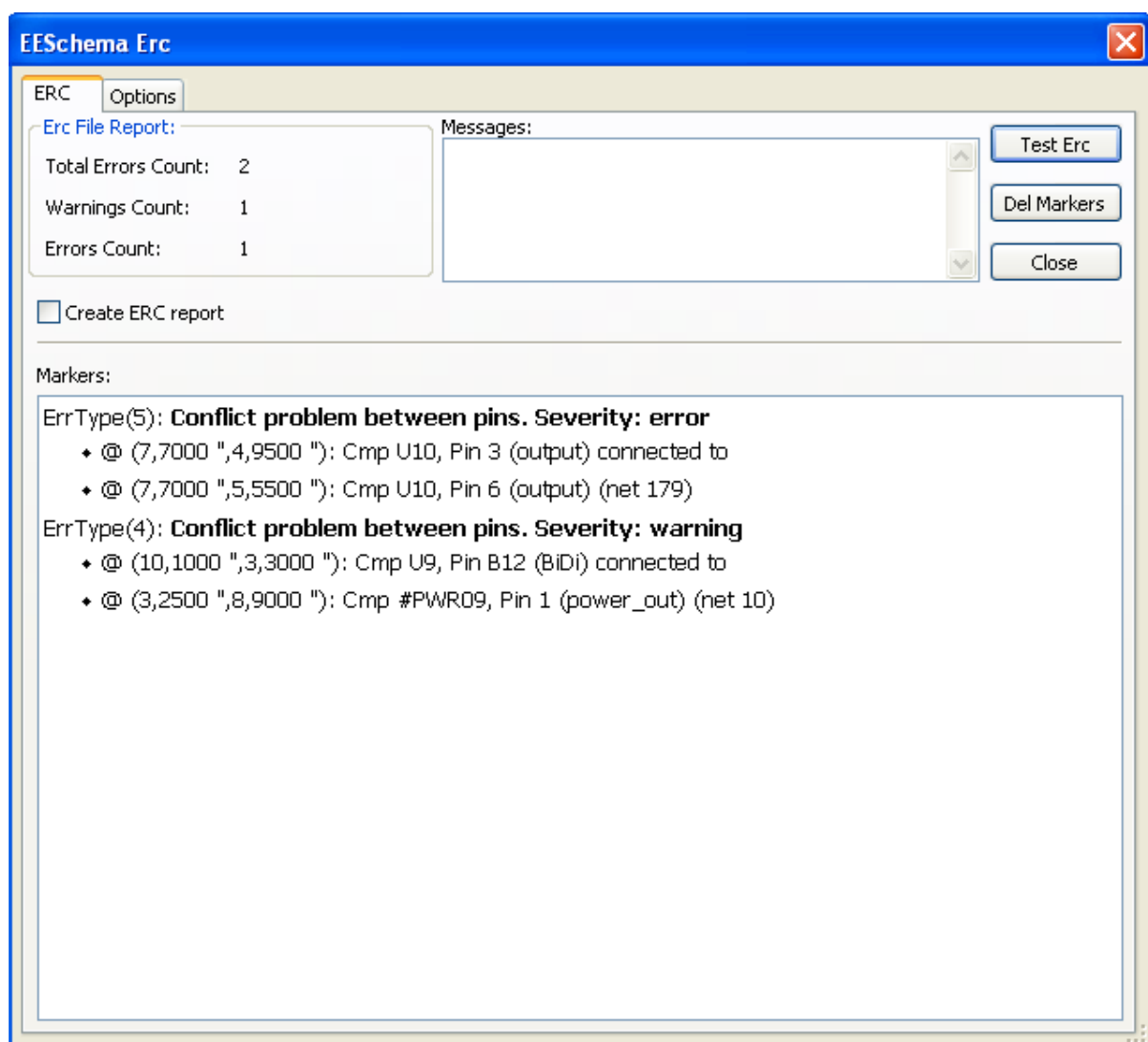
Функция **ERC (Electrical Rules Check)** выполняет автоматическую проверку схемы. Она выявляет различные ошибки на листах схемы, такие как не присоединенные выводы, не присоединные иерархические символы, закороченные выходы.

Естественно, автоматическая проверка не идеальна, а программа, позволяющая выявить абсолютно все ошибки в схеме, еще не написана. Но подобный контроль очень полезен, поскольку позволяет вам обнаружить множество промахов и небольших ошибок.


Фактически все обнаруженные ошибки должны быть проверены и исправлены для нормального продолжения. Качество проверки прямо зависит от аккуратности, с которой были объявлены электрические выводы при



создании библиотеки. ERC ошибки в отчетах декларируются, как **errors** (*ошибки*) или **warnings** (*предупреждения*)



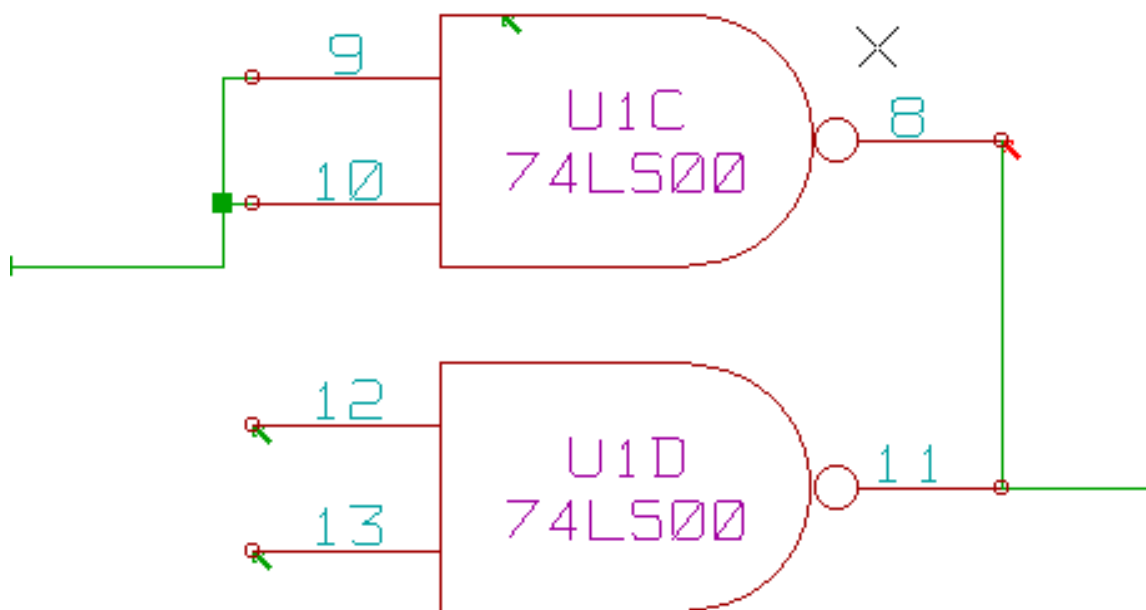
## 8.2 Применение ERC-теста

Диалоговое окно ERC-проверки схемы запускается иконкой . Активация теста схемы выполняется по команде **Test Erc**.

На элементах схемы (выводах или метках), вызвавших ошибки ERC, рисуются маркеры в виде наклонных стрелок. Щелкните по маркеру, чтобы увидеть соответствующую диагностику от ERC.

Вы можете также удалить маркирование ошибок в этом меню по команде **Del Markers**.

## 8.3 Пример ERC-теста

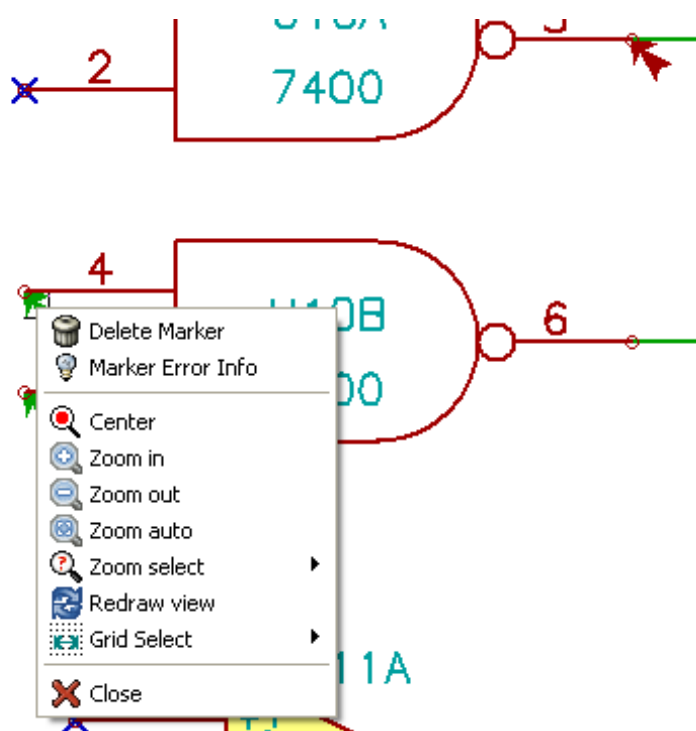


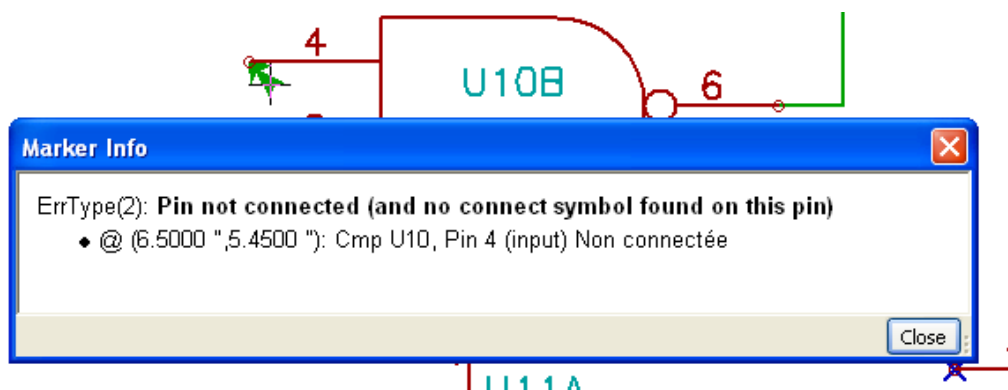
*Можно увидеть 4 ошибки :*

- Два выхода соединены вместе.
- Два входа остались неприсоединенными.
- Ошибка с **невидимым** портом питания (пропущен флаг питания).

#### 8.4 Вывод диагностики для маркера

Правый клик по графическому маркеру в виде стрелки выдает выпадающее меню для доступа к функциям отображения диагностик маркера. После этого вызовите функцию **Marker Error Info**.





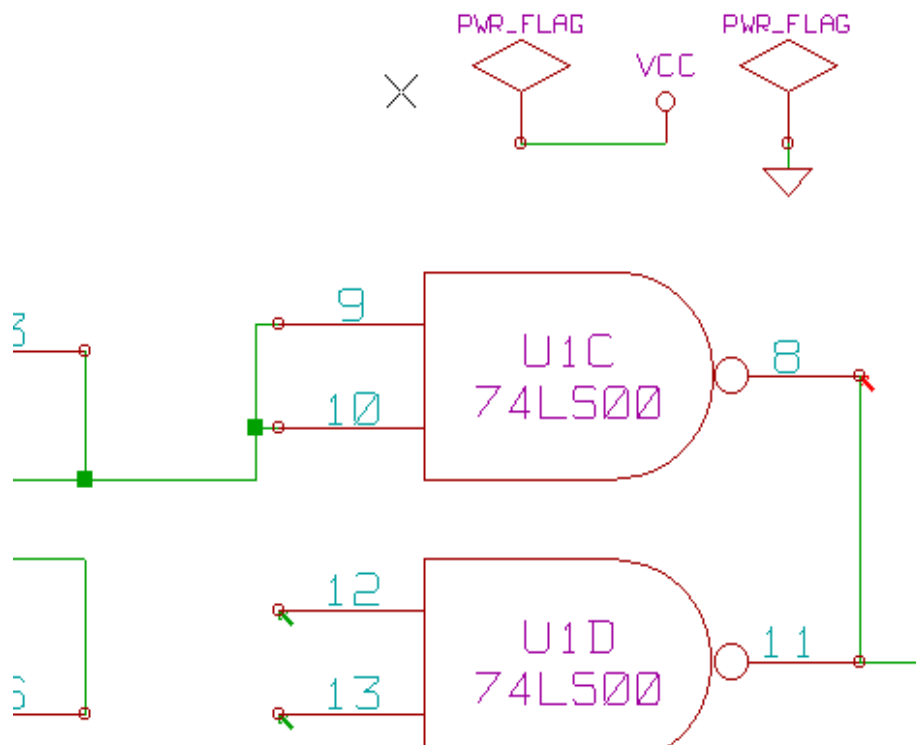
## 8.5 Выводы и флаги питания

Нет ничего необычного получить предупреждения на выводах питания, тогда как все выглядит нормально (посмотрите на пример выше).

Для большинства разработок причина этого в том, что питание подается на разъемы, которые не являются источниками питания (подобно выходу регулятора, который декларируется как **Power out** – выход питания).

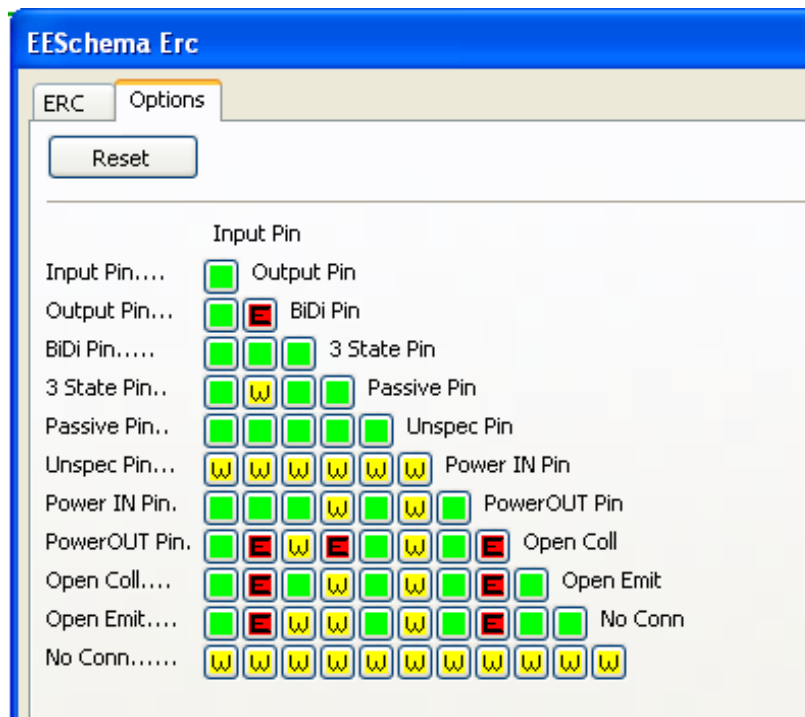
Таким образом, ERC не обнаружит никаких выводов выхода питания для проверки этих соединений и объявит их неподключенными.

Вы должны разместить **PWR\_FLAG** на таких портах питания. Он символизирует фактический выход питания. Маркер ошибки в этом случае исчезает.



### 8.3 Конфигурация для ERC-теста

Панель **Options** позволяет вам сконфигурировать правила выполнения соединений в схеме, чтобы задать условия для вывода сообщений об ошибках и предупреждений:



Формируемые правила могут быть изменены щелчком по выбранному квадрату матрицы, заставляя его циклически проходить через выбор одного из состояний: *нормально, предупреждение, ошибка* (normal, warning, error).

### 8.4 Файл отчета ERC-теста

Файл отчета ERC может быть сохранен выбором опции **Create ERC Report**. Расширение для файла отчета - .erc.

Вот пример:

```
ERC control (4/1/2008-14:16:4)

***** Sheet 1 (INTERFACE UNIVERSAL)
ERC: Warning Pin input Unconnected @ 8.450, 2.350
ERC: Warning passive Pin Unconnected @ 8.450, 1.950
ERC: Warning: BiDir Pin connected to power Pin (Net 6) @ 10.100, 3.300
ERC: Warning: Power Pin connected to BiDir Pin (Net 6) @ 4.950, 1.400

>> Errors ERC: 4
```

## 9. Создание Netlist-списка цепей схемы

### 9.1 Обзор функции

Эта команда позволяет вам создать файл описания электрических цепей (netlist) для вашей разработки.

Netlist – это файл, который описывает электрические соединения между компонентами. Он содержит:

- Список компонент;
- Список соединений между компонентами, называемых эквипотенциалами цепей.

Есть разные форматы netlist. Иногда список компонентов и список эквипотенциалов – это два отдельных файла.


