

## От автора

Это перевод на русский язык Руководства Пользователя программы NL5. Хочу обратить внимание читателей на следующее:

1. Огромная благодарность Владимиру Гололобову, чей первый перевод руководства является по существу основой настоящего перевода. Кое-что я добавил, кое-что отредактировал, но многие части просто нагло скопировал.
2. Так как сама программа по-прежнему на английском языке, все названия директив меню, кнопок, окон, стандартных функций Windows, а также специальных терминов, используемых в программе, на русский не переводятся, или используются как в английском, так и в русском вариантах. Например: Copy, Paste, trace (кривая), storage, transient (переходный процесс).
3. Может оказаться, что одно и то же называется по-разному в разных частях текста. Что ж, бывает: трудно за всем уследить.
4. Приношу извинения за корявость некоторых фраз и выражений. Точный дословный перевод с одного языка на другой не всегда возможен, а аккуратный «литературный» перевод занимает очень много времени. К тому же оказалось, что я не знаю принятых русскоязычных аналогов многих компьютерных/технических слов и выражений. Главной целью перевода было правильно передать смысл, и, надеюсь, это получилось.
5. Русское руководство скорее всего будет обновляться не так часто, как английское: нет времени.

Спасибо,  
Алексей Смирнов.

# NL5 Circuit Simulator

## Руководство Пользователя



Rev. 1.71

Rev. 1.71 31/04/2011 nl5.sidelinesoft.com

## ВЕРСИЯ

Эта версия Руководства Пользователя соответствует версии 1.71 (31/04/2011) программы NL5.

## LIMITED LIABILITY

Программа NL5 вместе с сопровождающими материалами поставляется на условиях «как есть» без гарантии любого сорта. Автор не принимает на себя никаких гарантийных обязательств, как явно выраженных, так и подразумеваемых, включая, но не ограничиваясь, какие-либо подразумеваемые гарантии коммерческой выгоды или пригодности для любых целей. Ни в каком случае автор не несет ответственности ни перед кем за прямой, косвенный или случайный ущерб или убытки, возникшие в результате использования или невозможности использования NL5.

## COPYRIGHTS

© 2011, А.Смирнов. Программа, Руководство Пользователя и перевод Руководства Пользователя защищены авторским правом. Никакая часть данного Руководства не может быть переведена или воспроизведена для коммерческих целей без письменного разрешения обладателя авторских прав. При публикации результатов, полученных при использовании программы, автор будет признателен за ссылку на NL5.

---

“**Smith**” is a registered trademark of Analog Instruments Company, New Providence, NJ. **Microsoft**, **Windows**, and **Microsoft Visual C++** are registered trademarks of Microsoft Corporation. **MATLAB** is a registered trademark of The MathWorks, Inc. **PYTHON** is a registered trademark of the Python Software Foundation. **Borland C++ Builder** is a registered trademark of Borland Corporation.

# Оглавление

|   |           |
|---|-----------|
| <b>I. Быстрый Старт.....</b>                            | <b>9</b>  |
| <b>Установка и запуск NL5 .....</b>                     | <b>10</b> |
| Установка NL5 .....                                     | 10        |
| Запуск NL5.....   | 10        |
| <b>Лицензия NL5 .....</b>                               | <b>11</b> |
| Single PC License .....                                 | 11        |
| Portable License.....                                   | 12        |
| Personal License .....                                  | 12        |
| Network License .....                                   | 12        |
| <b>Создание и Симуляция Вашей Первой Схемы .....</b>    | <b>13</b> |
| Ввод схемы .....  | 13        |
| Редактирование параметров компонентов .....             | 14        |
| Установки переходного процесса (transient) .....        | 15        |
| Данные переходного процесса .....                       | 16        |
| Запуск анализа переходного процесса.....                | 17        |
| Установки частотной характеристики (AC).....            | 18        |
| Данные AC.....  | 19        |
| Запуск AC анализа .....                                 | 20        |
| <b>II. Интерфейс Пользователя .....</b>                 | <b>21</b> |
| <b>Формат данных .....</b>                              | <b>22</b> |
| Нечувствительность к регистру (case-insensitivity)..... | 22        |
| Числа .....   | 22        |
| Имена .....   | 24        |
| Операторы.....  | 26        |
| Функции.....  | 27        |
| Выражения.....  | 27        |
| Язык C .....  | 28        |
| <b>Типы файлов.....</b>                                 | <b>29</b> |
| <b>Графический Интерфейс Пользователя.....</b>          | <b>30</b> |
| Основное Окно .....                                     | 30        |
| Основное Меню (Main Menu) .....                         | 30        |
| Основная инструментальная панель (Main Toolbar) .....   | 31        |
| Строка состояния (Status Bar).....                      | 31        |
| Панель выбора (Selection Bar) .....                     | 31        |
| Панель навигации .....                                  | 32        |
| Окна документа.....                                     | 33        |
| Другие окна .....                                       | 34        |
| Окна диалогов .....                                     | 34        |
| Помощь (Help).....                                      | 34        |
| Горячие клавиши .....                                   | 34        |
| <b>Предпочтения (Preferences).....</b>                  | <b>35</b> |
| Preferences (предпочтения) .....                        | 36        |

|   |           |
|---|-----------|
| Application (приложение) .....                                | 36        |
| Document (документ) .....                                     | 37        |
| Schematic (схема) .....                                       | 38        |
| Drawings (рисунки) .....                                      | 39        |
| Mouse (Schematic) (мышь, схема).....                          | 39        |
| Components (компоненты).....                                  | 40        |
| Symbols (символы компонентов).....                            | 40        |
| Warnings (предупреждения).....                                | 40        |
| Graphs (графики) .....  | 41        |
| Table (таблица) .....   | 42        |
| Legend (условные обозначения, список кривых на графике).....  | 42        |
| Annotation (аннотация) .....                                  | 43        |
| Text (текст).....   | 43        |
| Mouse (Graphs) (мышь, графики) .....                          | 44        |
| Transient (переходный процесс) .....                          | 44        |
| HTTP Server .....   | 45        |
| <b>Печать .....</b>   | <b>46</b> |
| Форматирование печати.....                                    | 47        |
| <b>III. Схема.....</b>  | <b>48</b> |
| <b>Окно схемы .....</b>                                       | <b>50</b> |
| Редактирование схемы .....                                    | 53        |
| Курсор .....  | 54        |
| Проводник .....   | 54        |
| Соединение (Connection).....                                  | 55        |
| Земля .....   | 56        |
| Компонент .....   | 57        |
| Вид компонента (View) .....                                   | 60        |
| Этикетка (Label) .....  | 61        |
| Атрибуты .....  | 62        |
| Рисунки (линия, прямоугольник, овал) .....                    | 63        |
| Текст и Переменные .....                                      | 64        |
| Прокрутка и масштабирование .....                             | 66        |
| Выделение и сброс выделения.....                              | 67        |
| Удаление .....  | 69        |
| Перемещение и копирование.....                                | 69        |
| Отключить и включить (Disable and Enable).....                | 71        |
| Повернуть, отразить, перевернуть (Rotate, Mirror, Flip) ..... | 72        |
| Формат .....  | 73        |
| Отменить и выполнить повторно (Undo, Redo) .....              | 74        |
| Команды редактирования схемы.....                             | 76        |
| Клавиатура и горячие клавиши .....                            | 78        |
| Операции с мышкой .....                                       | 79        |
| <b>Компоненты .....</b>                                       | <b>80</b> |
| Формулы .....   | 80        |
| Функции.....  | 81        |
| Редактирование конфигурируемого компонента (customized).....  | 82        |

|   |            |
|---|------------|
| Работа с подсхемами .....                                       | 86         |
| Работа с PWL.....   | 88         |
| Создание C-кода.....  | 102        |
| Работа с DLL .....  | 105        |
| <b>Окно Компонентов (Componets Window).....</b>                 | <b>107</b> |
| Инструментальная панель.....                                    | 108        |
| Список компонентов.....   | 109        |
| Выбор модели.....   | 110        |
| Редактирование параметров.....                                  | 110        |
| <b>Окно переменных (Variables Window).....</b>                  | <b>113</b> |
| Панель инструментов .....                                       | 113        |
| Редактирование переменных .....                                 | 114        |
| <b>Листы (Sheets).....</b>                                      | <b>115</b> |
| <b>Группы .....</b>   | <b>117</b> |
| <b>Проверка схемы .....</b>                                     | <b>119</b> |
| <b>Инструменты схемы.....</b>                                   | <b>121</b> |
| Renumber (перенумерация) .....                                  | 121        |
| Initial Conditions (начальные условия).....                     | 121        |
| Clean Up (чистка) .....   | 121        |
| Formulas (формулы).....   | 121        |
| Parameters (параметры).....                                     | 121        |
| Transform (преобразование).....                                 | 122        |
| <b>Свойства (Properties).....</b>                               | <b>124</b> |
| <b>IV. Анализ переходного процесса (Transient).....</b>         | <b>125</b> |
| <b>Симуляция .....</b>  | <b>127</b> |
| Алгоритм симуляции.....   | 127        |
| Данные симуляции.....   | 132        |
| <b>Установки переходного процесса (Transient Settings).....</b> | <b>133</b> |
| <b>Данные переходного процесса (Transient Data) .....</b>       | <b>136</b> |
| Инструментальная панель.....                                    | 137        |
| Traces (кривые).....  | 140        |
| <b>Симуляция .....</b>  | <b>142</b> |
| <b>Окно переходного процесса (Transient window).....</b>        | <b>144</b> |
| График.....   | 146        |
| Legend (условные обозначения) .....                             | 148        |
| Курсоры .....   | 149        |
| Текст.....  | 151        |
| Storage (накопитель данных, «хранилище») .....                  | 154        |
| Data table (Таблица данных) .....                               | 156        |
| Прокрутка и масштабирование .....                               | 158        |
| Команды переходного процесса.....                               | 161        |
| Клавиатура и горячие клавиши .....                              | 163        |