Netlist важен при использовании программ ввода схем, поскольку он необходим для перехода к другим программам САПР электроники:

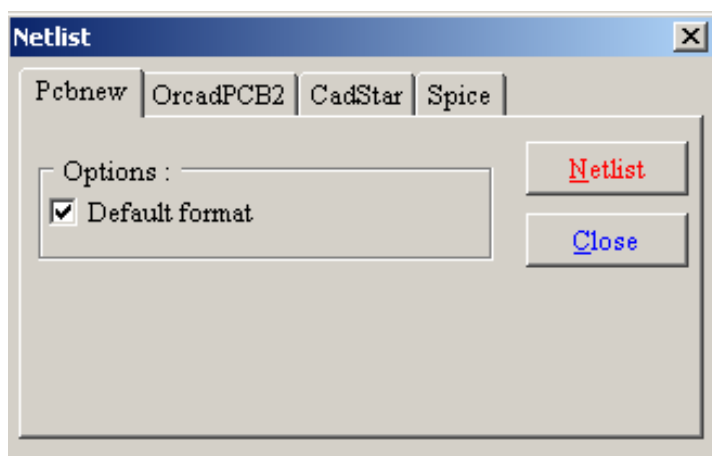
- для проектирования (разводки) печатных плат;
- для схемотехнического моделирования;
- для компиляторов ПЛИС (типа PAL и других программируемых микросхем).

Eeschema поддерживает несколько форматов netlist :

- PCBNEW формат (печатные платы).
- ORCAD PCB2 формат (печатные платы).
- CADSTAR формат (печатные платы).
- Spice формат для симуляторов схем типа Spice (и других).

### 9.2 Формирование списка цепей Netlist для PCBNEW

Выберите с панели кнопок инструмент  для открытия диалогового окна создания netlist :

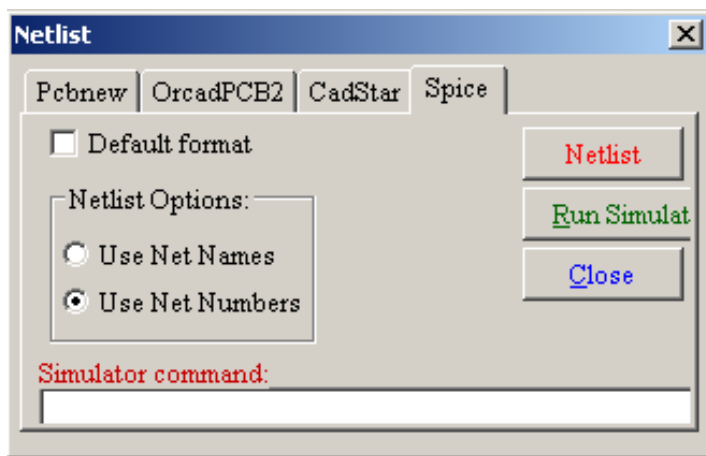


- Выбран Pcbnew вывод

# Структура полученного netlist для редактора печатных плат PCBNEW:

# EESchema Netlist Version 1.0 generee le 21/1/2007-16:51:15

```
(
(32E35B76 $noname C2 1NF {Lib=C}
(1 0)
(2 VOUT_1)
)
(32CFC454 $noname V2 AC_0.1 {Lib=VSOURCE}
(1 N-000003)
(2 0)
)
(32CFC413 $noname C1 1UF {Lib=C}
(1 INPUT_1)
(2 N-000003)
)
(32CFC337 $noname V1 DC_12V {Lib=VSOURCE}
(1 +12V)
(2 0)
)
(32CFC293 $noname R2 10K {Lib=R}
(1 INPUT_1)
(2 0)
)
(32CFC288 $noname R6 22K {Lib=R}
(1 +12V)
(2 INPUT_1)
)
(32CFC27F $noname R5 22K {Lib=R}
(1 +12V)
(2 N-000008)
)
(32CFC277 $noname R1 10K {Lib=R}
(1 N-000008)
(2 0)
)
(32CFC25A $noname R7 470 {Lib=R}
(1 EMET_1)
(2 0)
)
(32CFC254 $noname R4 1K {Lib=R}
(1 +12V)
(2 VOUT_1)
)
(32CFC24C $noname R3 1K {Lib=R}
(1 +12V)
(2 N-000006)
)
(32CFC230 $noname Q2 Q2N2222 {Lib=NPN}
(1 VOUT_1)
(2 N-000008)
(3 EMET_1)
)
(32CFC227 $noname Q1 Q2N2222 {Lib=NPN}
(1 N-000006)
(2 INPUT_1)
(3 EMET_1)
)
)
# End
```



(выбран Spice-вывод)

Переключая разные закладки, вы можете выбрать желаемый формат, как формат по умолчанию (**Default format**).

В формате Spice вы можете генерировать netlists либо с эквивалентными именами цепей (что предпочтительно), либо с номерами электрических цепей (старые версии Spice-симуляторов воспринимают только номера). После щелчка по клавише Netlist, вы получите вопрос об имени файла netlist, в который выполняется вывод списка цепей.

**Примечание :**

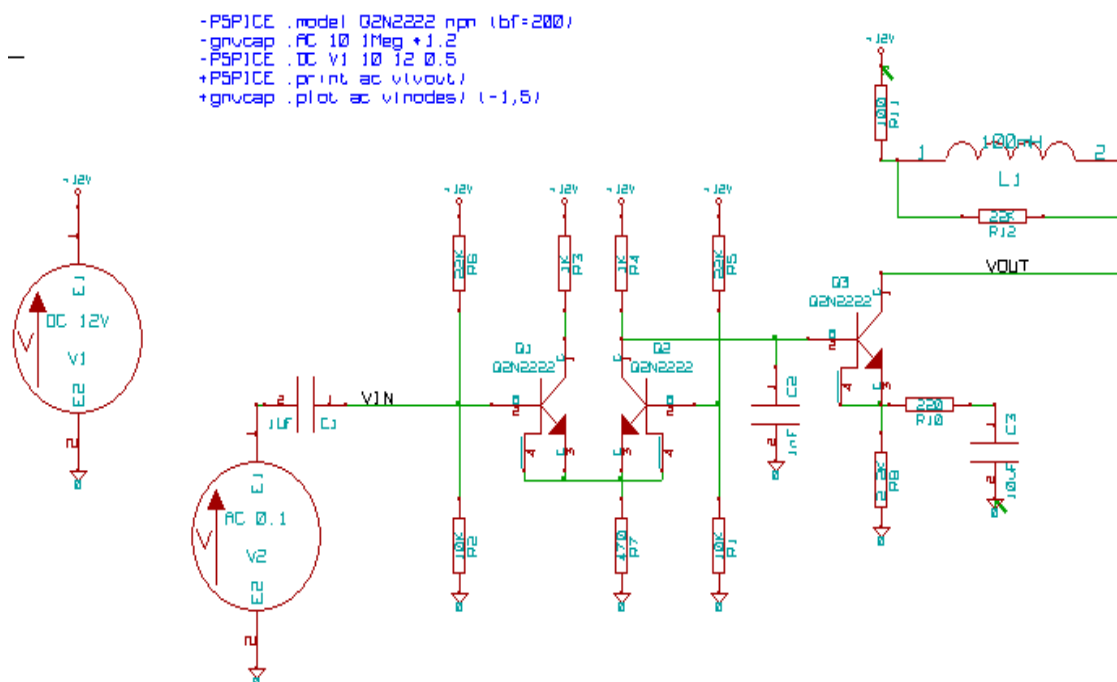
При больших проектах генерация netlist может занять несколько минут.

### 9.3 Взаимодействие с программами spice-моделирования схем

Редактор схем EESchema может взаимодействовать с программами Spice-моделирования схем, такими как PSPICE и GNUCAP. Задание на моделирование формируется в виде списка электрических цепей в формате Spice, дополненного управляющими директивами языка Spice по выбору режима и параметров моделирования и выводу его результатов.

В строке **Simulator command** диалогового окна Spice-вывода можно разместить команду по вызову программы моделирования схемы для заданного и дополненного чертежом файла соединений схемы.

Ниже вы можете увидеть схему, использующую библиотеку PSPICE:



В формате программы PSPICE netlist выглядит следующим образом:

\* EESchema Netlist Version 1.1 (Spice format) creation date: 18/6/2008-08:38:03

```

.model Q2N2222 npn (bf=200)
.AC 10 1Meg *1.2
.DC V1 10 12 0.5
R12 /VOUT N-000003 22K
R11 +12V N-000003 100
L1 N-000003 /VOUT 100mH
R10 N-000005 N-000004 220
C3 N-000005 0 10uF
C2 N-000009 0 1nF
R8 N-000004 0 2.2K
Q3 /VOUT N-000009 N-000004 N-000004 Q2N2222
V2 N-000008 0 AC 0.1
C1 /VIN N-000008 1uF
V1 +12V 0 DC 12V
R2 /VIN 0 10K
R6 +12V /VIN 22K
R5 +12V N-000012 22K
R1 N-000012 0 10K
R7 N-000007 0 470
R4 +12V N-000009 1K
R3 +12V N-000010 1K
Q2 N-000009 N-000012 N-000007 N-000007 Q2N2222
Q1 N-000010 /VIN N-000007 N-000007 Q2N2222
.print ac v(vout)
.plot ac v(nodes) (-1,5)
.end

```

Для симуляторов схем PSPICE и GNUCAP вы можете включить некоторые служебные строки текстовых команд языка Spice в сам netlist (**.PROBE**, **.AC**...).

Любые строки текста, включенные в чертеж схемы с ключевыми словами -PSPICE или -GNUCAP будут вставлены (без ключевого слова) в верхнюю часть netlist. Аналогично, текстовые строки с ключевыми словами



+PSPICE или +GNUCAP будут вставлены (без ключевого слова) в конец netlist. Например: если вы печатаете в схему следующий текст (а не метку!):

***-PSPICE .PROBE***, то строка ***.PROBE*** будет вставлена в начало netlist.

В предыдущем примере с помощью этой техники три управляющие строки были вставлены в начало spice-списка цепей и две - в конец.

Кроме того, заметим, что эквипотенциал GND для Pspice должен называться 0 (zero).

## 9.4 Замечания по применению функции Netlist

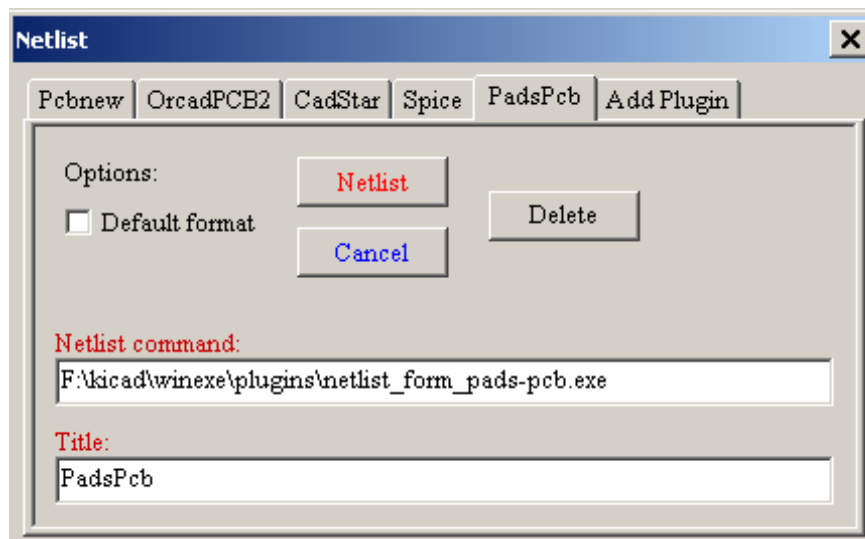
Многие версии программного обеспечения, которые используют netlists, не воспринимают пробелы в именах компонент, выводов, цепей и других. Избегайте пробелов в метках или именах и значениях полей.

Также некоторые символы, отличные от букв и цифр, могут вызвать проблемы.

Заметьте, что это ограничение относится не к EESchema, а к форматам netlist или к программам, которые используют эти netlists.

## 9.5 Применение Netlist-плагинов внешних программ

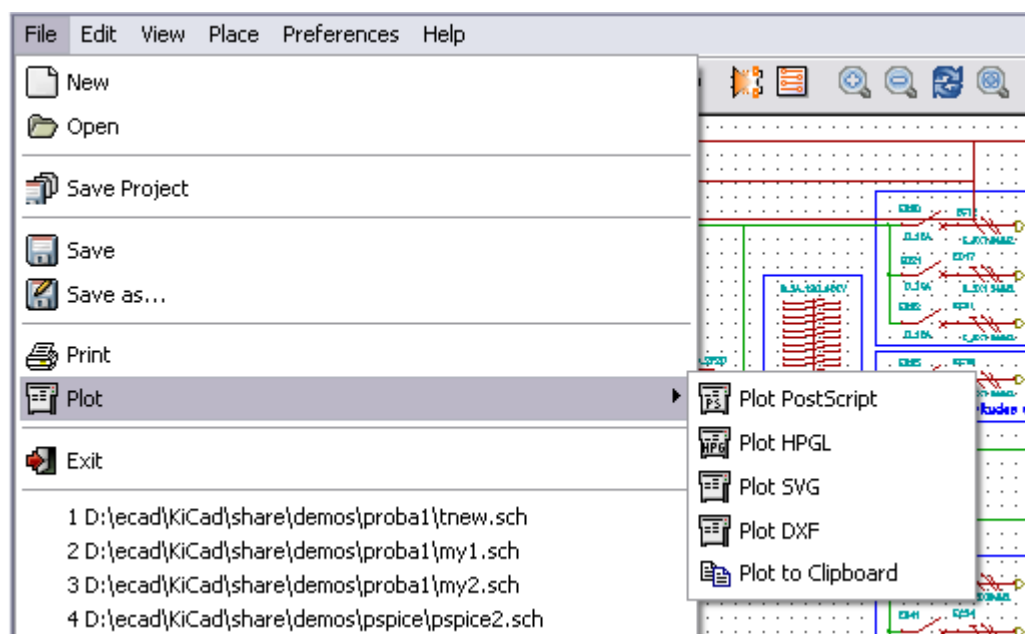
Дополнительные программы трансляции списка цепей можно подключить командой **Add Plugin** из меню программы. Ниже показана процедура подключения внешней программы формирования списка цепей для системы PADS PCB.



## 10. Черчение и печать схем

### 10.1 Обзор функций Plot/Print

Вы получаете доступ к этим командам через меню **File**:



Форматы вывода чертежей из редактора схем - это POSTSCRIPT, HPGL, SVG или DXF. Вы можете также передать чертеж или фрагмент схемы в буфер обмена или напечатать его непосредственно на принтере.

### 10.2 Общие команды печати

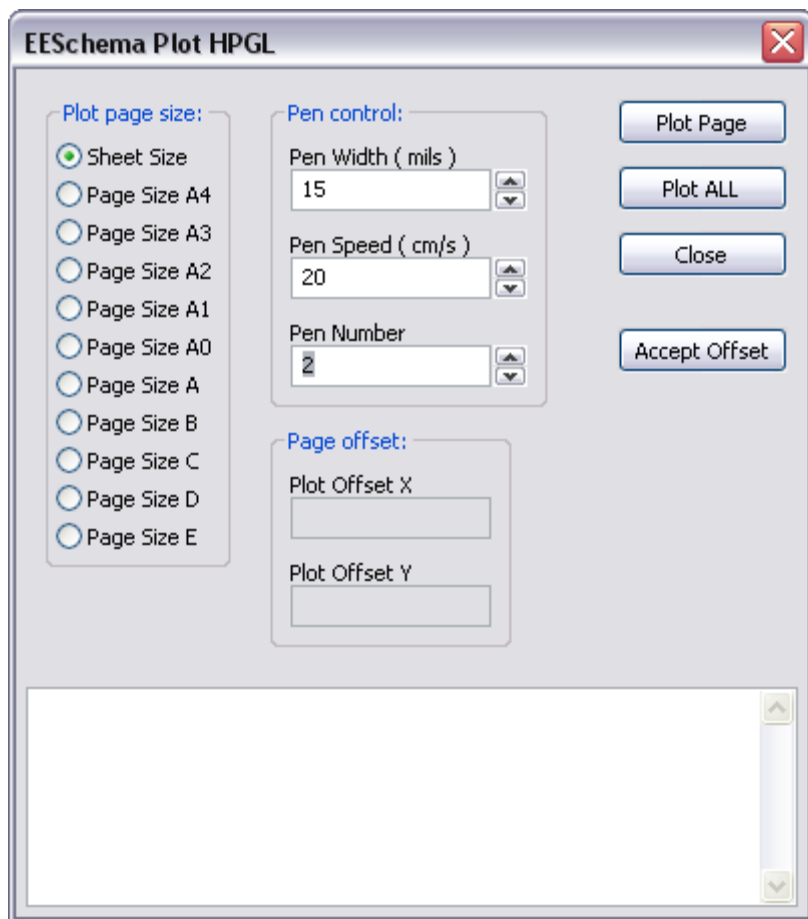
Общие команды можно задать в каждом из вариантов печати: PS, HPGL, SVG, DXF. Опция **Plot ALL** позволяет вывести на печать все листы иерархической схемы (файл формируется для каждого листа). Опция **Plot Page** формирует файл только для текущего листа схемы.

### 10.3 Черчение схемы в формате HPGL

Команда **Plot HPGL** позволяет вам создать чертежный файл в формате HPGL. Для этого формата вывода можно задать:

- Число перьев плоттера (Pen number)
- Толщину линий (Pen thickness) (в 0,001 дюйма).
- Скорость прорисовки (Drawing speed в см/сек).
- Размер листа (Sheet size).
- Смещение печати (Print offsets).

Окно диалога установок HPGL-плоттера выглядит так:



Имя выходного файла – это имя листа с расширением .plt.