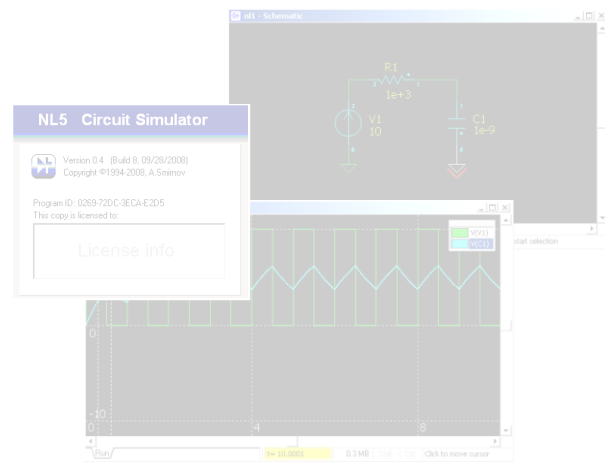
|   |            |
|---|------------|
| Операции с мышкой .....   | 164        |
| <b>Инструменты переходного процесса (Transient Tools) .....</b> | <b>165</b> |
| XY diagram (XY-диаграмма) .....                                 | 165        |
| Histogram (гистограмма) .....                                   | 168        |
| FFT (Быстрое Преобразование Фурье) .....                        | 172        |
| Eye diagram (Глазковая диаграмма) .....                         | 177        |
| Markers (маркеры) .....   | 179        |
| Power (мощность) .....  | 180        |
| <b>V. Частотная Характеристика (AC) .....</b>                   | <b>185</b> |
| <b>Симуляция .....</b>  | <b>187</b> |
| Метод линеаризации схемы .....                                  | 187        |
| Метод «включения» источника (Sweep AC source) .....             | 187        |
| Данные симуляции .....  | 189        |
| <b>Установки AC (AC Settings) .....</b>                         | <b>190</b> |
| <b>Данные AC (AC Data) .....</b>                                | <b>193</b> |
| Инструментальная панель .....                                   | 194        |
| Traces (кривые) .....   | 197        |
| <b>Симуляция .....</b>  | <b>199</b> |
| <b>Окно AC (AC window) .....</b>                                | <b>200</b> |
| График .....  | 202        |
| Legend (условные обозначения) .....                             | 204        |
| Курсоры .....   | 205        |
| Text .....  | 207        |
| Storage (накопитель данных) .....                               | 210        |
| Data table (Таблица данных) .....                               | 212        |
| Прокрутка и масштабирование .....                               | 214        |
| Команды AC .....  | 217        |
| Клавиатура и горячие клавиши .....                              | 219        |
| Операции с мышкой .....   | 220        |
| <b>Инструменты AC (AC Tools) .....</b>                          | <b>221</b> |
| Histogram (гистограмма) .....                                   | 221        |
| Диаграмма Смита .....   | 225        |
| Диаграмма Найквиста .....                                       | 228        |
| Диаграмма Николса .....   | 230        |
| Markers (маркеры) .....   | 232        |
| <b>VI. Инструменты (Tools) .....</b>                            | <b>233</b> |
| <b>Скрипт .....</b>   | <b>235</b> |
| Синтаксис скрипта .....   | 236        |
| Выполнение скрипта .....  | 237        |
| Примеры скрипта .....   | 237        |
| <b>Console (консоль) .....</b>                                  | <b>239</b> |
| Командная строка .....  | 240        |
| <b>Sweep (серия) .....</b>                                      | <b>241</b> |

|   |            |
|---|------------|
| <b>Оптимизация .....</b>                              | <b>244</b> |
| <b>HTTP link .....</b>                                | <b>246</b> |
| Запуск HTTP сервера .....                             | 246        |
| Посылка URL запроса .....                             | 247        |
| Запуск симуляции .....                                | 248        |
| Пример связи NL5-MATLAB .....                         | 249        |
| <b>VII. Attachments.....</b>                          | <b>251</b> |
| <b>1. Component Types, Models and Parameters.....</b> | <b>252</b> |
| Label .....   | 253        |
| A – Amperemeter.....                                  | 259        |
| C – Capacitor .....                                   | 260        |
| C – Voltage controlled capacitor .....                | 262        |
| C – Current controlled capacitor.....                 | 263        |
| D – Diode.....  | 264        |
| D – Zener .....                                       | 268        |
| D – Bidirectional zener .....                         | 269        |
| D – Bridge rectifier .....                            | 270        |
| D – Logic controlled thyristor.....                   | 271        |
| D – Voltage controlled thyristor .....                | 272        |
| D – Current controlled thyristor.....                 | 273        |
| F – Function .....                                    | 274        |
| F – Function-2.....                                   | 279        |
| F – Custom function.....                              | 287        |
| I – Current source.....                               | 289        |
| I – Voltage controlled current source.....            | 295        |
| I – Current controlled current source .....           | 299        |
| L – Inductor.....                                     | 303        |
| L – Voltage controlled inductor .....                 | 305        |
| L – Current controlled inductor .....                 | 306        |
| L – Coupled inductors.....                            | 307        |
| L – Custom coupled inductors .....                    | 308        |
| O – Amplifier .....                                   | 310        |
| O – Differential amplifier .....                      | 315        |
| O – Summing amplifier .....                           | 320        |
| O – Voltage controlled amplifier .....                | 324        |
| O – Current controlled amplifier.....                 | 325        |
| R – Resistor.....                                     | 326        |
| R – Potentiometer .....                               | 328        |
| R – Voltage controlled resistor .....                 | 329        |
| R – Current controlled resistor.....                  | 330        |
| S – Switch .....                                      | 331        |
| S – Logic controlled switch .....                     | 334        |
| S – Voltage controlled switch .....                   | 336        |
| S – Current controlled switch .....                   | 339        |
| S – SPDT switch .....                                 | 342        |
| S – SPDT logic controlled switch.....                 | 347        |
| S – SPDT voltage controlled switch.....               | 351        |




|   |            |
|---|------------|
| S – SPDT current controlled switch.....     | 355        |
| T – NPN transistor .....                    | 359        |
| T – PNP transistor .....                    | 363        |
| T – N-FET.....                              | 367        |
| T – P-FET .....                             | 370        |
| V – Voltage source.....                     | 373        |
| V – Voltage controlled voltage source.....  | 379        |
| V – Current controlled current source ..... | 383        |
| V – Voltmeter .....                         | 387        |
| W – Winding.....                            | 388        |
| W – Transformer.....                        | 389        |
| W – Differential transformer .....          | 390        |
| W – Custom transformer.....                 | 391        |
| W – Wattmeter.....                          | 393        |
| X – Delay .....                             | 394        |
| X – Transmission line .....                 | 395        |
| X – Sample/Hold.....                        | 397        |
| X – Directional coupler.....                | 399        |
| X – Block-2.....                            | 400        |
| X – Block-3.....                            | 401        |
| X – Block-4.....                            | 402        |
| X – Block-6.....                            | 403        |
| X – Block-8.....                            | 404        |
| X – Custom block .....                      | 405        |
| X – Code .....                              | 406        |
| Y – Logic-1 .....                           | 408        |
| Y – Logic-2.....                            | 409        |
| Y – Logic-3 .....                           | 410        |
| Y – Custom logic .....                      | 412        |
| Y – D-trigger.....                          | 414        |
| Y – RS-trigger.....                         | 415        |
| Y – Schmitt trigger.....                    | 417        |
| Y – Logic generator .....                   | 419        |
| <b>2. Operators .....</b>                   | <b>422</b> |
| <b>3. Functions.....</b>                    | <b>424</b> |
| <b>4. C language syntax .....</b>           | <b>427</b> |
| <b>5. Script commands.....</b>              | <b>429</b> |
| <b>6. END USER LICENSE AGREEMENT.....</b>   | <b>434</b> |

# I. Быстрый Старт





## Установка и запуск NL5

### Установка NL5

Одним из преимуществ NL5 является то, что она состоит только из одного файла:  n15.exe. NL5 не требует специальной установки: просто скопируйте n15.exe в любую директорию. Вы можете иметь несколько копий n15.exe в разных директориях. Чтобы переписать NL5 на другой компьютер, достаточно скопировать n15.exe.


Следующие файлы не являются необходимыми, но если они используются, поместите их в одну директорию с n15.exe:

-  n15.chm - NL5 help file (файл Помощи).
-  n15.nll - NL5 license file (файл Лицензии).

Чтобы создать иконку NL5 на рабочем столе, щелкните правой клавишей мышки по n15.exe в проводнике Windows и выберите команду **Отправить | Рабочий стол (создать ярлык)**.

Последнюю версию NL5 можно найти на сайте [nl5.sidelinesoft.com](http://nl5.sidelinesoft.com).

### Запуск NL5

Чтобы запустить NL5, дважды щелкните по n15.exe или по иконке NL5: . На экране появится заставка с информацией о номере версии, дате и информацией о лицензии:



Это окно закроется само через несколько секунд.

NL5 может также запускаться из командной строки с ключами и/или параметрами (или без ключей и параметров).

**Ключи.** Ключ – это текст, начинающийся с символа ‘-’ или ‘/’. Следующие ключи могут быть использованы в командной строке:

`-http` - запустить HTTP сервер.

Например:

```
>n15.exe -http
```

**Параметры.** Параметр – это имя файла. Один или более файлов разного типа могут быть использованы в качестве параметров. Например:

```
>n15.exe rc.n15 - загрузить схему из файла rc.n15
>n15.exe tran.n1t - загрузить данные переходного процесса из файла tran.n1t
>n15.exe rc.n15 pref.nlp - загрузить схему из файла schematic rc.n15 и preferences из
файла pref.nlp
```

Файл с расширением “txt” в качестве параметра командной строки рассматривается как скрипт, и будет выполнен немедленно. Например:

```
>n15.exe script.txt - загрузить и выполнить скрипт из файла script.txt
```

## Лицензия NL5

Без лицензии NL5 работает как **Демо-версия**. Демо-версия полностью идентична полнофункциональной версии, за исключением того, что количество компонентов в схеме не может превышать 20. Хотя в Демо-версии нельзя ввести больше 20 компонентов, она может читать из файла и симулировать схемы с неограниченным числом компонентов, если эти схемы созданы полной версией программы.

На сайте [n15.sidelinesoft.com](http://n15.sidelinesoft.com) можно заказать несколько видов лицензий, которые имеют разные ограничения и отличаются типом защиты. Все лицензии (за исключением Portable) используют файл лицензии `n15.n11`. Этот файл должен находиться в той же директории, что и `n15.exe` (или `NL5LicenseServer.exe` для сетевой лицензии).

## Single PC License

Single PC лицензия обеспечивает полнофункциональную работу только на одном компьютере. Лицензия привязывается к специфичной информации о компьютере (“PC fingerprint”). Когда NL5 запускается, она сравнивает “fingerprint” информацию из файла лицензии с информацией компьютера, и запускается в полнофункциональном режиме, только если эти данные идентичны. «PC fingerprint» можно получить в диалоговом окне **Support (Help | Support)**; затем его необходимо ввести в окне “fingerprint” на страничке заказа лицензии веб-сайта.

Single PC лицензия предлагается бесплатно на ограниченное время как **Trial License** (пробная лицензия). Когда время пробной лицензии истечет, можно купить постоянную Single PC Лицензию. Полную информацию можно получить на сайте [nl5.sidelinesoft.com](http://nl5.sidelinesoft.com).

## Portable License

Portable лицензия использует USB устройство – **dongle** – для хранения информации о лицензии. NL5 работает в полнофункциональном режиме на любом компьютере при условии, что USB dongle включен в USB порт компьютера. Благодаря “бездрайверной” технологии никаких проблем при использовании USB порта не предвидится.



## Personal License

Personal License (персональная лицензия) позволяет работать в полнофункциональном режиме неограниченное время на любом компьютере. Эта лицензия выдается индивидуальным пользователям и может быть установлена только на компьютерах, где сам владелец лицензии регулярно пользуется программой.

## Network License

Network License (сетевая лицензия) устанавливается на сервере – компьютере с операционной системой Windows. Лицензия включает в себя программу `NL5LicenseServer.exe` и файл лицензии `nl5.nll`. NL5 может работать в полнофункциональном режиме на любом компьютере, который имеет доступ к компьютеру-серверу по сети. При запуске NL5 запрашивает у сервера информацию о лицензии. Количество одновременно работающих программ NL5 (количество рабочих мест) не ограничено.

## Создание и Симуляция Вашей Первой Схемы

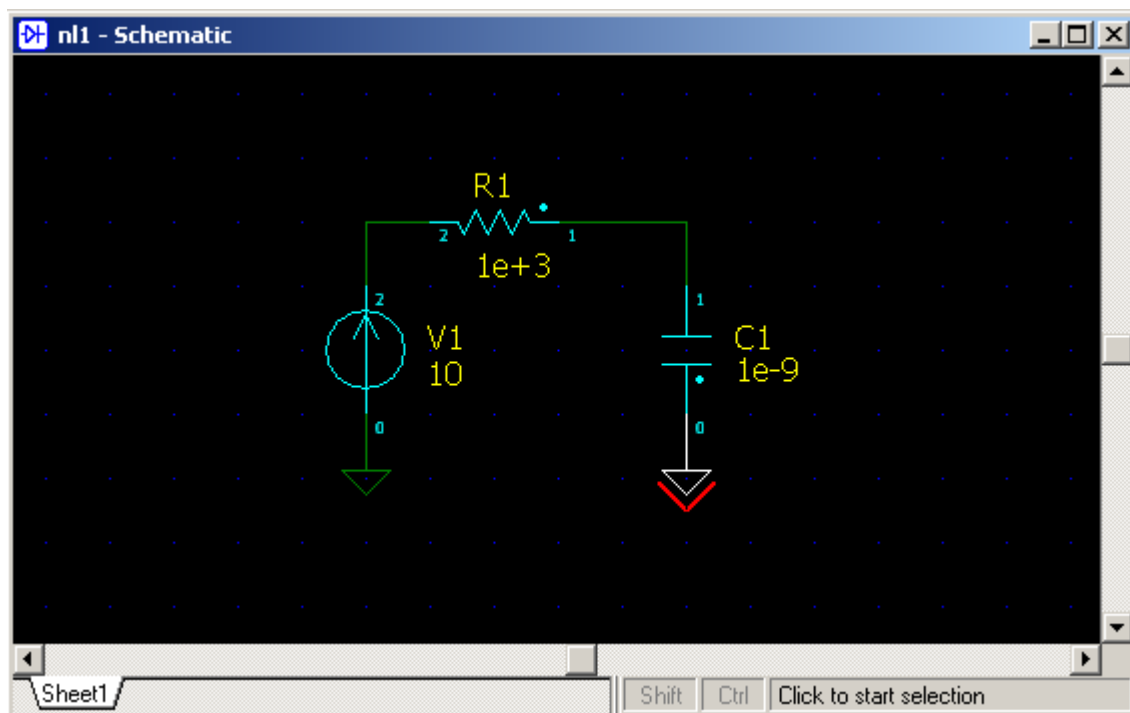
### Ввод схемы

Ввод и редактирование могут быть сделаны с использованием клавиатуры, мышки или и того, и другого. Ниже приведено пошаговое описание того, как ввести простую схему, используя клавиатуру.

Когда NL5 открывается, создается пустая схема. Красный курсор расположен в середине экрана и показывает направо.


- Нажмите **«пробел»**, чтобы переключиться в режим рисования.
- Нажмите **«стрелку вниз»** несколько раз, чтобы нарисовать короткий проводник вниз.
- Нажмите клавишу **V** и следом нажмите **Enter**, чтобы добавить источник напряжения.
- Нажмите клавишу **G** для размещения земли. Теперь курсор переключился обратно в режим выделения.
- Нажмите **«стрелку вверх»** несколько раз, чтобы переместить курсор к начальной точке.
- Нажмите **«стрелку вправо»**, чтобы изменить направление, затем нажмите **«пробел»** для перехода в режим рисования.
- Нажмите **«стрелку вправо»** несколько раз, вы нарисуете короткий горизонтальный проводник.
- Нажмите клавишу **R** и затем нажмите **Enter**, чтобы поместить резистор.
- Вновь нажмите **«стрелку вправо»** несколько раз, а затем несколько раз **«стрелку вниз»**.
- Нажмите клавишу **C** и следом нажмите **Enter**, этим вы добавите конденсатор.
- Нажмите клавишу **G** для размещения земли.

Схема готова. Вот, что вы должны увидеть:

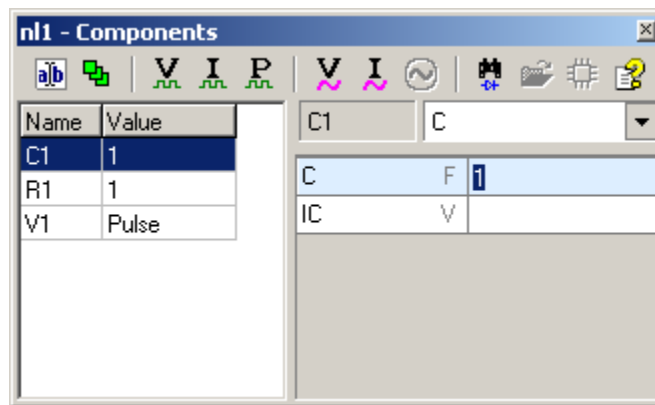


## Редактирование параметров компонентов


Теперь вы будете использовать мышку для выделения компонентов и клавиатуру для ввода параметров.