### 10.3.1 Выбор размера листа

Для этого используйте **Sheet Size** (обычно выбрано). В этом случае будет использован размер листа, определенный в меню штампа, с масштабом 1.

Если выбран другой размер (от A4 до A0, или от A до E), автоматически выбирается масштаб заполнения страницы.

### 10.3.2 Регулирование смещения чертежа

Для всех стандартных размеров можно установить “offsets”, центрирующий чертеж так тщательно, как только возможно.

Поскольку плоттеры имеют собственные точки начала черчения в центре или в нижнем левом углу листа, необходимо иметь возможность ввести смещение с целью правильного вычерчивания.

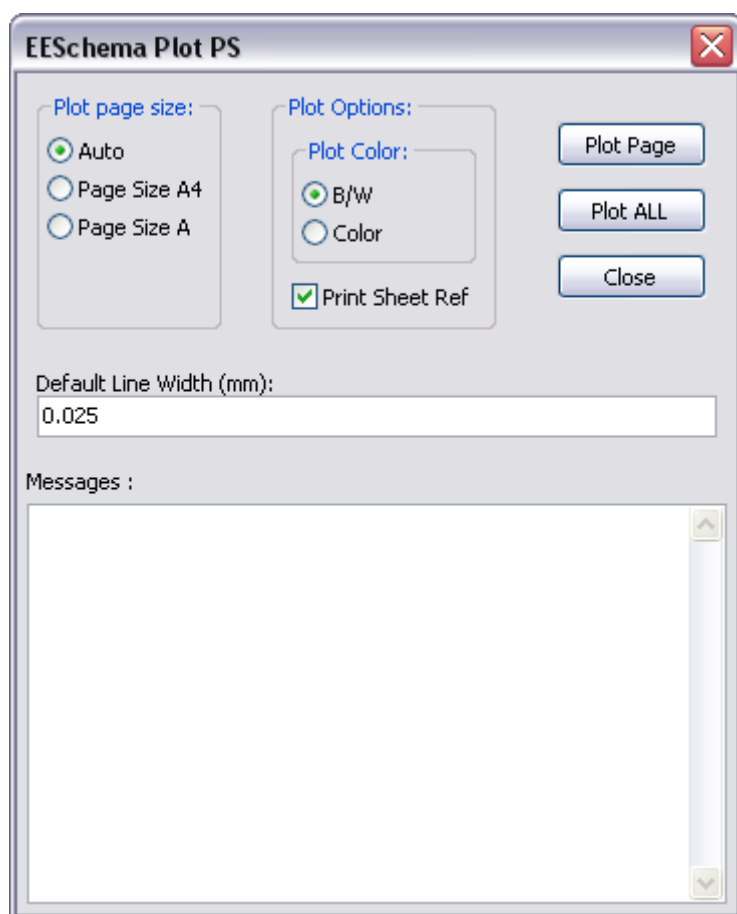
#### Обычно:

- Для плоттеров, имеющих свою собственную точку в центре листа, offsets должны быть отрицательны и установлены в половину размеров листа.
- Для плоттеров, имеющих свою собственную точку в левом нижнем углу, offset должен быть установлен близко к 0.

#### Установка смещения чертежа:

- Выберите размер листа.
- Установите **OffsetX** и **OffsetY**.
- Щелкните по клавише **Accept Offset**.

## 10.4 Черчение схемы в формате Postscript

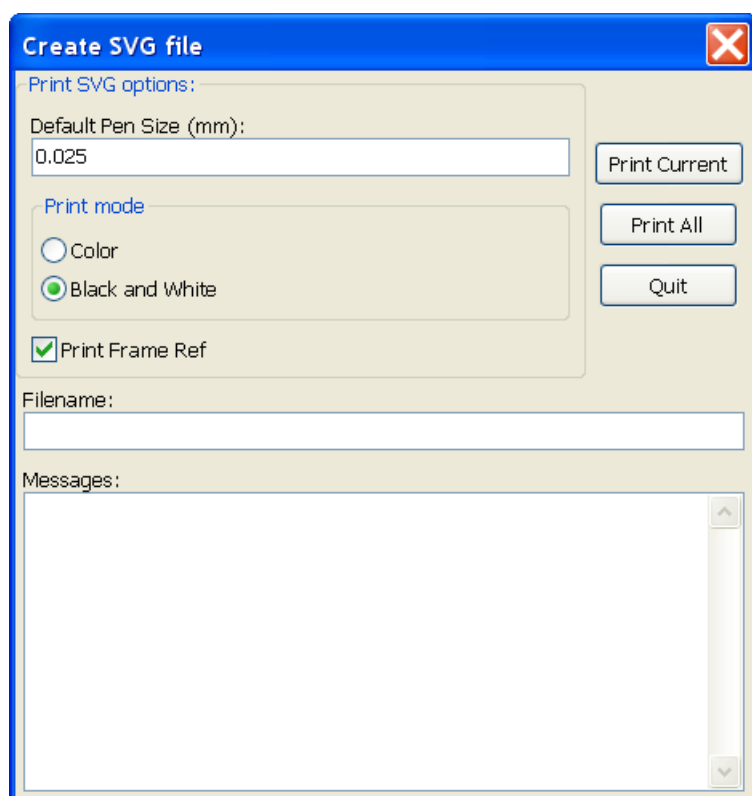


Эта команда позволяет вам создать файлы чертежа схемы в формате PostScript.

Имя файла – это имя листа с расширением .ps. Вы можете снять опцию : **Print Sheet Ref** (печать форматной рамки). Это полезно, например, если вы хотите создать файл postscript для инкапсуляции (формат .eps) при вставке схемы в тестовый редактор.

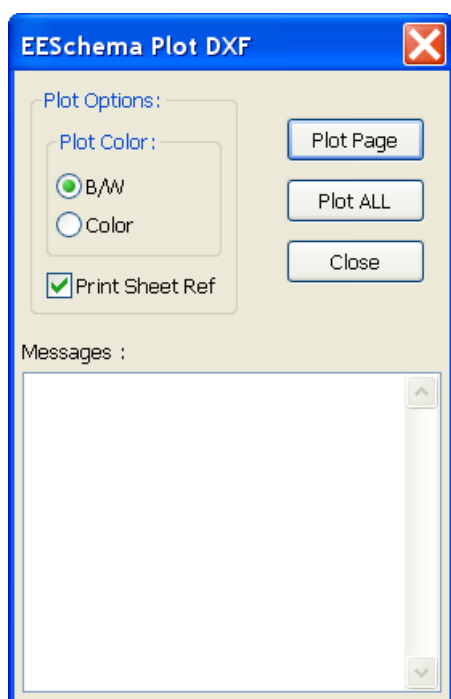
В окне сообщений Messages отображаются имена выходных файлов.

## 10.5 Черчение схемы в формате SVG




Эта функция позволяет создавать чертежные файлы схем в векторном формате SVG. Имена выходных файлов формируются из имени листа и расширения .svg.

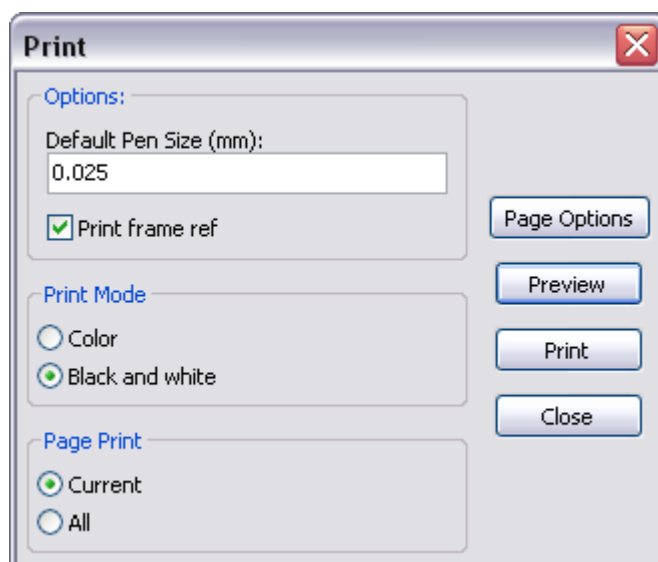
## 10.6 Черчение схемы в формате DXF



Эта функция позволяет создавать чертежные файлы схем в векторном формате DXF. Имена выходных файлов формируются из имени листа и расширения .dxf.

## 10.7 Печать схемы на принтере

Функция печати схемы (вызывается командой меню **File/Print** или инструментальной кнопкой ) похожа на предыдущую, она позволяет визуализировать и сгенерировать проектные файлы для стандартного принтера под LINUX и под Windows.



Опция **Page Print** позволяет вам печатать в режимах: **Current** - создает файл только для текущего листа и **All** - позволяет вывести всю иерархию схемы (файл генерируется для каждого листа).

Опция **Print Mode** позволяет выбрать цветную или черно-белую печать. В черно-белом режиме любой цвет, отличный от белого, печатается

черным. Эта опция обычно необходима, если вы используете черно-белый лазерный принтер, поскольку цвета при этом печатаются не очень отчетливыми полутонами.

Кнопка **Preview** обеспечивает предварительный просмотр схемы и позволяет оценить ее размещение на странице. **Page Options** задает параметры страницы печати. **Default Pen Size** – толщина следа пера (линии) принтера по умолчанию. **Print Frame Ref** – разрешает или запрещает печать форматной рамки схемы.

## 11. Управление библиотеками компонентов схемы

### 11.1 Основная информация о библиотеках

#### 11.1.1 Библиотеки схемных компонент

Все компоненты, используемые в схеме, описаны в библиотеках компонент. Чтобы иметь возможность разумно простого управления этими компонентами, некоторые библиотеки группируют компоненты по теме (по функциям или производителям).

Меню управления компонентами (Library management) позволяет вам создавать библиотеки, добавлять в них, или удалять из них, или трансформировать компоненты. Также, естественно, оно позволяет вам быстро отобразить компоненты библиотеки.

#### 11.1.2 Меню управления библиотекой

Есть два меню управления :

- **Library Browser (ViewLib)** – обозреватель компонент библиотек, который позволяет вам только увидеть компоненты и открывает быстрый доступ к ним.

Щелкните по кнопке .

- **Component Library Editor (LibEdit)** – редактор схемных компонент, который позволяет вам реально управлять компонентами и библиотеками.

Щелкните по кнопке .

### 11.2 Обзор состава компонентов

Компонент в библиотеке – это композиция из :

- Его графического изображения (линий, окружностей, текстовых полей).
- Выводов, которые описывают электрические свойства, используемые функцией E.R.C. Они должны соответствовать обычным графическим стандартам - регулярные выводы, или выводы синхронизации, или инверсные, или с низкоуровневой активностью...
- Текстовые поля, такие как ссылки, значения, соответствующие имена посадочных мест модулей для разработки платы.


Компонент может иметь псевдонимы, то есть несколько имен (например, 7400 может иметь несколько псевдонимов, подобных 74LS00, 74HC00, 7437, поскольку все эти компоненты идентичны с точки зрения графической разработки схемы).

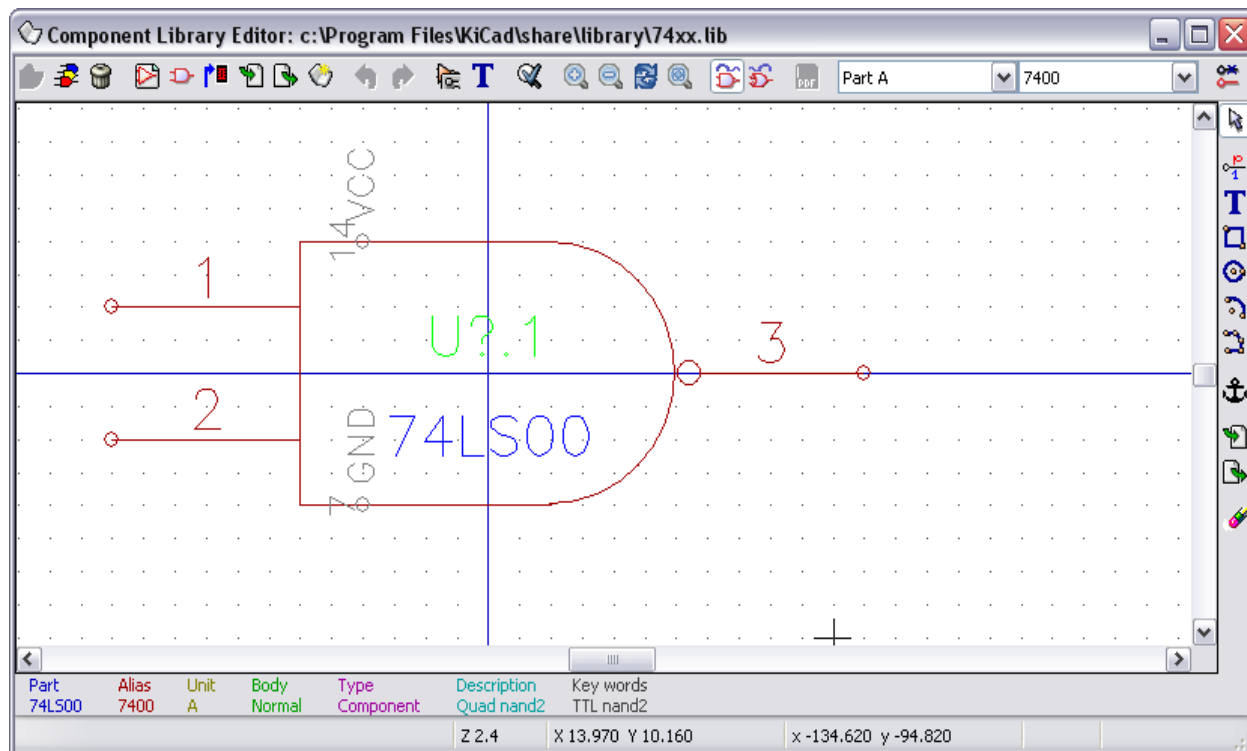
Использование псевдонимов – очень интересный метод создания полных, но компактных и относительно быстро строящихся библиотек.

### Разработка компонента это:

- Определение его основных свойств: имеет ли он много элементов, и как много; имеет ли он второе представление (известное, как Morgan, а в EESchema двойное - нормальное и конвертированное представление).
- Прорисовка образа компонента (исключая выводы) с использованием линий, прямоугольников, окружностей, многоугольников и текста.
- Добавляя выводы, внимательно задавайте их графическую проработку, имена и число выводов, их электрические свойства (вход, выход, 3 состояние, порт питания и др.).
- Добавьте псевдонимы, если другие компоненты имеют такой же вид и цоколевку (или удалите один, если компонент был создан копированием другого компонента).
- Добавьте поля, если нужно (и, возможно, имя модуля используемого в программе разработки платы) и/или определите их видимость.
- Сделайте описание компонента.
- Сохраните его в выбранной библиотеке.

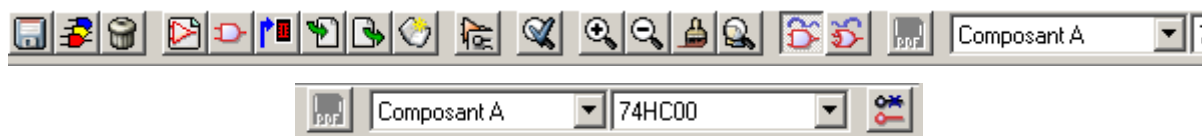
### 11.3 Загрузка компонента для редактирования













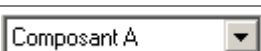
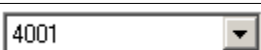

Щелкните инструмент  для открытия окна редактирования компонент и управления библиотекой в программе Component Library Editor (LibEdit). Окно программы показано ниже:






### 11.3.1 Основная инструментальная панель





	Сохранить текущую библиотеку на жестком диске
	Выбрать текущую библиотеку
	Удалить компонент в текущей библиотеке
	Создать новый компонент
	Загрузить компонент из текущей библиотеки для редактирования
	Сохранить текущий компонент в текущей библиотеке (только в RAM), файл библиотеки на диске не меняется
	Импортировать один компонент
	Экспортировать текущий компонент
	Создать новый файл библиотеки с текущим компонентом
	Редактировать свойства компонента
	Показать представление: нормальное или инвертированное (De Morgan)
	Показать ассоциированную документацию (если есть)
	Выбор элемента (для многоэлементных компонент)
	Выбор псевдонима (если текущие компоненты имеют псевдонимы)
	Редактирование выводов: независимое редактирование вида вывода и позиции (для многоэлементного и De Morgan представлений)

### 11.3.2 Выбор библиотеки для загрузки

Выбор текущей библиотеки выполняется иконкой , которая выводит список доступных библиотек для выбора. Когда компонент загружен или сохранен, он останется в этой библиотеке. Библиотечное имя компонента задается также полем VALUE.

#### **Примечание:**


Вы должны загрузить библиотеку в Eeschema, чтобы воспользоваться ею. Текущая библиотека может быть сохранена после модификации щелчком по . Компонент может быть удален из библиотеки щелчком по .

### 11.3.3 Выбор и сохранение компонента

Когда компонент редактируется, реально вы работаете не с компонентом в библиотеке, а с его копией в памяти. Значит, вы можете легко прервать любое редактирование.

Компонент можно получить из библиотеки или из старого компонента. После загрузки он будет отображен в окне программы.