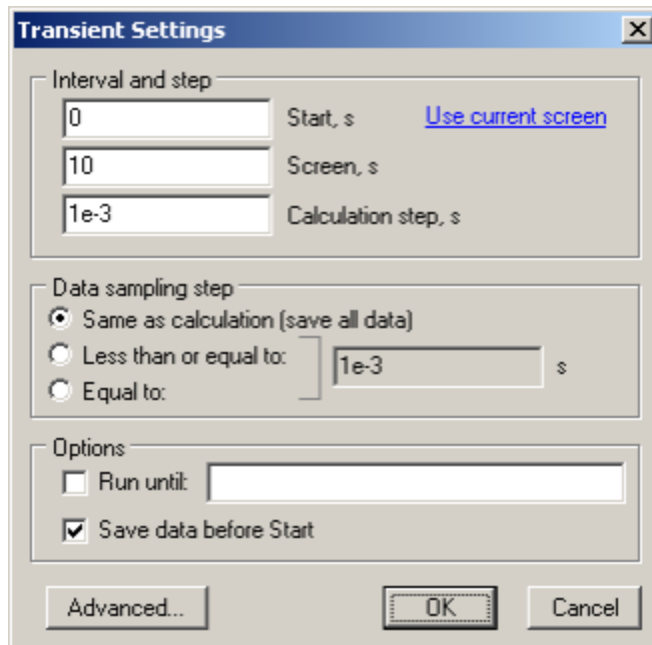
- Дважды щелкните по источнику напряжения **V1**. Откроется окно компонентов. На левой панели должен быть выделен V1. Правая панель показывает имя компонента (**V1**), модель (**V**) и параметры (пока один параметр «**V**»).
- Щелкните кнопку  справа от имени модели. Выпадающее окно покажет доступные модели источников напряжения.
- Выберите **Pulse**.
- Дважды щелкните по резистору **R1** на левой панели. Значение резистора «1e+3» будет выбрано на правой панели.
- Нажмите клавишу **1** («один»), сопротивление R1 станет 1 Ом.
- Дважды щелкните по конденсатору **C1** на левой панели, и измените емкость с «1e-9» на «1».

Компоненты готовы. Вот результат:




## Установки переходного процесса (transient)

Щелкните кнопку **Transient settings** инструментальной панели , или выберите команду **Transient | Settings** в Основном Меню. Вам не нужно ничего здесь менять, но вы можете это сделать, если хотите. Щелкните кнопку **OK**.



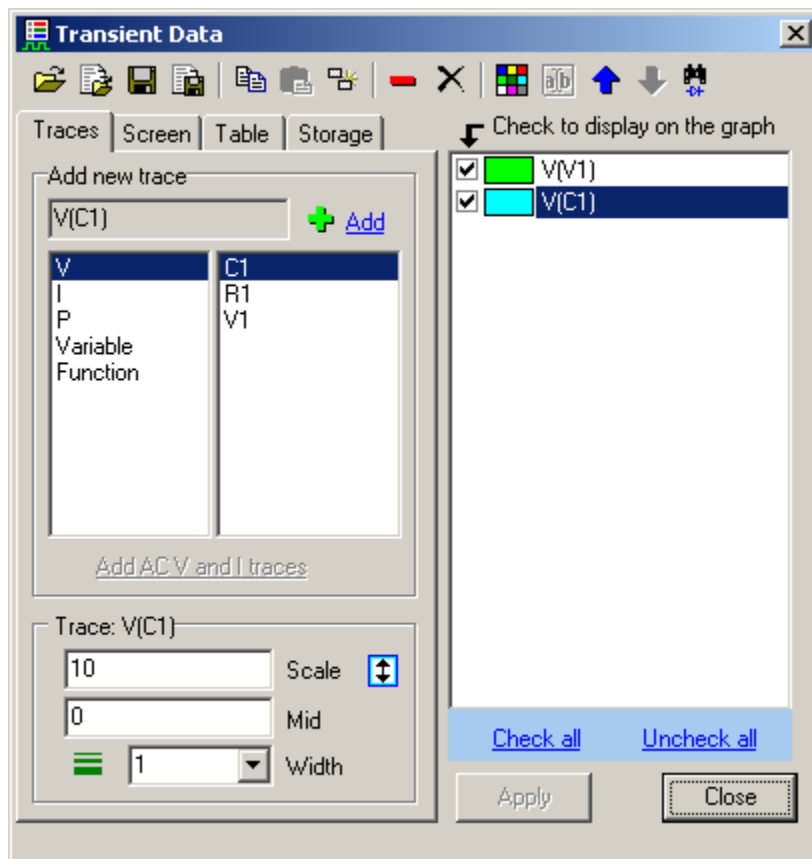


## Данные переходного процесса


Щелкните по кнопке **Transient data** инструментальной панели , или выберите команду **Transient | Data** в Основном Меню. Удостоверьтесь, что выбрана закладка **Traces (кривые)**.

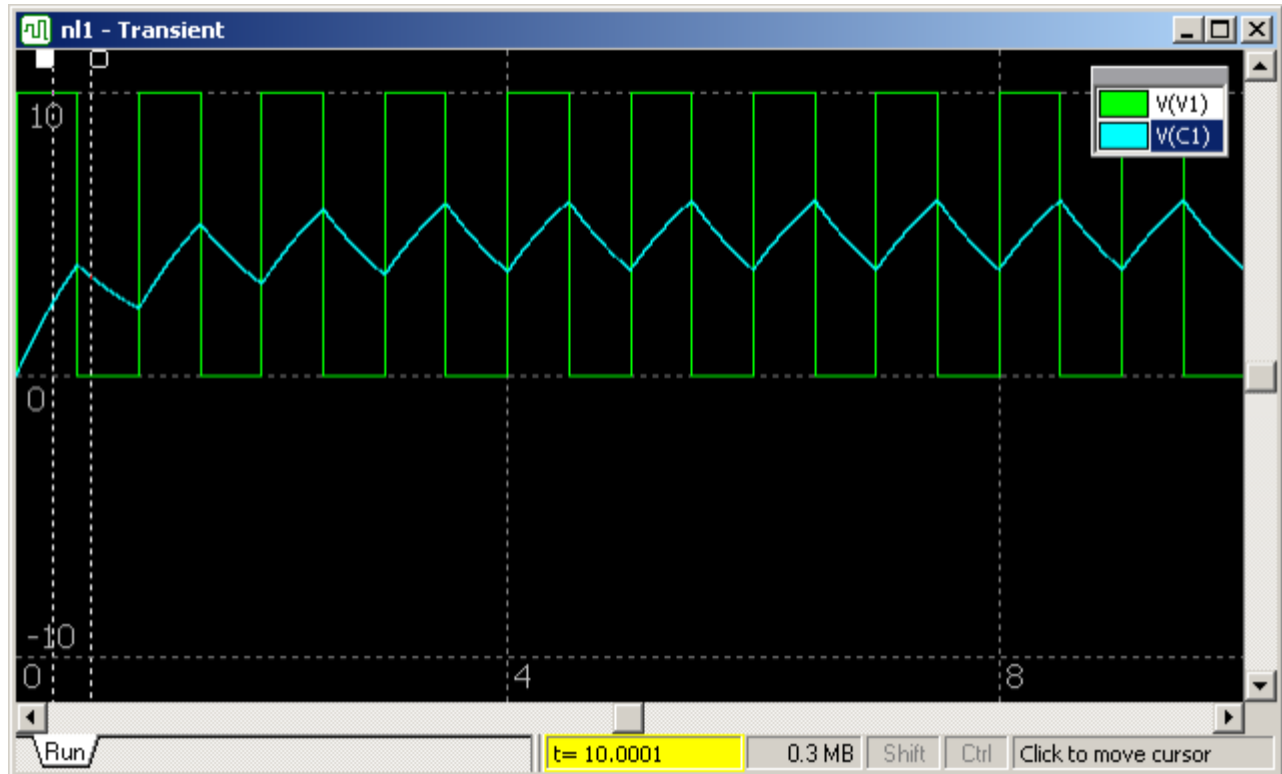
- В окне панели **Add new trace** выделите V (кривая напряжения).
- Дважды щелкните по **V1** и **C1** в списке компонентов. Кривые напряжения будут добавлены в список кривых.
- Щелкните по кнопке **Close**.

Вот вид окна (до закрывания):




## Запуск анализа переходного процесса

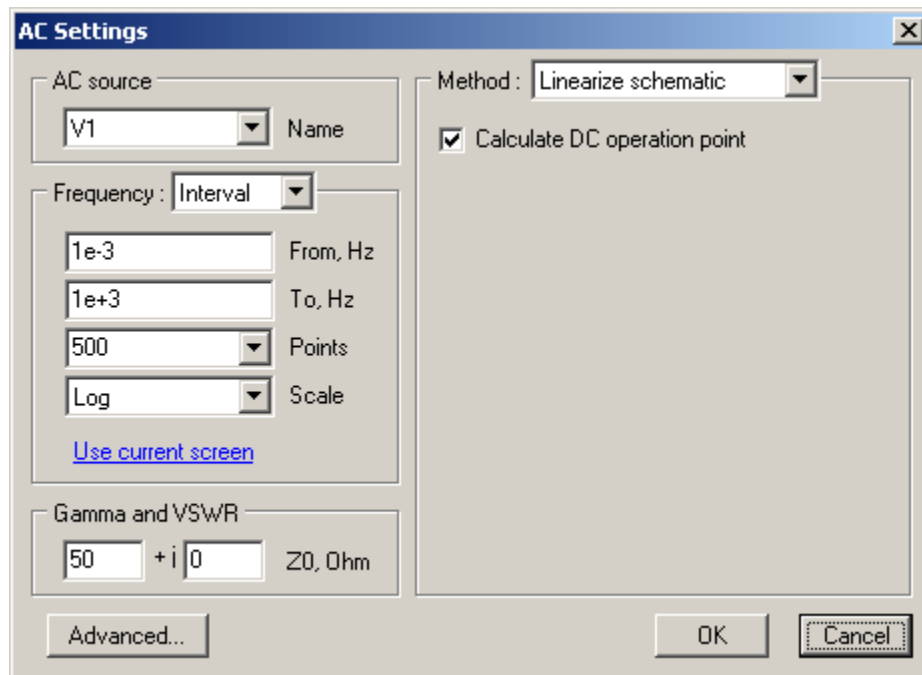
Щелкните по кнопке **Start transient** инструментальной панели , или выберите команду **Transient | Start** в Основном Меню. Переходной процесс будет рассчитан и отображен:




## Установки частотной характеристики (АС)

Щелкните по кнопке **AC settings** инструментальной панели , или выберите команду **AC | Settings** в Основном Меню.

- Щелкните по выпадающему списку **Name** на панели **AC source** и выберите **V1**.
- Щелкните по кнопке **OK**.

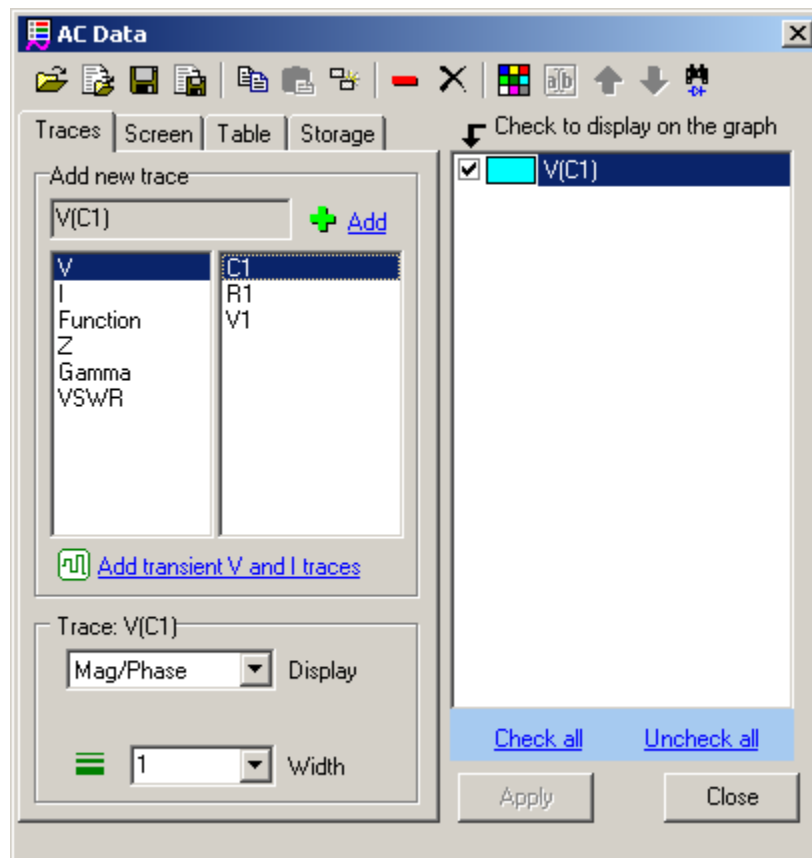


## Данные АС


Щелкните по кнопке **AC data** инструментальной панели , или выберите команду **AC | Data** в Основном Меню. Убедитесь, что выбрана закладка **Traces (кривые)**.

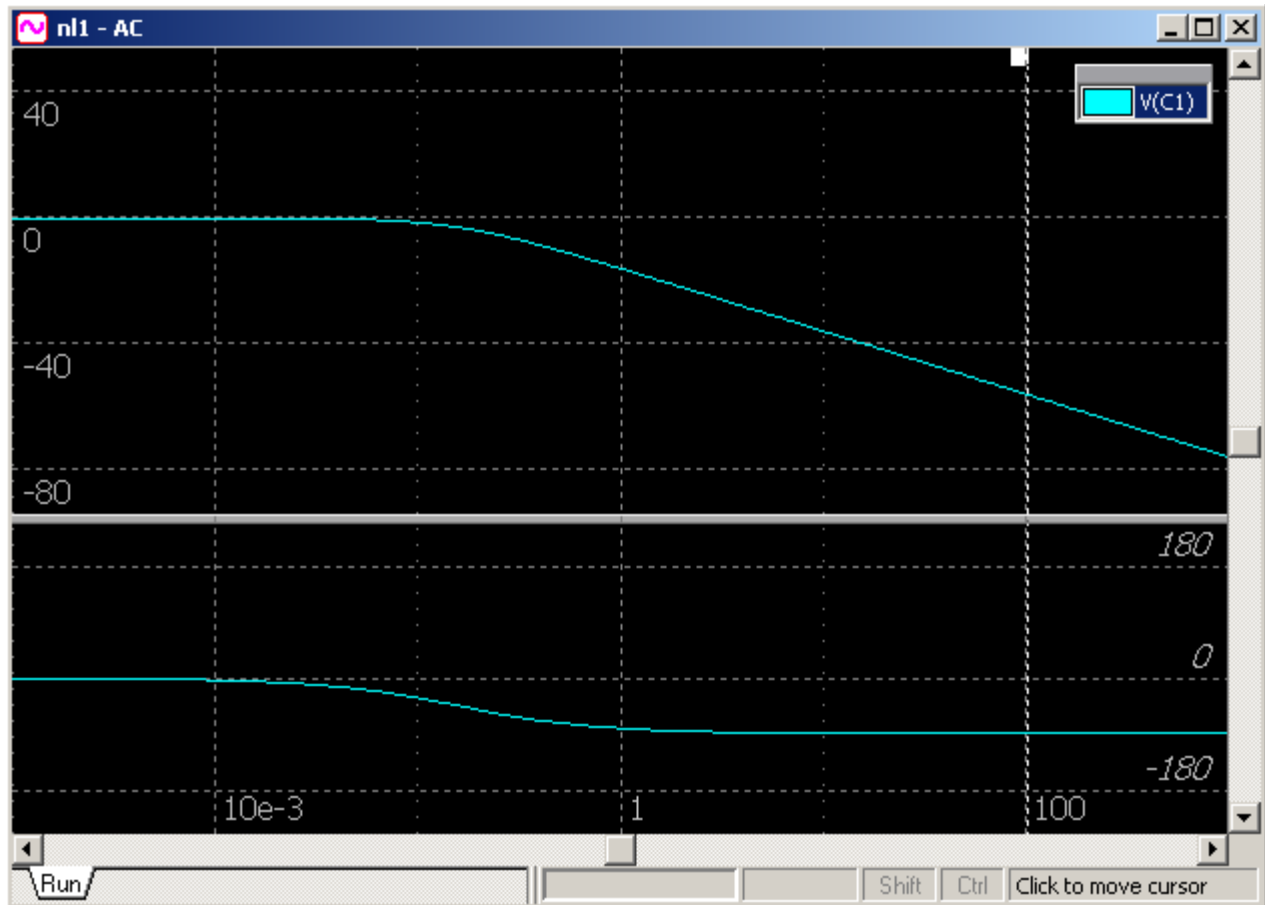
- На панели **Add new trace** выберите **V** (кривая напряжения).
- Дважды щелкните по **C1** в списке компонентов. Кривая напряжения будет добавлен в список.
- Щелкните по кнопке **Close**.