#### **Выбор**

Иконка  отображает список всех доступных компонентов для выбора и загрузки желаемых компонент.

#### **Примечание 1:**


Если компонент выбран псевдонимом (alias), будет загружен основной компонент (*EESchema всегда отображает имя реально загруженного компонента*).

- Список псевдонимов всегда загружается с каждым компонентом, и может, следовательно, редактироваться.
- Когда вы хотите редактировать один **псевдоним**, этот псевдоним должен быть выбран в окне инструментальной панели :




**Первым в списке стоит корневой компонент.**


#### **Примечание 2:**


Альтернативно команда Import () позволяет вам загрузить компонент, который был прежде сохранен командой Export ().

#### **Сохранение**

После модификации компонент может быть сохранен в текущей библиотеке, или в новой библиотеке, или экспортирован в файл резервирования.


Для сохранения текущей библиотеки используйте команду Update (). Однако обновление осуществляется только в оперативной памяти (таким путем вы можете освежить в своей памяти схему).

Если кому-то хочется полностью сохранить компонент, он может использовать инструмент сохранения , который модифицирует файл библиотеки на жестком диске.

Если вы хотите создать новую библиотеку для этого компонента, используйте команду  (NewLib). Вам будет задан вопрос об имени новой библиотеки.

#### **Примечание:**






Если вы хотите иметь возможность найти ее, не забудьте добавить ее в список библиотек для поиска EESchema (смотрите конфигурацию EESchema).

И еще, можно использовать команду **Export** () для создания файла с этим компонентом (это стандартный файл библиотеки, которая содержит только один компонент).

Фактически команды **NewLib** и **Export** идентичны, первая по умолчанию создает библиотеку в папке, предопределенной для библиотек, а вторая - в папке пользователя.

### Перенос в другую библиотеку

Можно легко копировать компонент из исходной библиотеки в другую библиотеку, используя следующие команды:

- Выбрать исходную библиотеку в качестве текущей библиотеки .
- Загрузить компонент, который будте перемещен .
- Выбрать библиотеку, в которую будет перенесен компонент, как текущую библиотеку .
- Сохранить компонент в текущей библиотеке .
- Сохранить модифицированную библиотеку .


### Отмена редактирования компонента

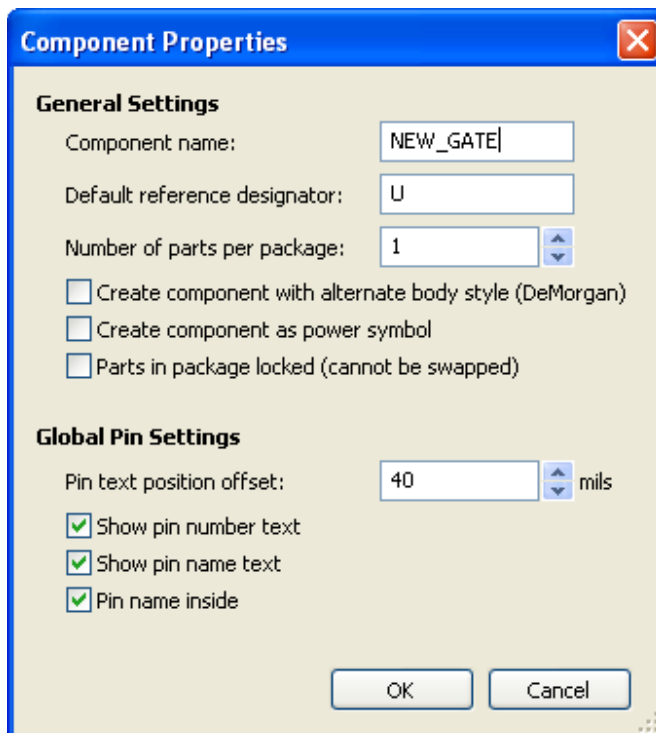
Редактируемый компонент — только рабочая копия реального компонента в библиотеке. Если он не был сохранен в RAM-памяти, вы должны перезагрузить его (или перезагрузить другую библиотеку), чтобы отменить изменения, внесенные в этот компонент.

Если вы уже сохранили его в RAM, и вы не сохранили файл библиотеки на жестком диске, можно выйти и запустить **EESchema** вновь, а затем загрузить его из библиотеки.

## 11.4 Создание компонент схемы

### 11.4.1 Создание нового компонента

Новый компонент может быть создан командой **NewPart** () . Вам будет задан вопрос об имени компонента для размещения его в библиотеке (это имя одновременно значение поля для редактора схем), об его ссылке (U, IC, R...), числе элементов в корпусе (например, стандартный компонент 7400 имеет 4 логических элемента в



**Component Properties**

**General Settings**

Component name: NEW\_GATE

Default reference designator: U

Number of parts per package: 1

☐ Create component with alternate body style (DeMorgan)

☐ Create component as power symbol

☐ Parts in package locked (cannot be swapped)

**Global Pin Settings**

Pin text position offset: 40 mils

☒ Show pin number text

☒ Show pin name text

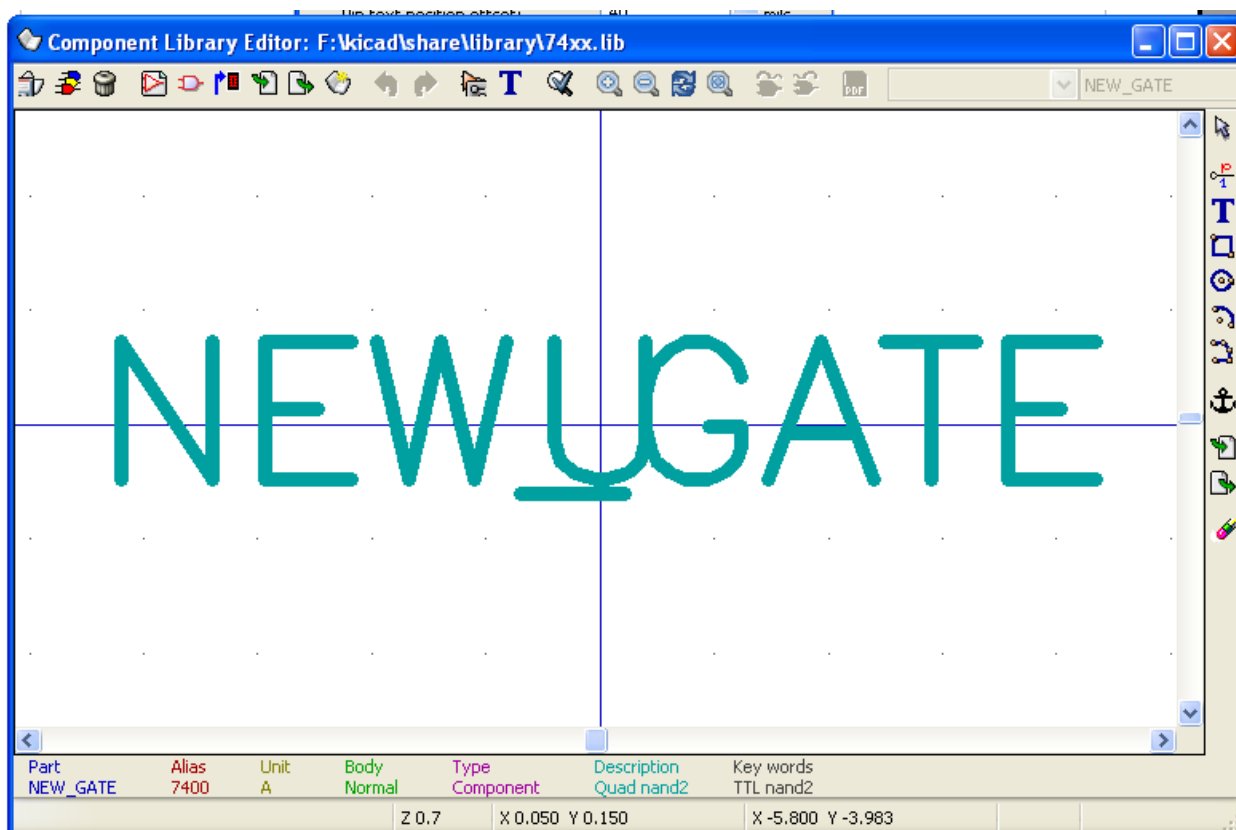
☒ Pin name inside

OK Cancel

корпусе), и существует ли инвертированное представление (*de Morgan*, как стандарт).

Если поле ссылки остается пустым, ссылка будет определена, как “U”. Все эти данные могут быть заданы позже, но предпочтительней задать их в начале разработки компонента.



Начальный вид компонента похож на приведенный ниже:

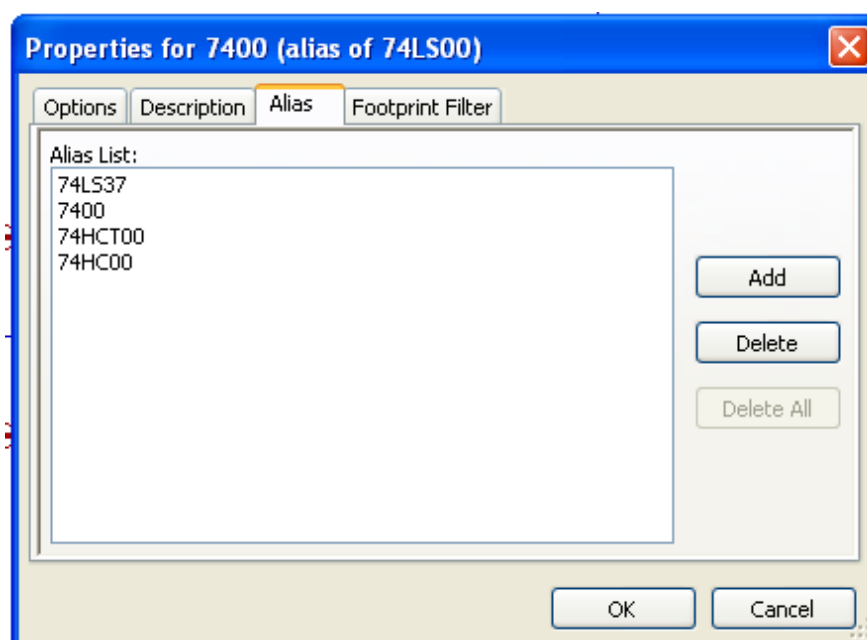


#### 11.4.2 Создание на базе другого компонента

Когда один компонент похож на другой, подчас выгоднее загрузить этот другой компонент и модифицировать его.

**Вы должны :**


1. Загрузить компонент, который будет использован в качестве модели.
2. Модифицировать его имя (редактировать командой , или щелкнуть правой клавишей мышки по имени и отредактировать текст).
3. Модифицировать список псевдонимов и удалить все нежелательные псевдонимы. Вы, возможно, будете должны удалить их все (инструмент , вкладка **Alias**, команда **Delete** или **Delete All**), поскольку любые используемые компонентом псевдонимы будут связаны впоследствии с новым компонентом.

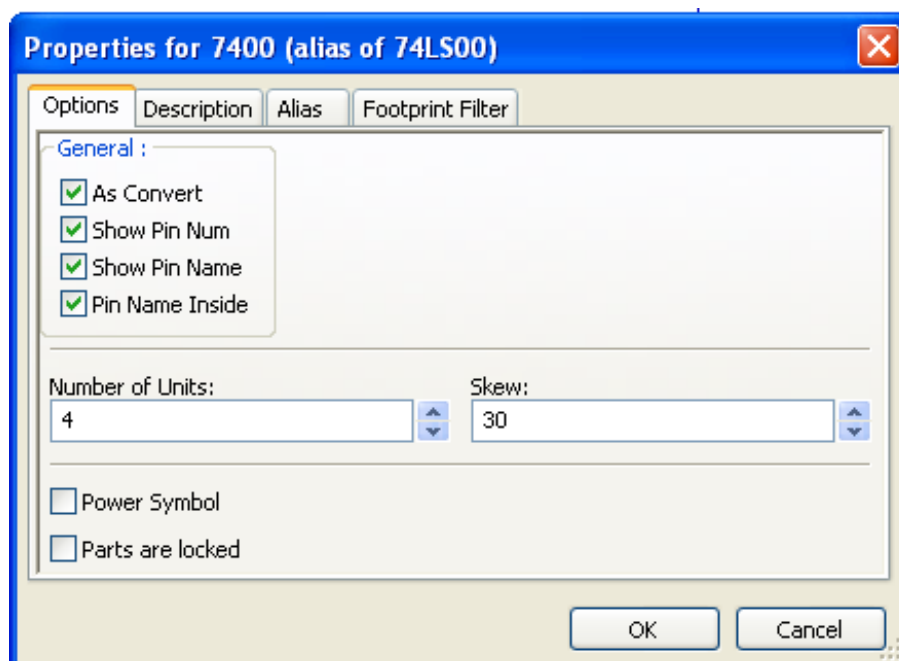


### 11.4.3 Редактирование основных характеристик компонента

Основные характеристики это:

- Число элементов в корпусе.
- Наличие ковертированного представления.
- Ассоциированная документация.
- Обновление текстовых полей.

Эти характеристики должны быть заданы правильно, поскольку они запрашиваются при создании компонента или приходят из модели аналогичного компонента. Итак, для модифиции компонента необходимо вызвать команду редактора . Появится следующее окно редактирования:



Важные опции, определяющие основные свойства:

**Number off Units**, определяющая число элементов в корпусе.

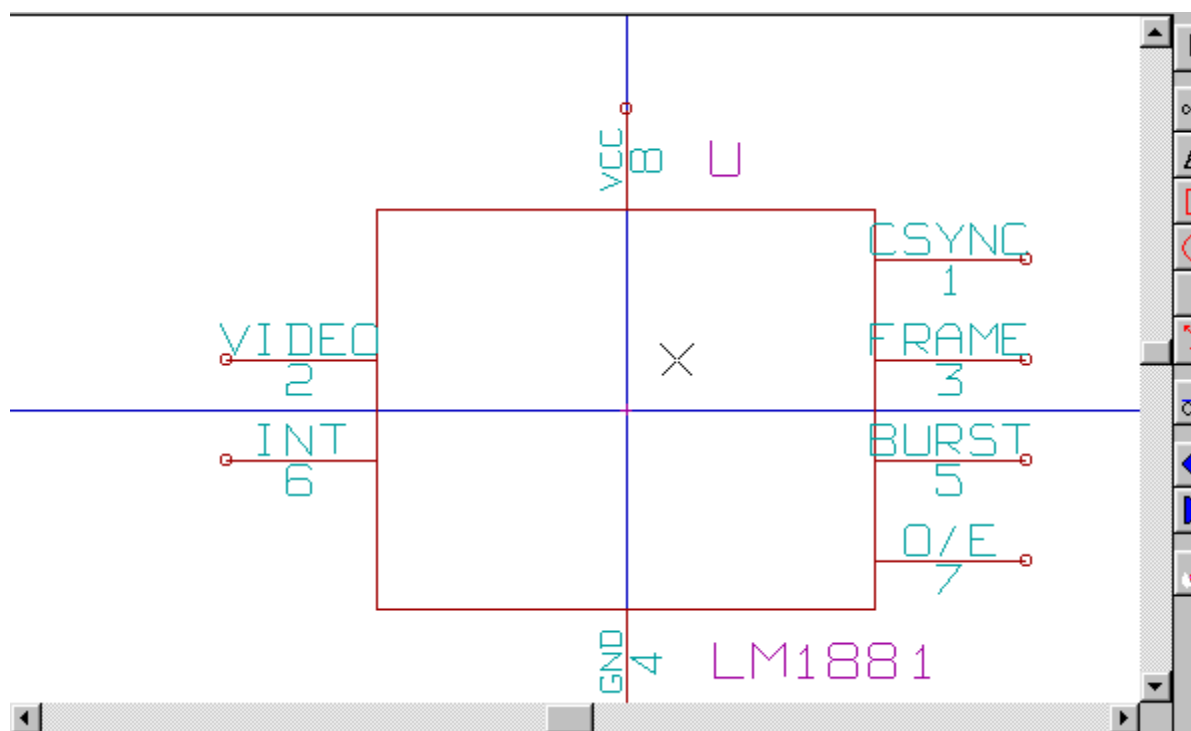
**As Convert** : устанавливается, если компонент имеет двойное представление.

Важно, чтобы эти два параметра были заданы корректно, поскольку при редактировании или создании выводов соответствующие выводы всех элементов будут печататься или создаваться вместе.

Если вы увеличиваете число элементов после создания/редактирования выводов, это увеличение вызовет дополнительную работу. Тем не менее, возможно модифицировать эти параметры в любой момент.


Графические опции **Show Pin Num** и **Show Pin Name** определяют видимость номера вывода и текста имени вывода (этот текст будет видимым, если соответствующие опции активированы).