Вот вид окна (до закрывания):

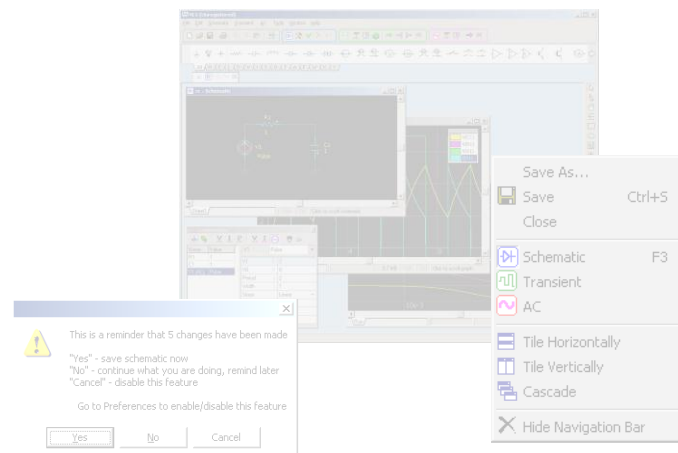


## Запуск АС анализа

Щелкните по кнопке **Start AC** на инструментальной панели , или выберите команду **AC | Start** в Основном Меню. Частотная характеристика будет рассчитана и отображена:



## II. Интерфейс Пользователя



**Формат данных** в NL5 в основном согласован с обычной инженерной и научной практикой. Это делает его доступным и понятным любому человеку, знакомому с популярными программными пакетами. NL5 использует несколько специализированных **типов файлов** для схем и данных анализа. **Графический Интерфейс Пользователя (GUI)** программы основан на стандартной архитектуре Microsoft Windows Multi-Document Interface (MDI). Он состоит из разных компонентов GUI, таких как окна, диалоги, меню, инструментальные панели и т.д. NL5 поддерживает множество команд и горячих клавиш, обычно используемых в разнообразных приложениях Windows, например: **Edit | Copy (Ctrl-C)**, **Edit | Paste (Ctrl-V)**, **Window | Tile**, использование клавиши **Ctrl** совместно с мышкой для операций выделение/копирование, использование полос прокрутки в окнах и т.д. Другие команды интуитивно понятны, так что начало работы со схемой не займет много времени. **Preferences** (предпочтения) используются для индивидуальной настройки интерфейса, «вида и поведения» (look and feel), и параметров по умолчанию (default). Настройка **печати** позволяет приспособить вид и отформатировать окно вывода на печать.

## Формат данных

### Нечувствительность к регистру (case-insensitivity).

Все текстовые данные в NL5, такие как имена компонентов, переменные, функции, команды и т.д., **не чувствительны к регистру**, если не указано обратное. Буквы нижнего регистра и верхнего регистра рассматриваются как равноценные. Например:

```
Rin = RIN = rin
sin(45) = SIN(45)
```

### Числа

Формат чисел в NL5 очень гибкий и согласован со многими часто используемыми стандартами и стилями. Число может использовать экспоненциальный множитель **E** или **e**, и **чувствительные к регистру (case-sensitive)** буквенные множители:

| Буква(ы) | Множитель  |
|----------|------------|
| T        | $10^{12}$  |
| G        | $10^9$     |
| M, mg    | $10^6$     |
| k, K     | $10^3$     |
| m        | $10^{-3}$  |
| u, mk    | $10^{-6}$  |
| n        | $10^{-9}$  |
| p        | $10^{-12}$ |

Например:

1.3e+3 47E-9 100k 0.33mk 2.2M

За буквенным множителем может следовать любой текст, который рассматривается как единица измерения и игнорируется:

1.3kOhm 47nF 0.1mkH 333ps

Любой текст, начинающийся не с буквенного множителя, рассматривается как единица измерения и игнорируется:

1.3Ohm 0.001F 0.1H 333apples

Буквенные множители и единицы измерения (с буквенным множителем или без него) могут использоваться вместо десятичной точки:

1k3 5n6 3nF3 47F0 2s2

Ноль перед десятичной точкой или буквенным множителем может быть опущен:

.47 n47 uF5

Число может быть введено в децибелах, используя суффикс dB (не чувствительный к регистру). Величина в dB будет автоматически преобразована в нормальное число (работает только для положительных значений dB):

20db = 100  
3.01dB = 1.41416472507  
6DB02 = 1.99986186963

Бесконечное значение обозначается как:

inf

Мнимая часть комплексного числа имеет букву **нижнего регистра** «j» в конце числа. Буква «j» не может использоваться одна, а только как суффикс:

50+45j  
1+1e-3j = 1+.001j  
30j  
1+j - неверно! Правильный формат: 1+1j

Следующие предопределенные константы (не чувствительный к регистру) могут быть использованы в выражениях:

PI = pi = 3.14159265359  
RAD = rad = 180/pi = 57.2957795131  
LOW = Low = low - низкий логический уровень, Вольт  
HIGH = High = high - высокий логический уровень, Вольт

Константа RAD может использоваться для перевода градусов в радианы, а радиан в градусы:

Degrees = Radians\*RAD  
Radians = Degrees/RAD



где `Degrees` — значение в градусах, а `Radians` – в радианах.

Будучи введено в одном из многочисленных разрешенных текстовых форматов, число автоматически преобразуется и хранится в вещественном числовом формате двойной точности (**double**). Отображаются числа обычно в инженерной нотации с экспоненциальным множителем и степенью кратной трем:

| Введенное | Отображаемое |
|-----------|--------------|
| 1k3       | 1.3e+3       |
| 47e-8     | 470e-9       |
| 5600000   | 5.6e+6       |

## Имена

**Компонент.** Когда новый компонент создается, ему присваивается предопределенное имя: буква плюс порядковый номер:

`R1, V2, C123`

Затем компонент может быть переименован. Имя не чувствительно к регистру и может состоять из любых букв и символов. При использовании в формуле или функции имя заключается в кавычки:

`"R out", "V pulse", "+12V"`

Однако если имя начинается с буквы и содержит только буквы и цифры, оно может использоваться без кавычек:

`Rout, V123, Plus12V`

Если компонент был переименован, его имя будет автоматически модифицировано во всех случаях появления имени компонента в именах кривых, формулах и функциях.

Для доступа к параметрам компонента в формуле, функции или выражении в скрипте, используйте имя компонента, сопровождаемое точкой и именем параметра:

`R1.R, V2.slope, C123.IC, "R out".R`

Если имя параметра не задано, будет использован первый параметр компонента:

`R1 = R1.R  
C2 = C2.C`

Для доступа к компоненту, являющемуся частью подсхемы (`subcircuit`), используйте имя подсхемы компонента, сопровождаемое точкой и именем компонента в подсхеме. Вложение уровней не ограничено: компонент внутри подсхемы, которая, в свою очередь, является частью подсхемы, может быть доступен с помощью аналогичной нотации:

```
X1.R2
X1.F1.V3.period
```

где X1 и F1 - подсхемы.

Для доступа к имени модели компонента (в скрипте или командной строке) используйте имя компонента, сопровождаемое точкой и словом «model»:

```
V1.model=pulse
```

**Переменная.** Имя переменной имеет тот же формат, что и имя компонента, за исключением того, что переменная не имеет параметров. Например:

```
Freq, "max limit", X1.var
```

Имейте в виду, что это относится только к переменным **схемы**: определенным в **Variables window (Окно Переменных)**. Имена переменные, определенных в С коде (скрипт, Code компонент) подчиняются стандарту языка С.

**Trace (кривая).** Имя кривой переходного процесса или частотной характеристики (АС), содержащих данные текущей симуляции, состоит из буквы заданного типа кривой (**V, I, P**), сопровождаемой именем компонента в скобках:

```
V(R1), I(C2), P(L3)
```

Такие кривые не могут быть переименованы. Все другие кривые (сдублированные, загруженные из файла, или скопированные) могут быть переименованы в произвольный текст:

```
"Copy of V(R1)"
"Old trace of R1"
"V pulse"
```

Имя кривой типа **Function** - это сама функция. Переименование кривой изменит функцию:

```
"V(r1)*V(r1)/r1"
```

**Данные кривой и курсоров.** Данные кривой и курсоров, показанные в таблице переходного процесса (transient) или АС (значение на курсоре, минимум, максимум, среднее и т.д.) и значение кривой в определенный момент времени (на определенной частоте) может быть использовано в скрипте. Кривая должна быть добавлена в список данных переходного процесса или АС, но не обязательно, чтобы она отображалась на графике или в таблице данных. Чтобы получить данные кривой и курсоров, используется следующая нотация:

- **Курсоры (экран)**

|       |  |
|-------|--|
| left  | – положение левого курсора, или левый край экрана (если курсоры выключены)   |
| right | – положение правого курсора, или правый край экрана (если курсоры выключены) |
| delta | = right-left   |

- **Амплитуда переходного процесса**

|                          |   |
|--------------------------|---|
| <code>V(R1).(1.2)</code> | – значение в $t=1.2$                                    |
| <code>V(R1).left</code>  | – значение на левом курсоре или у левого края экрана    |
| <code>V(R1).right</code> | – значение на правом курсоре или у правого края экрана  |
| <code>V(R1).delta</code> | $= V(R1).right - V(R1).left$                            |
| <code>V(R1).min</code>   | – минимум   |
| <code>V(R1).max</code>   | – максимум  |
| <code>V(R1).pp</code>    | – от пика до пика ( $\max - \min$ )                     |
| <code>V(R1).mean</code>  | – среднее   |
| <code>V(R1).rms</code>   | – RMS   |
| <code>V(R1).acrms</code> | – RMS сигнала после вычета среднего значения из сигнала |