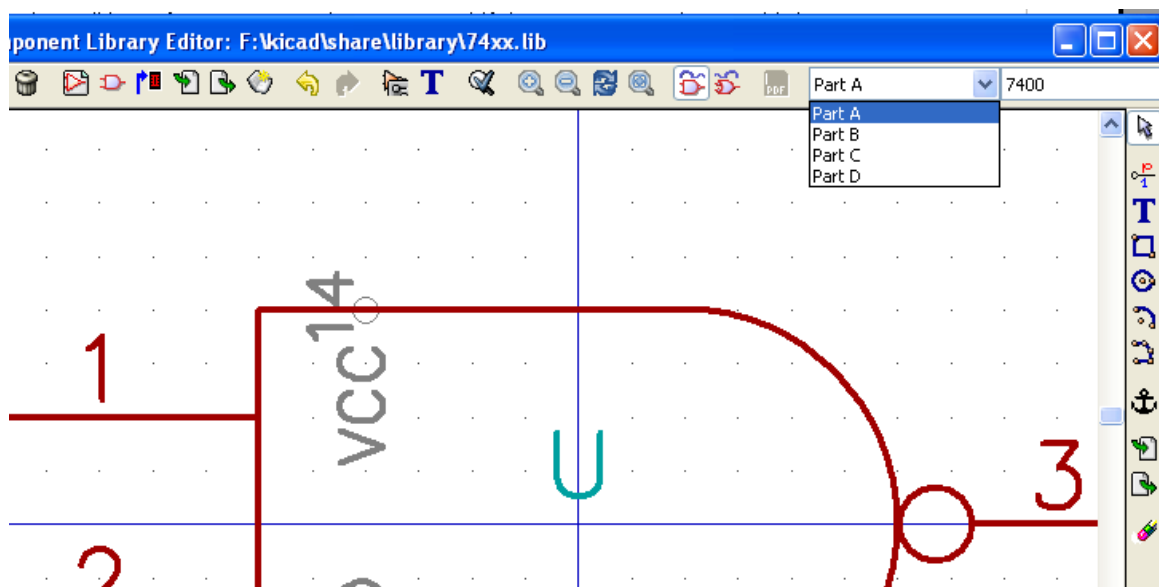
Опция **Pin Inside** определяет позицию имени вывода: этот текст будет отображен внутри контура компонента, если опция активна. В этом случае параметр **Pin Name Skew** определяет сдвиг текста внутрь. Значение от 30 до 40 (в 1/1000 дюйма) разумно. Пример ниже показывает тот же компонент со сброшенной опцией **Pin Inside** (обратите внимание на позиции имен и номеров выводов):



### 11.4.4 Многоэлементные компоненты

Во время редактирования элементов компонент, и если компоненты имеют множество элементов или представлений, вы должны будете выбрать разные элементы или представления этого компонента.

Для выбора представления щелкните по  или . Для выбора элемента



### 11.5 Разработка компонента

Вертикальная инструментальная панель позволит вам разместить все элементы компонента :



Для прорисовки компонента вы можете использовать следующие графические элементы:

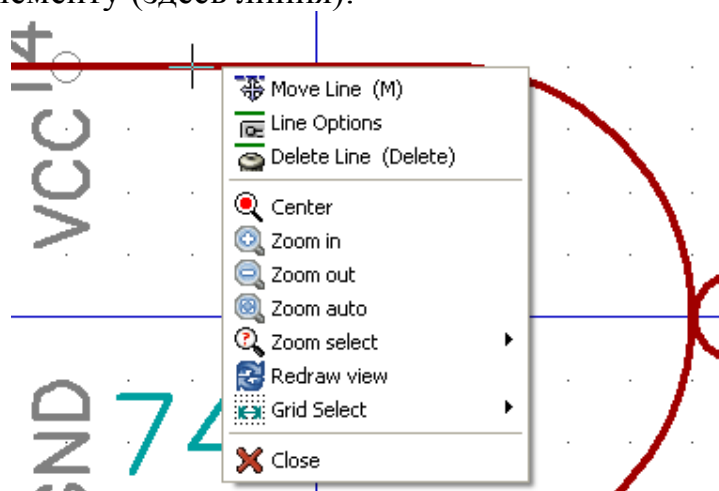
- Линии и многоугольники (простые или с заливкой)
- Прямоугольники
- Окружности
- Дуги окружности
- Тексты (иные, чем поля и тексты выводов)

Выводы и поля (значение, ссылка) стоят особняком, поскольку они не чисто графические элементы.

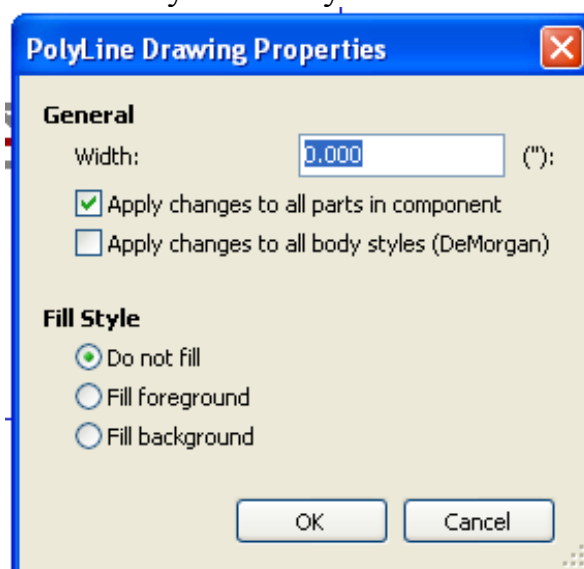
### 11.5.1 Опции членства в графических элементах

Каждый графический элемент может быть определен как обычный или специальный, либо по типу представления (нормальное или конвертированное), или по различным элементам компонента.

Меню опций доступно по щелчку правой клавишей мышки по привязанному элементу (здесь линия):



или двойным щелчком по этому элементу:



Нормальные опции графического элемента:

- **Common to Units** **установлено**, поскольку обычно разные элементы компонента имеют одинаковое графическое представление, и, таким образом, достаточно нарисовать только один элемент.
- **Common to convert** **не установлено**, поскольку двойное представление вводится для того, чтобы иметь различные графические изображения с каждым из представлений. Так что впоследствии необходимо нарисовать каждое графическое представление.



Для элементов типа “**polygon-многоугольник**” (линии проводятся последовательно) опция **Filled** позволит вам генерировать залитый многоугольник.

Однако вы можете, таким образом, обратиться к случаю (удача редка) многоэлементных компонент, разработанных с разными графическими типами через установку опции “**Parts Specific**”.

Каждый элемент тогда должен быть нарисован, и если опция “**Specific to the representations**” установлена, для каждого элемента будет необходимо нарисовать два представления.

Наконец, может быть интересно установить опцию “**Common to representations**” для отрисовки компонент в современном стандарте IEEE, так как основа графики идентична в нормальном и инверсном представлениях.

### 11.5.2 Геометрические графические элементы


Их разработка возможна, благодаря инструментам :

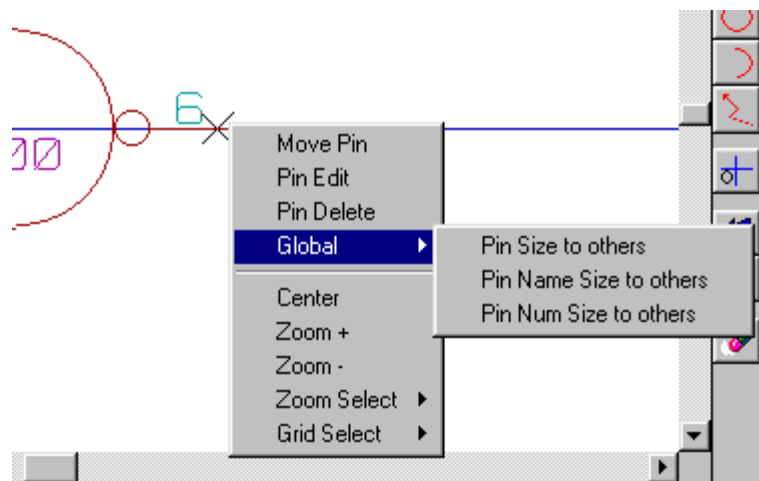
- Линии и многоугольники, простые или залитые, если опция установлена.
- Прямоугольники, определенные по диагонали.
- Окружности, определенные по центру и точке на окружности.
- Дуги, определенные по начальной и конечной точкам дуги и ее центру. Дуга проходит от 0 до 180 градусов.

### 11.5.3 Графические элементы текстового типа

**A** Позволяет создать графический текст (свободный текст). Текст всегда читаем, даже если компонент трансформируется (отражается).

## 11.6 Создание и редактирование выводов

Щелкните по  для создания вывода. Редактирование выполняется после двойного щелчка по выводу. Вы можете щелкнуть правой клавишей, чтобы открыть меню быстрого редактирования :



Выводы должны создаваться тщательно, поскольку любые ошибки будут иметь последствия при разработке платы, или вызовут проблемы при контроле схемы типа E.R.C.

Любой вывод, уже размещенный, может редактироваться, удаляться или перемещаться.

### 11.6.1 Обзор выводов

Вывод определяется по его форме (длине, графическому виду), его имени и его номеру, который всегда число (выводы гнезда PGA определяются буквами и числом, подобно A12 или AB45)

В EESchema **pin number** определяется множеством из 4 букв или цифр.

Для E.R.C. «электрический» тип выводов (вход, выход, 3 состояния...) должен также быть определен. Если этот тип не будет определен, то E.R.C. проверка схемы будет неработоспособной.

#### Примечание:

- Избегайте пробелов в именах и номерах выводов.
- Имя вывода с инверсным сигналом начинается с символа “~”.
- Если имя сокращено до этого единственного символа, вывод рассматривается как неименованный.
- Имена выводов, начинающиеся с “#”, зарезервированы для символов порта питания.
- Номер вывода состоит из букв или цифр от 1 до 4. 1,2,..9999 правильные номера, а также A1, B3... (стандартная PGA нотация) или Anod, Gnd, Wine...

### 11.6.2 Многоэлементные компоненты, двойное представление

Давайте вспомним, что, особенно для логических вентилях, символ может иметь два представления (представление, известное как “De Morgan”, а микросхемы (IC) могут включать несколько элементов, то есть несколько вентилях NOR).

Для некоторых IC вы можете предпочесть несколько разных элементов графики и выводов. Например, реле может быть представлено с различными элементами:

- катушка
- переключающий контакт 1
- переключающий контакт 2

Управление многоэлементными микросхемами и компонентами с двойным представлением гибко. Действительно, вывод может быть:

- Общим или специфическим для разных элементов.
- Подходящим для обоих представлений или специфическим для каждого.

По умолчанию, выводы специфичны для каждого представления каждого элемента, поскольку их номера различны для каждого элемента, а их вид различен для каждого представления.

Когда вывод общий, вам остается нарисовать его только раз (то есть, в случае выводов питания).

Это также случай вида, который почти всегда идентичен для каждого элемента (но есть различия между нормальным и конвертированным представлением).

### 11.6.3 Выводы: базовые опции

Компоненты с множеством элементов и/или представлений озадачивают специфической проблемой при создании выводов и редактировании.

Так как множество выводов специфично для каждого элемента (поскольку нумерация их выводов особенна для каждого элемента) и для каждого представления (поскольку их форма своеобразна для каждого представления), создание и редактирование выводов, похоже, по этой причине становится затяжным и утомительным.

Фактически EEschema позволяет одновременную поддержку выводов :



По умолчанию для каждого многоэлементного компонента и/или двойного представления эти модификации осуществляются для всех выводов, относящихся к элементам и представлениям, когда вы создаете, редактируете (исключая форму и номер), удаляете или перемещаете вывод (то есть, для всех выводов, размещенных в тех же координатах).

- При разработке модификации, выполняемые для текущего представления, осуществляются для всех элементов.
- Номера выводов модифицируемые для текущего элемента, модифицируются для 2 представлений.
- Имена модифицируются независимо.

Эта зависимость была установлена, чтобы позволить быструю модификацию в большинстве случаев.

Эта зависимость при модификации может быть отменена в меню опций (**Options Menu**), позволяя вам создавать компоненты с элементами и представлениями с полностью независимыми характеристиками.

Эта опция зависимости управляется инструментом :

- Если  не активна (не подсвечена), редактирование будет обращаться ко всем элементам и ко всем представлениям. *Это нормальная опция.*
- Если  активна (подсвечена), редактирование будет обращаться только к текущему элементу и в текущем представлении (то есть, к тому, что вы видите на экране). *Эта опция редко используется.*

### 11.6.4 Выводы: определение характеристик

Окно свойств вывода позволяет вам редактировать характеристики вывода.

**Pin Properties**

Name:  Name text size:  inches

Number:  Number text size:  inches

Orientation:  Length:  inches

Electrical type:

Style:

☐ Add to all parts in package

☐ Add to all alternate body styles (DeMorgan)

☒ Visible

OK Cancel

Это меню автоматически выпадает, когда вы создаете вывод, или когда дважды щелкните по существующему выводу.

Это позволяет вам определить или модифицировать:

- Имя и размер имени вывода.
- Номер и размер номера вывода.
- Длину вывода.
- Электрический тип и вид.
- Членство (общий для нормального и "Morgan" представлений или нет).
- Невидимый вывод (используется для выводов питания).

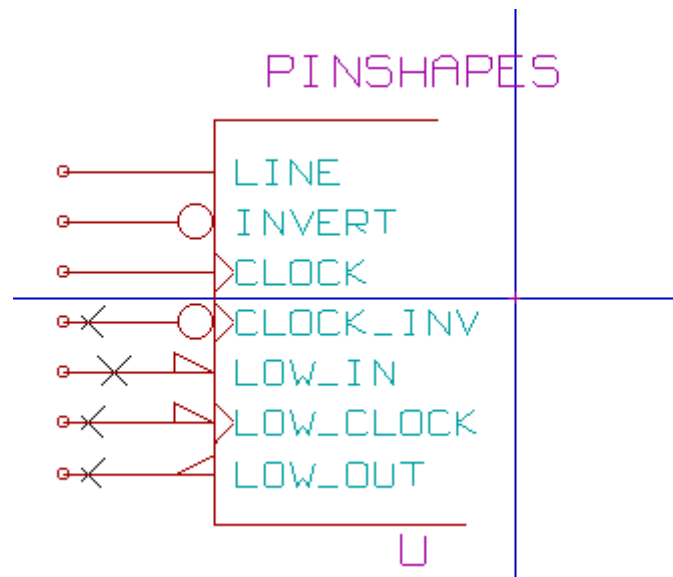
### **Напомним:**

- Имя вывода начинается с “~” для **инвертированных** сигналов.
- Если имя сокращено только до этого символа, вывод считается неименованным.

Номер вывода состоит из от 1 до 4 символов (букв или цифр). - 1,2..9999 это правильные номера, но так же A1, B3... (стандартная нотация PGA) или Anod, Gnd, Wine...

### **11.6.5 Очертания выводов**

Вы можете видеть на рисунке ниже различные очертания выводов. Выбор формы имеет чисто графическое значение и не имеет влияния на E.R.C. или netlist функции.



### 11.6.6 Выводы : электрические разновидности

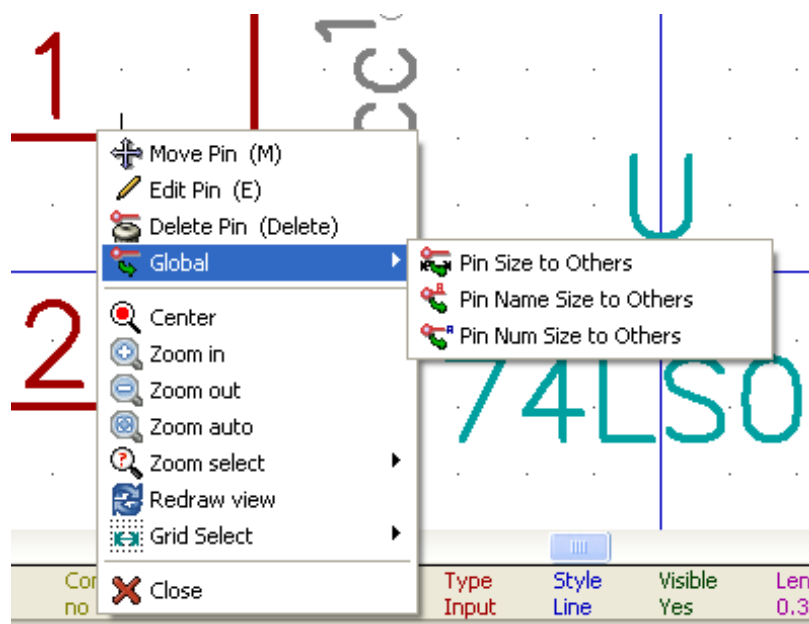
Выбор типа важен для функции E.R.C. Выбор банален для **input** и **output** выводов микросхем (IC).

- **BiDi** тип указывает двунаправленные выводы, коммутируемые между входом и выходом (микропроцессорная шина данных, например).
- Тип **3 States** – это обычный выход с тремя состояниями.
- **Passive** тип используется для выводов пассивных компонент – резисторы, разъемы....
- **Unspec** тип (неспецифицированный) может использоваться, когда проверка E.R.C. безразлична.
- **Power In** тип должен использоваться для выводов питания компонент.
- **Power Out** для выходов регуляторов. В особенности, если вывод это порт питания ( **Power In** или **Out**, и объявленные как невидимые "Invisible", не отображаются на чертеже схемы, и они автоматически соединяются с другими того же типа и тем же именем ( **Invisible Power Pin**)).
- Вы также можете использовать **Open Emitter** и **Open Collector** типы.