- **Амплитуда АС (AC magnitude)**

|                          |  |
|--------------------------|--|
| <code>V(R1).(1.2)</code> | – значение в $f=1.2$                                   |
| <code>V(R1).left</code>  | – значение на левом курсоре или у левого края экрана   |
| <code>V(R1).right</code> | – значение на правом курсоре или у правого края экрана |
| <code>V(R1).delta</code> | $= V(R1).right - V(R1).left$                           |
| <code>V(R1).min</code>   | – минимум  |
| <code>V(R1).max</code>   | – максимум   |
| <code>V(R1).pp</code>    | – от пика до пика ( $\max - \min$ )                    |
| <code>V(R1).slope</code> | – наклон усиления, dB/dec                              |

- **Фаза АС (AC phase)**

|                                |  |
|--------------------------------|--|
| <code>V(R1).phase.(1.2)</code> | – значение в $f=1.2$                                   |
| <code>V(R1).phase.left</code>  | – значение на левом курсоре или у левого края экрана   |
| <code>V(R1).phase.right</code> | – значение на правом курсоре или у правого края экрана |
| <code>V(R1).phase.delta</code> | $= V(R1).phase.right - V(R1).phase.left$               |
| <code>V(R1).phase.min</code>   | – минимум  |
| <code>V(R1).phase.max</code>   | – максимум   |
| <code>V(R1).phase.pp</code>    | – от пика до пика ( $\max - \min$ )                    |

## Операторы

NL5 поддерживает следующие арифметические и логические операторы:

|                 |                         |                 |                 |
|-----------------|-------------------------|-----------------|-----------------|
| <code>=</code>  | <code>&amp;&amp;</code> | <code>==</code> | <code>%</code>  |
| <code>+=</code> | <code>  </code>         | <code>!=</code> | <code>^</code>  |
| <code>-=</code> | <code>&lt;</code>       | <code>+</code>  | <code>++</code> |
| <code>*=</code> | <code>&lt;=</code>      | <code>-</code>  | <code>--</code> |
| <code>/=</code> | <code>&gt;</code>       | <code>*</code>  |                 |
| <code>?:</code> | <code>&gt;=</code>      | <code>/</code>  |                 |

Полный лист операторов с описанием и примерами приведен в **Приложении 2**.

## Функции

NL5 поддерживает много стандартных и уникальных математических функций. Для удобства пользователей некоторые функции могут иметь несколько разных имен (например,  $\log_{10}$  и  $\lg$ ), так что пользователь может использовать имя, к которому он больше привык. Следующие

|         |           |            |              |
|---------|-----------|------------|--------------|
| sin     | exp       | mag, abs   | im           |
| cos     | ln, log   | sign       | par          |
| tan, tg | lg, log10 | db         | random, rand |
| sqrt    | lb, log2  | min        | limit, lim   |
| sqr     | asin      | max        | islow        |
| sq      | acos      | int, round | ishigh       |
| pow     | atan      | phase      |              |
| pwr     | atan2     | re         |              |

функции поддерживаются в NL5:

Полный лист функций с описанием и примерами приведен в **Приложении 3**.

## Выражения

Выражение может состоять из:

- Чисел.
- Предопределенных констант (PI и RAD).
- Имен компонентов, параметров и переменных.
- Локальных переменных скрипта или переменных C-кода.
- Операторов.
- Функций.
- Скобок с неограниченным уровнем вложения.

Например:

```
2*2
2^10-1
sin(2*PI*f) // "f" - это переменная схемы
max(R1,R2,R3)
1/((R1+R2)*C1)
```

Имейте в виду, что NL5 работает только с вещественными числами (типа **double**). Даже если число было введено как целое (без десятичной точки), оно будет преобразовано в вещественный формат до выполнения каких-либо операций над этим числом. Параметры и результаты всех операторов и функций также вещественные. Некоторые операторы и функции могут также использовать комплексные числа.

Выражения могут быть использованы вместо чисел почти во всех полях ввода в диалоговых окнах, и для некоторых параметров компонентов. Когда вы нажмете на клавишу **Enter** или щелкнете по кнопке **OK** или **Apply** (если она есть), выражение будет немедленно вычислено и замещено численным результатом.

## Язык C

В NL5 реализован упрощенный язык программирования C. Он используется в скрипте и в C модели компонента **Code**. Несмотря на то, что поддерживаются не все возможности стандартного языка C, имеющиеся функции позволяют эффективно решать многие задачи. Вкратце, реализованный в NL5 язык C отличается от стандартного следующим:

- Поддерживается только тип данных **double**.
- побитовые операции не поддерживаются.
- структуры и объединения не поддерживаются.
- указатели и ссылки не поддерживаются.
- **goto** оператор не поддерживается.

Реализованы следующие ключевые слова и операторы::

|                        |                         |                       |
|------------------------|-------------------------|-----------------------|
| <code>double</code>    | <code>default</code>    | <code>continue</code> |
| <code>if...else</code> | <code>for</code>        | <code>break</code>    |
| <code>switch</code>    | <code>while</code>      | <code>return</code>   |
| <code>case</code>      | <code>do...while</code> |                       |

Операторы языка C с примерами приведены в **Приложении 4**. Общую информацию о синтаксисе и использовании языка C вы можете найти во множестве доступных источников.

## Типы файлов

В NL5 используется несколько специальных типов файлов. Каждый тип имеет определенное расширение и иконку.

| Иконка  | Расширение | Описание                                |
|---|------------|---|
|    | n15        | Схема.                                  |
|    | n15~       | Резервная копия схемы.                  |
|    | nlp        | Preferences (предпочтения).             |
|    | nlt        | Данные переходного процесса (бинарные). |
|   | nlf        | Данные АС (бинарные).                   |
|  | nll        | Файл лицензии.                          |

Если NL5 запускается из командной строки, один или более файлов разного типа могут быть использованы в качестве параметров. Например:

```
>n15.exe rc.n15           - загрузить схему из файла rc.n15
>n15.exe tran.nlt        - загрузить данные переходного процесса из файла tran.nlt
>n15.exe rc.n15 pref.nlp - загрузить схему из файла schematic rc.n15 и preferences из
                          файла pref.nlp
```

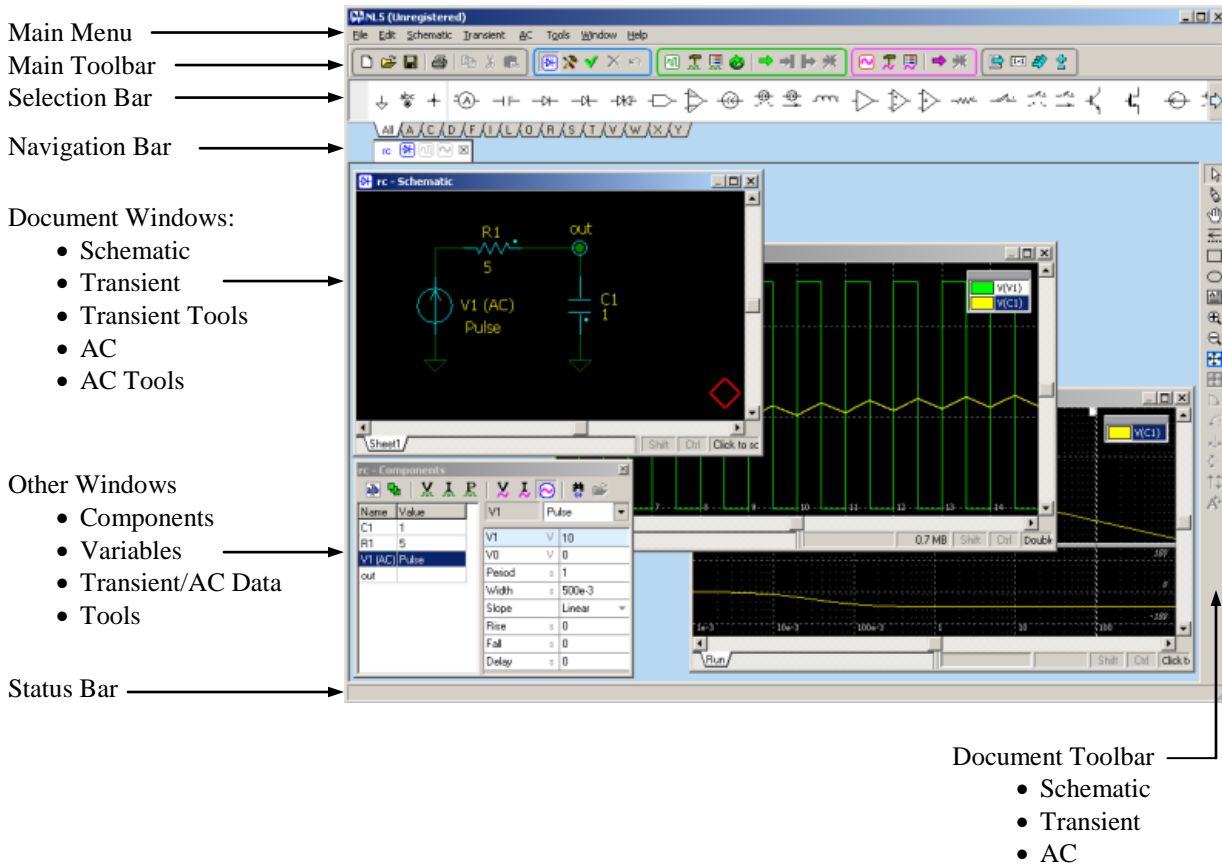
Файл с расширением “txt” в качестве параметра командной строки рассматривается как скрипт, и будет выполнен немедленно. Например:

```
>n15.exe script.txt      - загрузить и выполнить скрипт из файла script.txt
```

# Графический Интерфейс Пользователя

## Основное Окно

Основное окно NL5 и его компоненты показаны ниже:

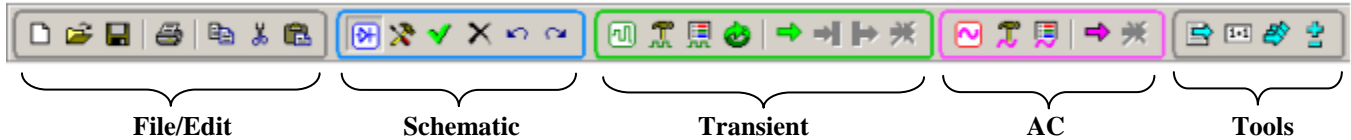


## Основное Меню (Main Menu)

Основное Меню содержит стандартный набор меню Windows (таких как **File**, **Edit**, **Window**, **Help**) и специфичные для NL5 (**Schematic**, **Transient**, **AC**, **Tools**).