### 11.6.7 Выводы : глобальные модификации

Можно модифицировать длину всех выводов, или размер текста (имя, номер элемента), используя команду **Global** из выпадающего меню, чтобы установить один из этих трех параметров.


Щелкните по параметру, который хотите модифицировать, и введите новое значение, которое будет затем относиться ко всем выводам компонента в текущем представлении.



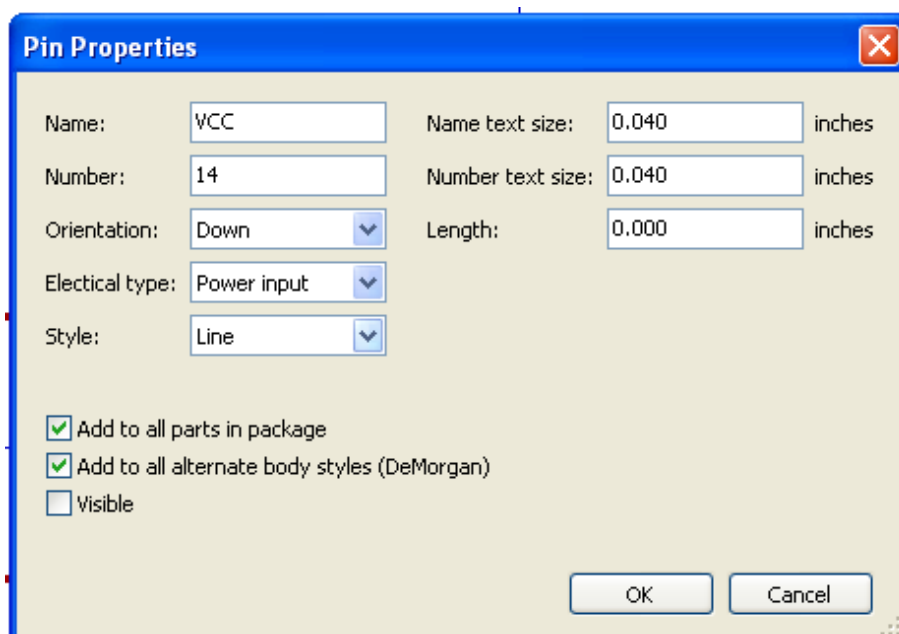
### 11.6.8 Выводы : многоэлементные компоненты и двойное представление

Разные элементы представления (такие, как встречаются у 7400, 7402...) могут нуждаться в дополнительном редактировании.

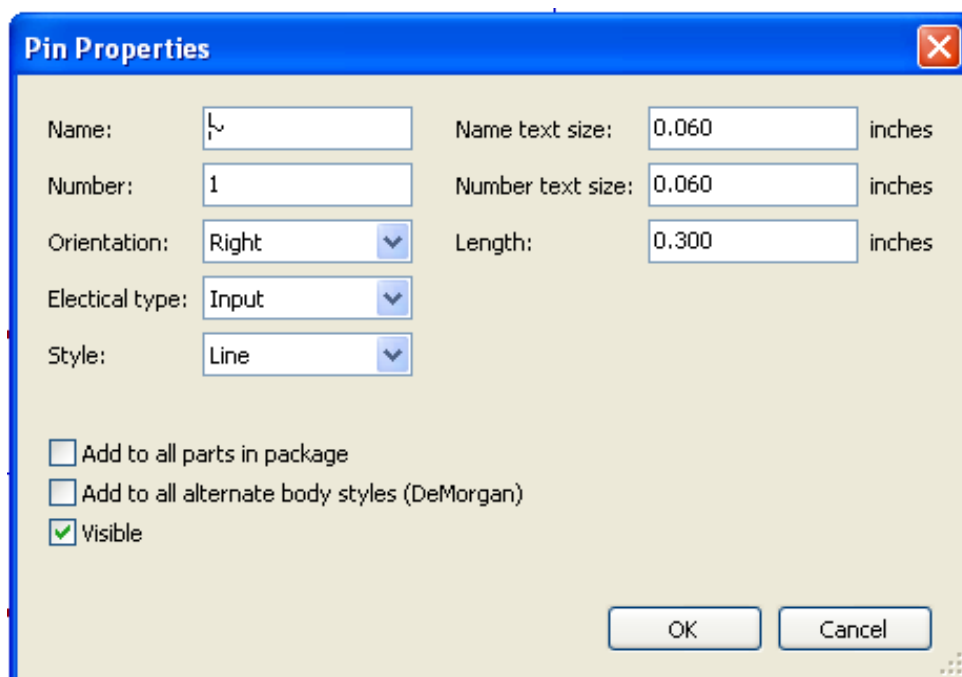
Эта дополнительная работа будет минимальна, если приняты следующие меры предосторожности:

- Генеральная опция **Edit pin part per part**  должна оставаться сброшенной.
- Выводы питания будут создаваться с активизированными атрибутами **CommonUnit** и **CommonConvert** (они могут быть также невидимы - **No Draw**)).

Корректные установки похожи на эти:



Когда другие выводы были созданы, они созданы для каждого элемента и каждого представления.



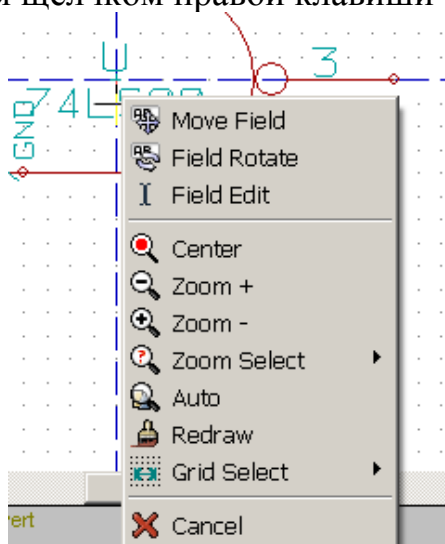
Например, вывод выхода элемента А компонента 7400 был бы создан EESchema в 8 образцах: 2 на каждый элемент (есть 4 элемента А, В, С, D и для каждого элемента нормальное представление и инвертированное представление, известное как Morgan – представление Моргана).

Однако сначала получим, возможно, корректно созданный элемент «А» в его нормальном представлении. Таким образом, необходимо для каждого элемента :

- Выбрать инвертированное представление и отредактировать форму и длину каждого вывода.
- Для других элементов отредактировать номера выводов.

## 11.7 Редактирование полей

Для уже существующих полей вы можете использовать команды быстрого редактирования щелчком правой клавиши :



Для более полного редактирования или заполнения пустых полей необходимо вызвать окно редактирования свойств компонент.

Поля – это тексты, ассоциированные с компонентом. Не следует путать их с текстами, принадлежащими графическому представлению этого компонента.

**На этих полях :**

- Значение
- Ссылка
- Поля от 1 до 8 (для комментариев)
- Имя ассоциированного модуля (имя посадочного места для платы)
- Имя ассоциированной схемы (для компонент, которые отображаются в некоторой иерархии).

**Fields Properties**

Fields

Référence	U	
Valeur	74LS00	
Module		
Documentation		
Field1		
Field2		
Field3		
Field4		
Field5		
Field6		
Field7		
Field8		

Add Field

Delete Field

Move Up

Options

Horiz. Justify

☐ Align left

☒ Align center

☐ Align right

Vert. Justify

☐ Align bottom

☒ Align center

☐ Align top

Visibility

☒ Show

☐ Rotate

Style:

☒ Normal

☐ Italic

☐ Bold

☐ Bold Italic

Field Name

Référence

Field Value

U

Size ("):

0.060

Pos X ("):

0.000

Pos Y ("):

-0.050

OK Cancel

Поля значения и ссылки определяются в процессе создания компонента, и могут здесь модифицироваться.

Возможно, что это может быть полезно для редактирования поля **Name** ассоциированного модуля для прямой генерации netlists (для программы PCBNEW), включая имя (посадочное место) модуля.



Поле Name ассоциированной схемы для специального использования в других программах электронных САПР.

Поля от 1 до 8 не имеют оснований для использования в библиотеке, поскольку они скорее могут использоваться при разработке схемы.

Для библиотеки редактирование полей **Value** и **Reference** позволяет определить их размер и позицию.

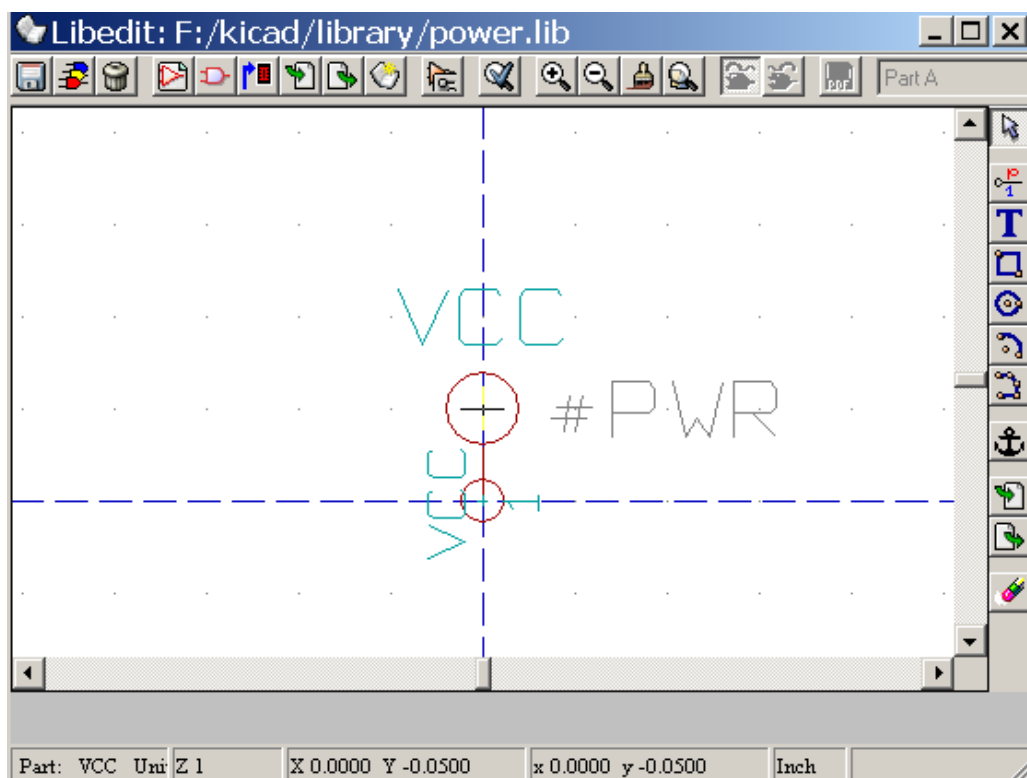
### **Важные замечания :**

- **Модификация текста поля value эквивалентно созданию нового компонента**, начинающегося со старого, используемого в качестве модели, поскольку этот новый компонент имеет имя, содержащееся в **поле значения**, когда вы сохраняете его в библиотеке.
- Для редактирования невидимого поля (то есть пустого, поскольку даже если поле имеет невидимый атрибут, оно отображается в LibEdit) вы должны будете использовать окно основного редактирования (выше).

## **11.8 Создание символов порта питания**

Символы порта питания создаются точно так же, как обычные компоненты. Полезно собрать их в специальную библиотеку, такую как Power.lib. Они содержат графический символ (желаемой формы) и входной вывод типа **Power Invisible**.

Они будут поддерживаться, как и обычные компоненты, программами ввода схем. Однако необходимы некоторые предосторожности. Вот символ **power + 5V** :



**Символ выводится следующим образом :**

- Вывод **Invisible Power**, названный **+ 5V** (важно, поскольку это имя будет присвоено цепи **+ 5V**), номер вывода 1 (номер не важен) и длина нулевая.
- Форма типа **Line**, явный тип **Power** и атрибут **Invisible** (невидимый).
- Графика: небольшой кружок и сегмент от вывода к кружку.
- Якорь (anchor) символа на выводе.
- Значение **+ 5V** похоже на имя вывода для отображения значения этого символа (вывод невидимый по умолчанию, его имя не появляется).
- Ссылка **# + 5V** (так и отображается **# + 5V**) подобна имени вывода. Текст ссылки не имеет значения, исключая первую литеру, которая должна быть **#**. По соглашению каждый компонент, чья ссылка начинается с этого символа, не будет появляться ни в списке компонент, ни в спецификациях (netlists). Более того, в опциях символа ссылка объявлена как невидимая.

Создание нового символа порта питания достаточно просто и быстро, если вы используете другой символ в качестве модели. Вы только должны :

- Загрузить модель.
- Отредактировать **имя вывода** (который затем примет имя нового порта питания).
- Отредактировать поле **Value** (то же имя, что и у вывода, если вы хотите, чтобы значение порта питания отображалось...).
- Сохранить новый компонент.

## 12. Дополнительные функции редактора компонент

### 12.1 Обзор функций

Компонент состоит из нескольких элементов :

- его графического представления (геометрические очертания, тексты);
- выводов;
- полей для ассоциированных текстов, используемых постпроцессорами при выпуске списков цепей и компонентов.

Два поля подлежат заполнению: ссылка и значение. Имя разработки ассоциировано с компонентом и именем посадочного места модуля (footprint). Другие поля – свободные, зачастую остающиеся пустыми, они могут заполняться в процессе ввода схемы.

Имеющаяся возможность манипуляции документацией, ассоциированной со всяким компонентом библиотеки, облегчает изучение, использование и поддержку библиотек.

Ассоциированная документация состоит из :

- строки комментария.

- строки ключевых слов, таких как TTL CMOS NAND2..., разделенных пробелами.

- прикрепленного имени файла (например, примечание к приложению, описание в файле типа **pdf**).

Директория по умолчанию для прикрепленных файлов документов – это *kicad/share/library/doc*. Если не так, то дополнительно *kicad/library/doc*.

При работе в Linux также просматриваются каталоги:

*/usr/local/kicad/share/library/doc*

*/usr/share/kicad/library/doc*


*/usr/local/share/kicad/library/doc*

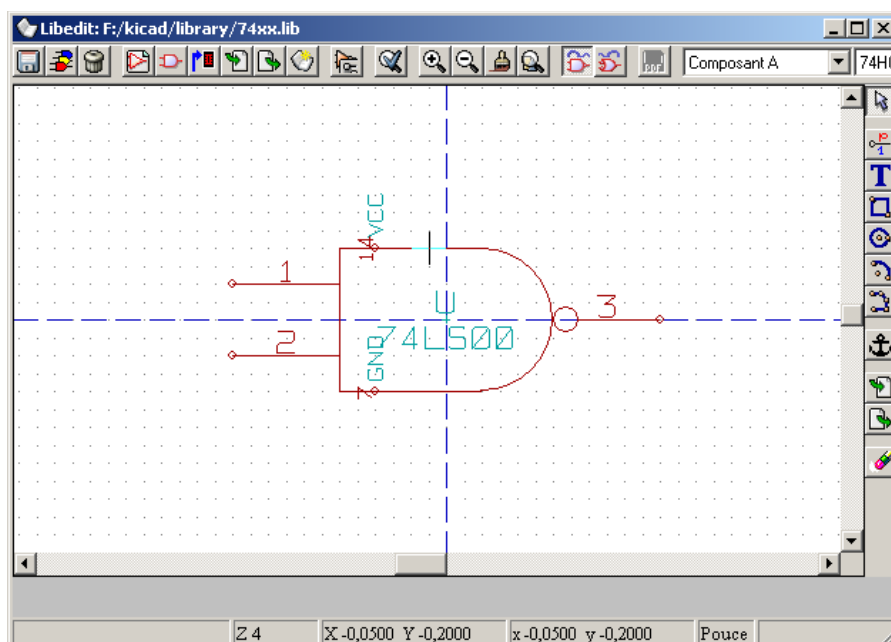
Ключевые слова позволяют выборочно искать компонент по различным критериям отбора. Комментарий и ключевые слова отображаются в различных меню, и особенно, когда вы выбираете компонент в библиотеке.

Компонент также имеет точку привязки (или точку якоря - anchoring point). Поворот или отражение компонента происходят относительно этой точки, а при размещении эта точка используется, как позиция ссылки. Так что рекомендуется выбирать позицию якоря аккуратно.

И, наконец, компоненты распространяются в библиотеках, классифицированных по теме или производителю, с целью облегчения управления ими.

## 12.2 Позиционирование якоря


Якорь (anchor) компонента – точка привязки компонента с координатами (0,0), обозначенная на экране синими осями. Якорь может быть перемещен следующим образом: выберите инструмент , а затем щелкните по новой выбранной для якоря позиции. Чертеж будет автоматически перецентрирован к этой новой точке закрепления.



## 12.3 Псевдонимы компонентов (алиасы)

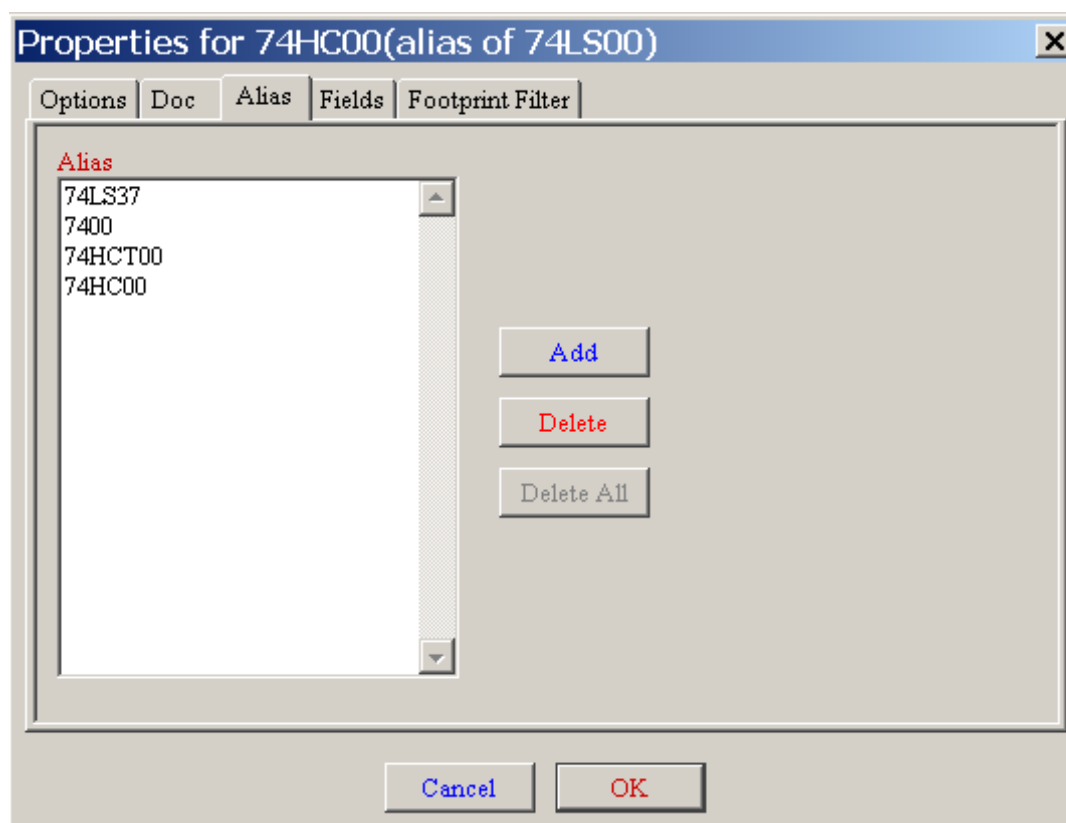
Псевдоним (Alias) – это иное (альтернативное) имя компонента в библиотеке. Компоненты со сходной цоколевкой и представлением, вследствие этого, могут быть представлены только одним компонентом, имеющим несколько псевдонимов (7400 с псевдонимами 74LS00, 74HC00, 74LS37).

Использование псевдонимов позволяет построить полную библиотеку гораздо быстрее. Вдобавок, такие библиотеки, будучи много компактнее, загружаются значительно быстрее.

Для модификации списка псевдонимов вы должны выбрать основное окно редактирования (инструмент ) и выбрать закладку **Alias**:

Теперь вы можете добавить или удалить псевдоним. Текущий псевдоним не может быть явно удален, пока он редактируется.

Для удаления всех псевдонимов вы должны вначале выбрать корневой компонент (первый компонент в списке псевдонимов в окне выбора основной инструментальной панели).

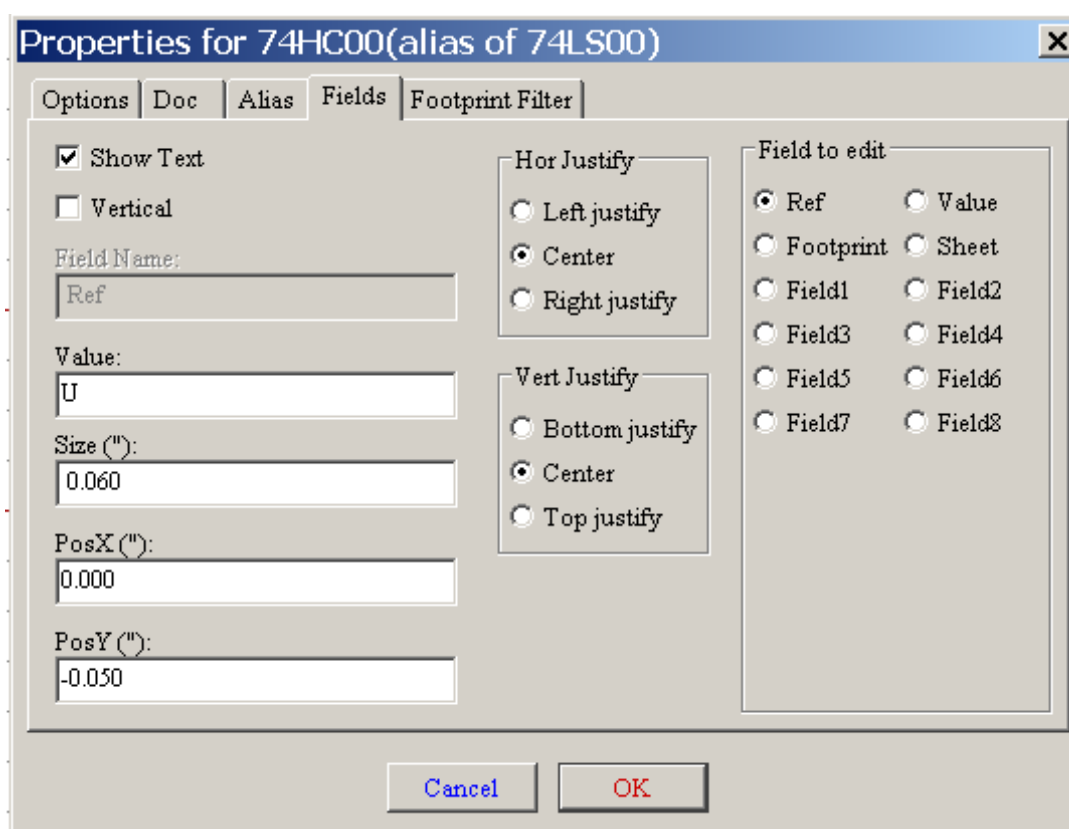


## 12.4 Текстовые поля компонента


С компонентов можно связать 4 специальных тестовых поля и до восьми пользовательских полей.

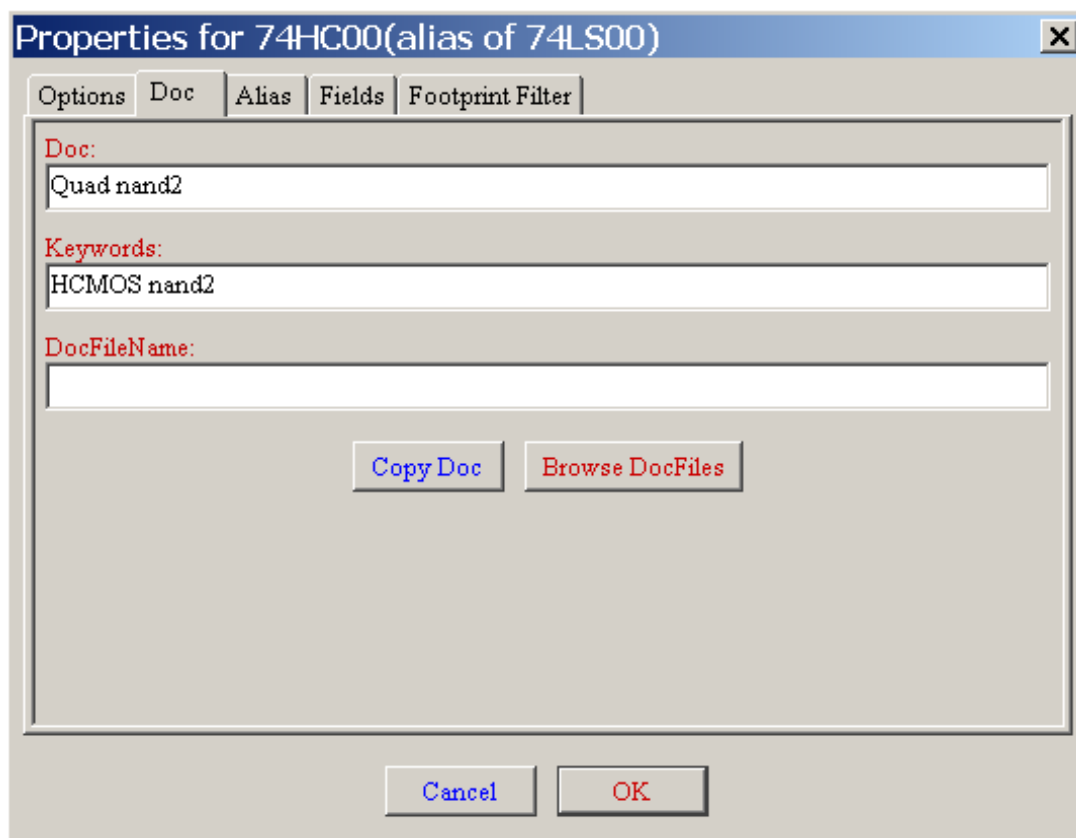
*Специальными полями являются:*

- Ссылка (Ref) – позиционное обозначение
- Значение (Value) – имя компонента в библиотеке, оно же значение поля в схеме по умолчанию
- Посадочное место (Footprint) – имя посадочного места модуля, полезно для программы CVPCB, обеспечивающей переход к проекту печатной платы
- Лист (Sheet) – зарезервировано.



## 12.5 Документация на компонент

Для редактирования информации о документировании компонента необходимо вызвать основное окно редактирования компонент (инструмент ) и выбрать закладку **Doc**:



### ***Предостережение:***

Удостоверьтесь, что выбран правильный псевдоним или коренной компонент, поскольку эта документация специфична, различна для псевдонимов.

Клавиша **Copy Doc** позволит вам скопировать документированную информацию корневого компонента прямо в текущий редактируемый псевдоним.

### **12.5.1 Применение ключевых слов**

Ключевые слова позволяют отыскивать избирательным путем компоненты по определенным критериям отбора (функция, технология).

Средство поиска EESchema не чувствительно к регистру. Наиболее ходовые ключевые слова, используемые в библиотеках, это :

- CMOS TTL для логических семейств
- AND2 NOR3 XOR2 INV... для вентилях (AND2 = 2х-входовый вентиль И, NOR3 = 3х-входовый вентиль НЕ-ИЛИ)
- JKFF DFF... для JK или D-триггера
- ADC, DAC, MUX...
- OpenCol для вентиля с выходом, имеющим открытый коллектор.

В результате, если в программе ввода схемы Eeschema вы ищете компонент: ( обратите внимание на "=" в начале команды) **=NAND2 OpenCol**, то EESchema отобразит список компонент, имеющих эти два ключевых слова.

### 12.5.2 Доступ к информации о компонентах

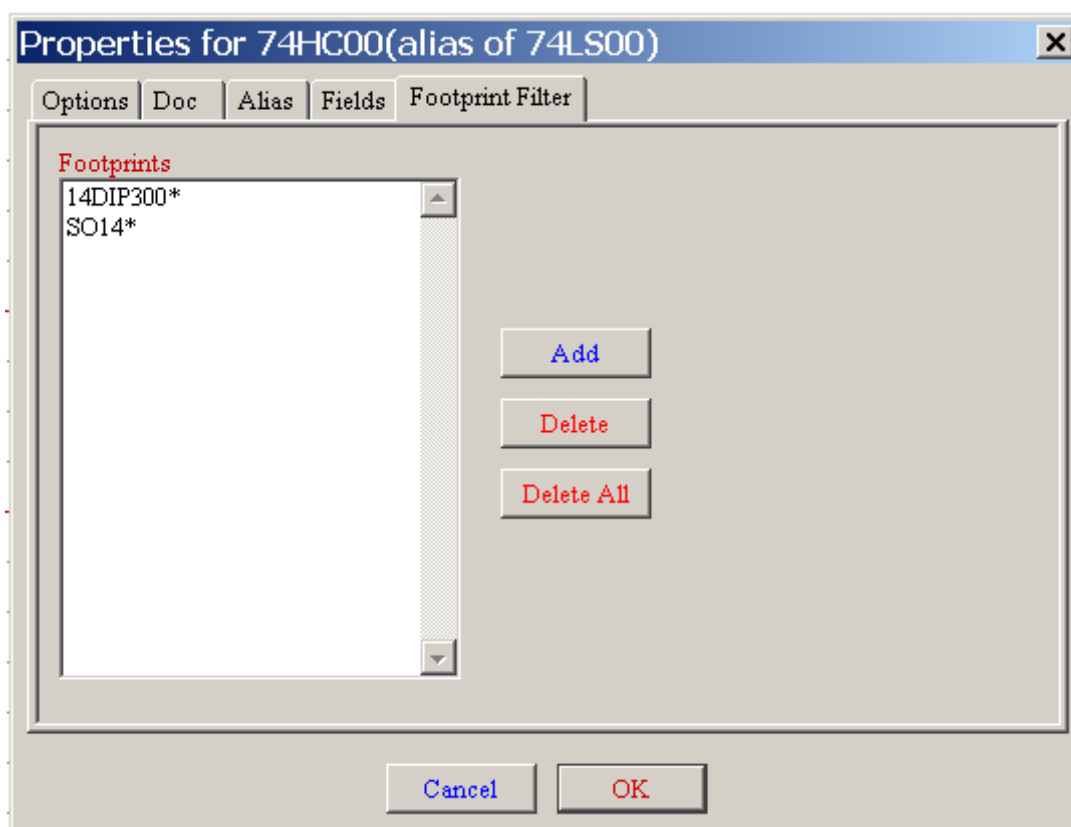
Строка описания компонента и ключевые слова поиска помогают при его выборе из списка компонент библиотеки и в меню Library Browser.

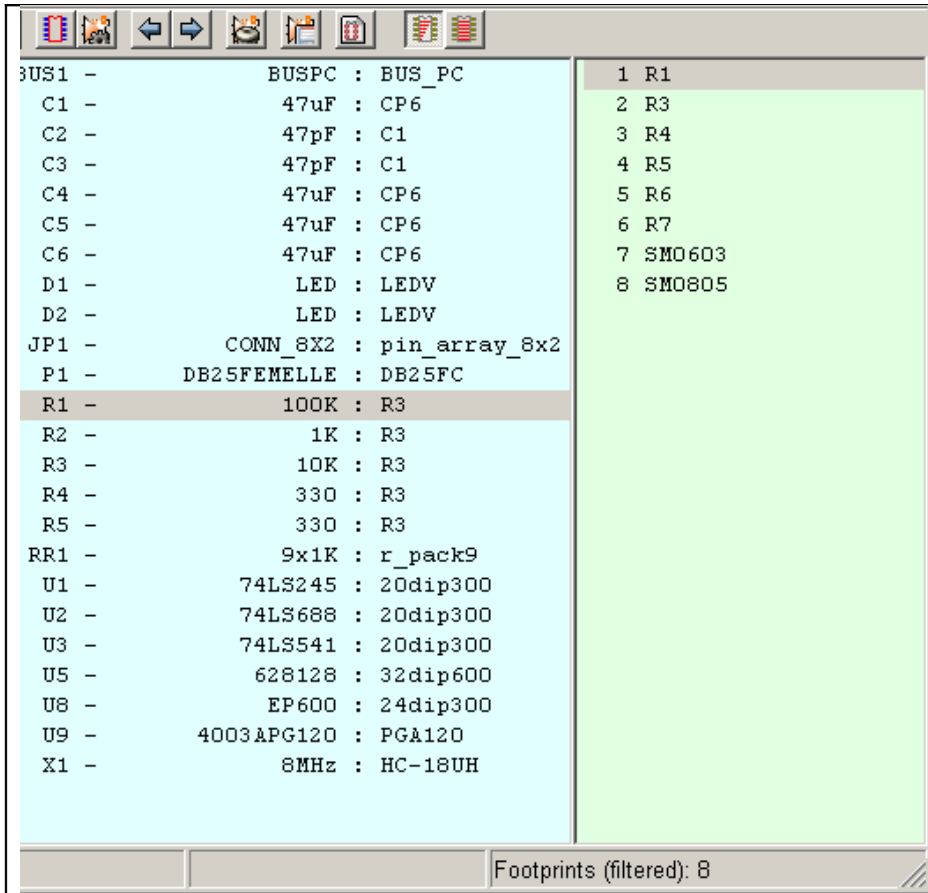
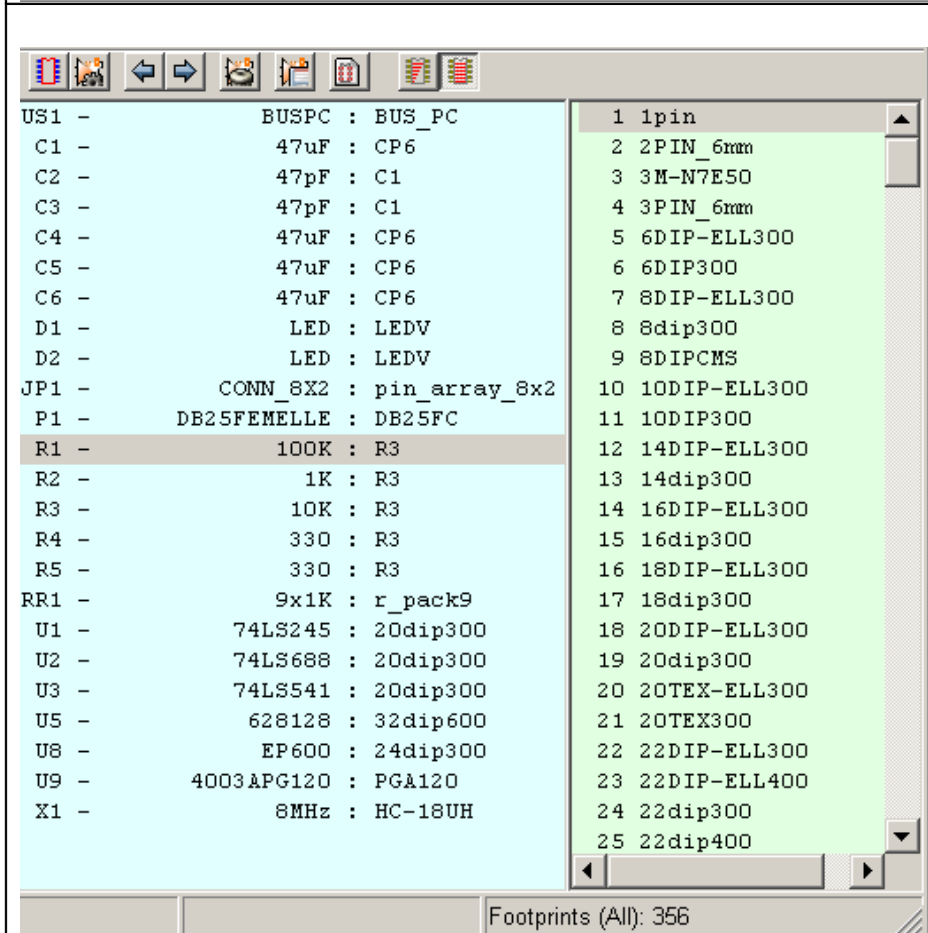
Если doc-файл со спецификацией компонента существует, он также доступен в программе ввода схемы через выпадающее меню после двойного щелчка по компоненту. При этом выводится прикрепленный файл (pdf-документация, схема приложения или чертеж).

### 12.5.3 Фильтр для назначения посадочных мест (модулей) для CVPCB

На следующей закладке можно выставить фильтр на имена для ограничения модулей, претендующих на посадочные места на плате, при их назначении в программе CVPCB.

**SO14\*** разрешает все имена, начинающиеся с SO14. А, например, **R?** указывает на имена модулей резисторов из двух символов, начинающиеся на R.



 <p>3US1 - BUSPC : BUS_PC</p> <p>C1 - 47uF : CP6</p> <p>C2 - 47pF : C1</p> <p>C3 - 47pF : C1</p> <p>C4 - 47uF : CP6</p> <p>C5 - 47uF : CP6</p> <p>C6 - 47uF : CP6</p> <p>D1 - LED : LEDV</p> <p>D2 - LED : LEDV</p> <p>JP1 - CONN_8X2 : pin_array_8x2</p> <p>P1 - DB25FEMALE : DB25FC</p> <p>R1 - 100K : R3</p> <p>R2 - 1K : R3</p> <p>R3 - 10K : R3</p> <p>R4 - 330 : R3</p> <p>R5 - 330 : R3</p> <p>RR1 - 9x1K : r_pack9</p> <p>U1 - 74LS245 : 20dip300</p> <p>U2 - 74LS688 : 20dip300</p> <p>U3 - 74LS541 : 20dip300</p> <p>U5 - 628128 : 32dip600</p> <p>U8 - EP600 : 24dip300</p> <p>U9 - 4003APG120 : PGA120</p> <p>X1 - 8MHz : HC-18UH</p> <p>Footprints (filtered): 8</p>	С фильтром
 <p>US1 - BUSPC : BUS_PC</p> <p>C1 - 47uF : CP6</p> <p>C2 - 47pF : C1</p> <p>C3 - 47pF : C1</p> <p>C4 - 47uF : CP6</p> <p>C5 - 47uF : CP6</p> <p>C6 - 47uF : CP6</p> <p>D1 - LED : LEDV</p> <p>D2 - LED : LEDV</p> <p>JP1 - CONN_8X2 : pin_array_8x2</p> <p>P1 - DB25FEMALE : DB25FC</p> <p>R1 - 100K : R3</p> <p>R2 - 1K : R3</p> <p>R3 - 10K : R3</p> <p>R4 - 330 : R3</p> <p>R5 - 330 : R3</p> <p>RR1 - 9x1K : r_pack9</p> <p>U1 - 74LS245 : 20dip300</p> <p>U2 - 74LS688 : 20dip300</p> <p>U3 - 74LS541 : 20dip300</p> <p>U5 - 628128 : 32dip600</p> <p>U8 - EP600 : 24dip300</p> <p>U9 - 4003APG120 : PGA120</p> <p>X1 - 8MHz : HC-18UH</p> <p>Footprints (All): 356</p>	Без фильтра




## 12.6 Файлы символов компонент


Вы можете собрать наборы файлов графических символов, содержащие часто используемые символы компонент. Такие символы могут облегчить создание будущих компонент (треугольник, изображающий вентили И, ИЛИ, исключающее ИЛИ...).

Эти файлы запоминаются по умолчанию в директории библиотек, и имеют расширение .sym. Символы не собираются в библиотеки, подобно компонентам, поскольку они, как правило, не столь многочисленны.

### 12.6.1 Создание символа экспортом


Компонент может экспортироваться для получения символа инструментом . Вы обычно заимствуете при этом только одно изображение, так что неплохо бы удалить все выводы, если они есть.

### 12.6.2 Импорт символа

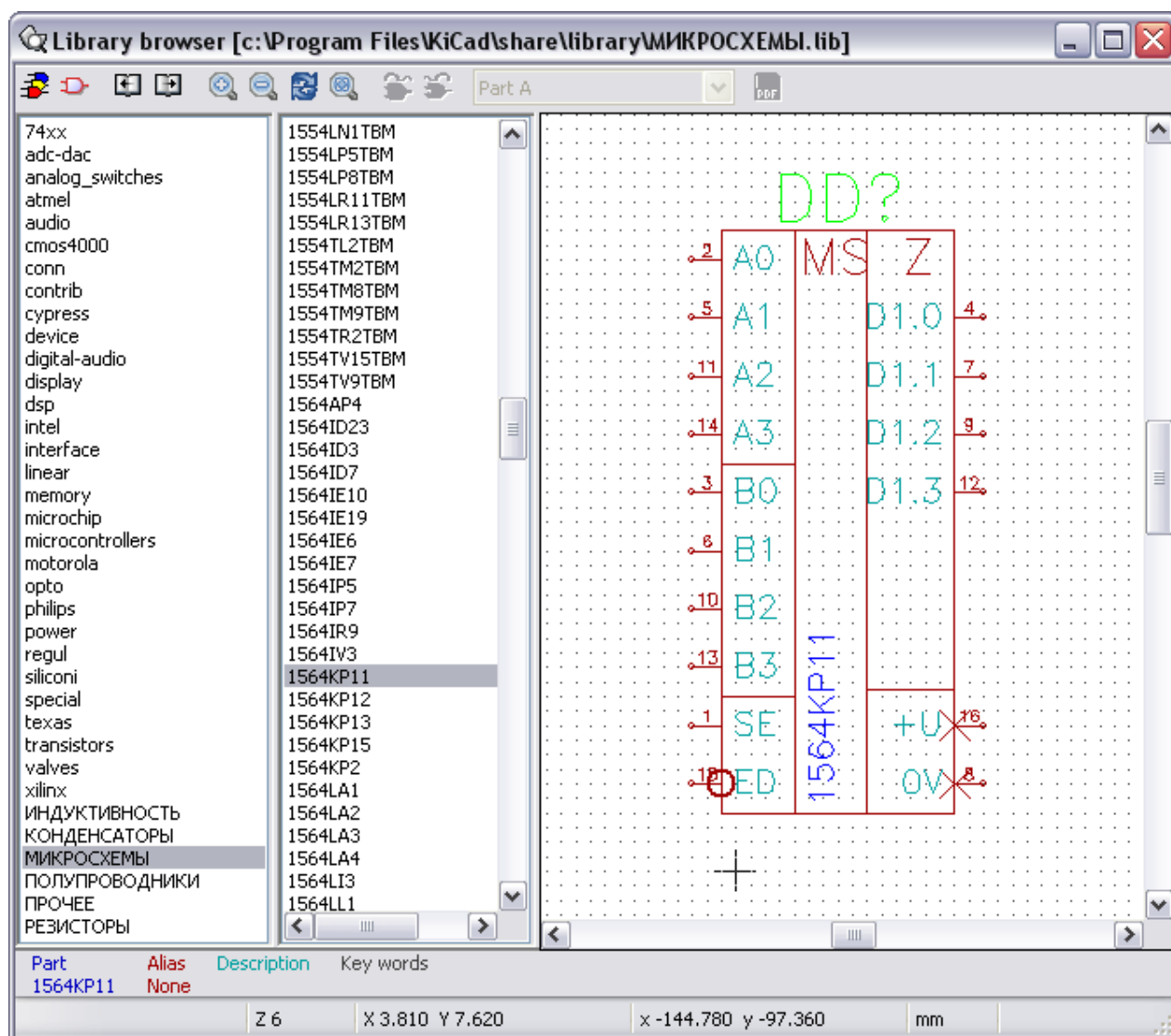
Операция позволяет вам добавить изображение к компоненту, который вы редактируете. Символ импортируется инструментом . Импортированное изображение добавляется таким же, каким оно было создано в уже существующих изображениях.

## 13. Визуализация библиотек

### 13.1 Основные функции Library Browser

**Library browser** позволяет провести быструю визуальную проверку содержания библиотек системы KiCAD. Визуализатор вызывается с помощью инструментальной кнопки .

### 13.2 Основное окно Library Browser











Для проверки библиотеки вы должны выбрать ее в списке, отображаемом слева. Содержание библиотеки появится во втором списке, который позволит вам выбрать компонент для просмотра.

На инструментальной панели также есть кнопки для выбора библиотеки и компонента, для перехода к следующему или предыдущему изображению, управления его масштабом.

### 13.3 Инструментальная панель Library Browser



Команды :

	Выбор желательной библиотеки (либо выбирается в отображенном списке)
	Выбор компонента (либо выбирается в отображенном списке).
	Отобразить предыдущий компонент.
	Отобразить следующий компонент.
	Управление масштабом отображения.
	Выбор представления (нормальное или конвертированное), если есть.
	Выбор элемента (если компонент многоэлементный).
	Отображение ассоциированных документов (при наличии).

## 14. Ссылки

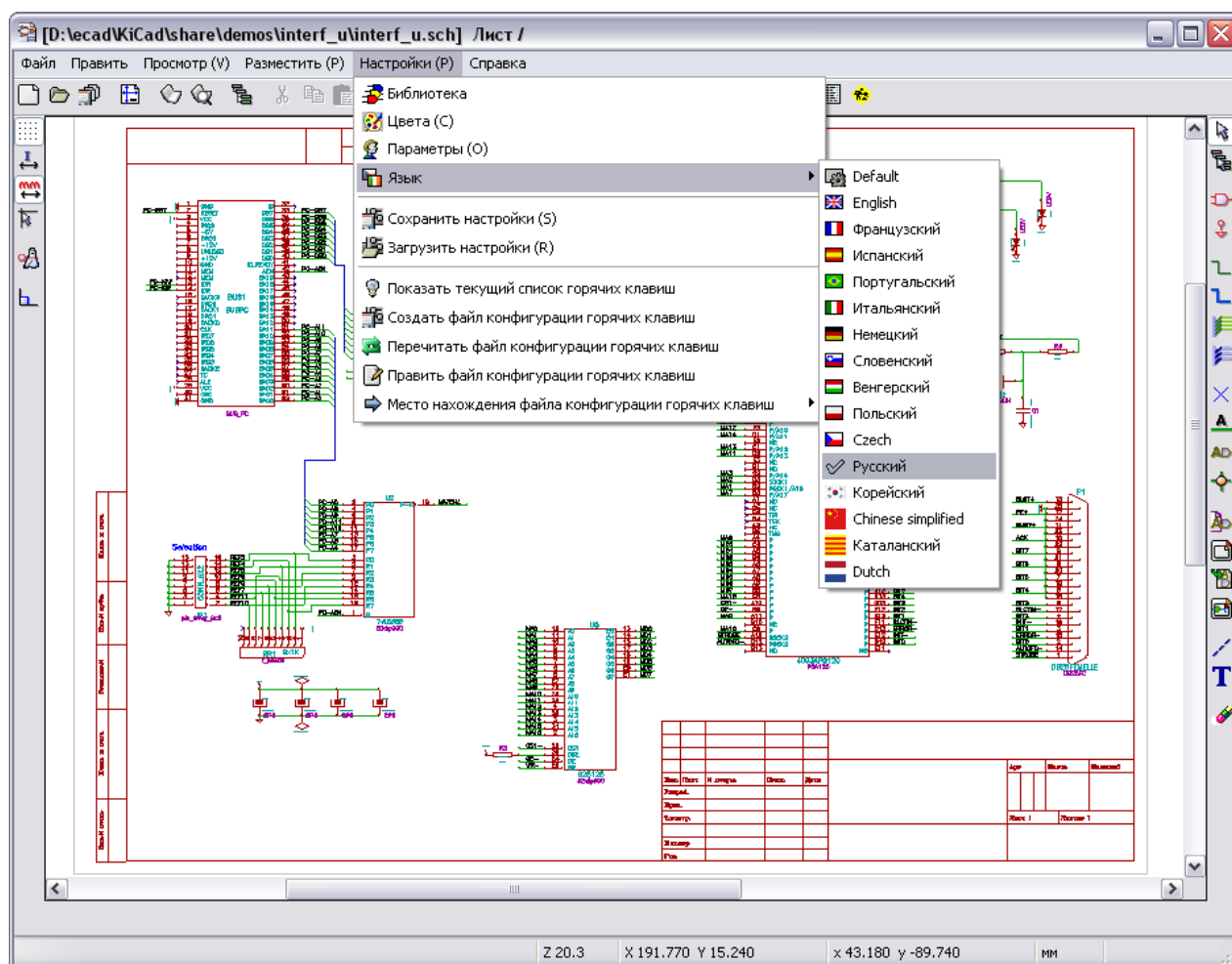
1. Gnucap. The Gnu Circuit Analysis Package. Users Manual, 2007

## 15. Послесловие к русской редакции перевода

KiCAD – система автоматизированного проектирования печатных плат (ПП) со свободной лицензией типа GPL и открытым программным кодом (open source).

Перевод руководства по графическому редактору электрических принципиальных схем EESchema выполнен на основе англоязычной версии руководства и свежих SVN-сборок программ KiCAD для операционных систем Linux и Windows XP.

В тексте использованы рисунки с английским интерфейсом, хотя редактор легко переключается и на русский интерфейс для пользователя (рисунок ниже).



В этом случае этот документ будет вызываться по команде **Справки/Содержание** из головного меню программы EESchema (папка на диске **kicad/doc/help/ru**). При англоязычном интерфейсе программа обращается за помощью в папку **.../help/en**.

В работе над руководством использовались не официальные сборки KiCAD, а специальные сборки системы для применения в России, модификация которых выполнена отечественными программистами из группы KiCAD Developers Team.

Эти сборки приспособлены для работы с кириллицей не только в интерфейсе программы, но и в проекте, лучше настроены на отечественные стандарты проектирования (другой вид позиционных обозначений компонентов, другая форматная рамка чертежей и т.д.). Они могут быть получены самостоятельно из исходных кодов программ (при определенных навыках), либо скачаны из Интернет (*например, с <ftp.ntcsm.ru/pub/kicad>*).

В Интернет можно найти главы wiki-энциклопедии, посвященные KiCAD, а также сайт для русскоязычных пользователей KiCAD (<http://kicad.sourceforge.net/phpBB2/>), где обсуждаются вопросы работы в KiCAD и даются полезные ссылки. В частности, это ссылки на библиотеки символов схемных элементов и модулей печатных плат, конвертированные из P-CAD и Eagle, а также на пособия по KiCAD, выпущенные в учебных заведениях страны (хороший лабораторный практикум подготовлен на кафедре САПР ВС Рязанского ГРТУ Семеновым А.А.).

В заключении следует сказать, что система KiCAD – это динамично развивающаяся САПР для разработки электрических схем и печатных плат. Так, в релизах этого года улучшены функции управления библиотеками компонентов и работа с комплексными иерархическими схемами, разработаны функции отката undo/redo, улучшено использование текстовых полей в атрибутах компонентов, введены вывод схем через формат Autodesk DXF (для Qcad, Компас, AutoCAD и др.) и новый векторный генератор текста с атрибутами наклона и жирности. Много нового превнесено в редактор проектов печатных плат PCBNEW. Не все функции Eeschema развернуто освещены в руководстве. Эти недостатки будут преодолеваются в будущем.

Каждый может принять участие в улучшении качества кода и документации САПР KiCAD.

#### **Адаптация документации и программы на русский язык:**

Викулов Ю.Н. (Саров)  
Федорушков А.А. (Москва)  
Плятов И.А. (Таганрог)

-----  
Сентябрь 2009 - Январь 2010