## Основная инструментальная панель (Main Toolbar)

Основная инструментальная панель обеспечивает быстрый доступ к наиболее часто используемым командам и содержит пять групп кнопок:



Поместите курсор мышки поверх кнопки, и вы увидите подсказку с описанием кнопки.

## Строка состояния (Status Bar)

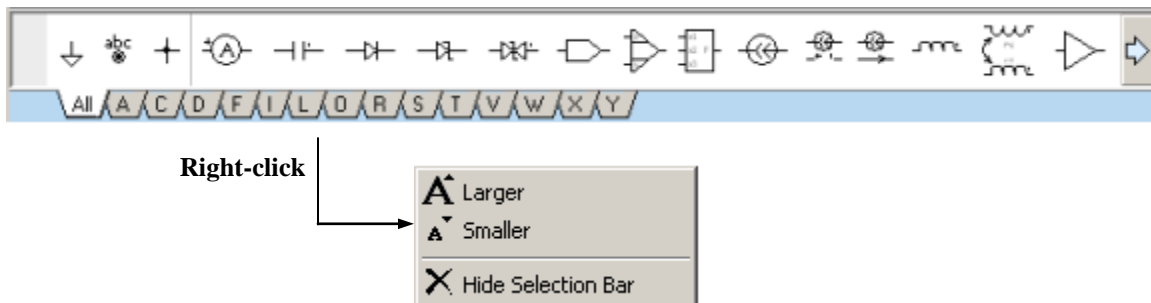
Строка состояния показывает некоторые сообщения, зависящие от состояния приложения, такие как:

- Загрузка документа.
- Сохранение документа.
- Проверка обновлений.

Выберите команду меню **Window | Status Bar**, чтобы показать/скрыть строку состояния.

## Панель выбора (Selection Bar)

Панель выбора состоит из закладок, по одной на букву (только если существует компонент на эту букву). Каждая закладка содержит символы компонентов с такой буквой и три общих элемента: землю, этикетку (label) и точку соединения. Закладка «**All**» (**Все**) содержит символы всех компонентов.



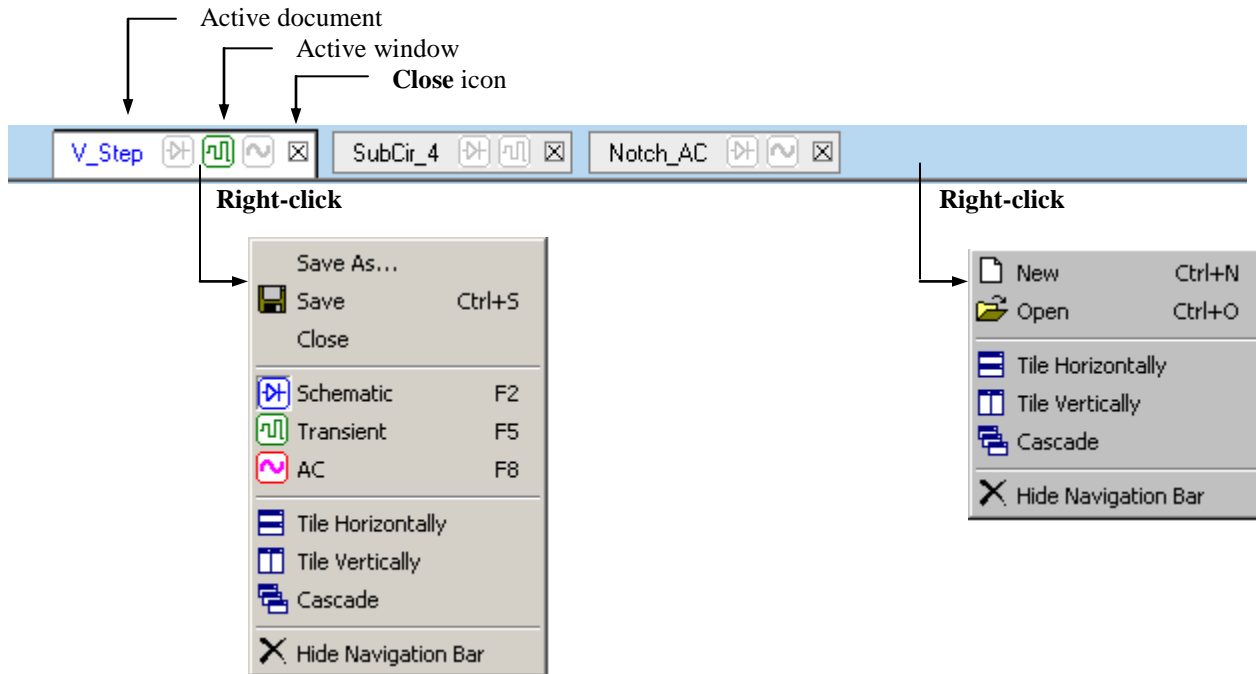
- Если не все компоненты видны, щелкните по изображению стрелки влево/вправо, чтобы прокрутить панель.
- Поместите указатель мышки поверх символа компонента, чтобы увидеть подсказку с кратким описанием компонента.
- Щелкните по символу, чтобы поместить компонент в схеме.



- Щелкните правой клавишей мышки по панели, чтобы увидеть всплывающее (контекстное) меню с соответствующими командами.
- Выберите команду меню **Select Window | Selection Bar**, чтобы показать/скрыть панель выбора.

## Панель навигации

Панель навигации отображает все открытые документы и окна, показывает активный документ и активное окно с подсвеченной иконкой.



- Если некоторые закладки не видны, щелкните по изображению стрелки влево/вправо, чтобы прокрутить панель.
- Поместите указатель мышки поверх иконки, чтобы увидеть подсказку.
- Щелкните по закладке, чтобы активизировать документ.
- Щелкните по иконке окна, чтобы активизировать окно.
- Щелкните по иконке **Close**, чтобы закрыть документ.
- Щелкните правой клавишей мышки по закладке документа или пустой области панели навигации, чтобы увидеть контекстное меню с соответствующими командами.
- Выберите команду основного меню **Select Window | Navigation Bar**, чтобы показать/скрыть панель навигации.

## Окна документа

Одновременно может быть открыто несколько окон документа NL5:

- Схема (Schematic)
- Переходный процесс (Transient)
- Инструменты анализа переходного процесса (Transient Tools)
- Частотная характеристика (AC)
- Инструменты анализа частотной характеристики (AC Tools)




Окна **Schematic**, **Transient**, и **AC Windows** являются частью стандартного многодокументного интерфейса и ведут себя в основном похоже на другие приложения Windows. Окно документа:

- Может быть минимизировано и максимизировано.
- Расположено внутри основного окна (**Window | Tile, Window | Cascade**).
- Перечислено в меню **Window**.
- Имеет соответствующую инструментальную панель (Document Toolbar), отображаемую справа в основном окне, когда это окно активизировано.
- Закрывание окна схемы автоматически закрывает весь документ.
- Используйте панель навигации или меню **Window** для активизации нужного окна и расположения их на экране.

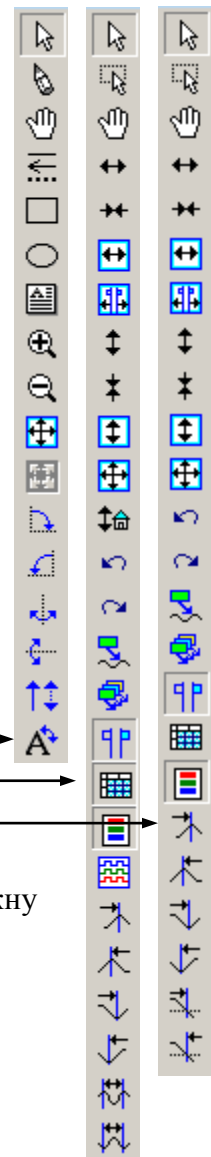
Окна инструментов **Transient Tools and AC Tools Windows** не являются частью многодокументного интерфейса и ведут себя иначе. Окна инструментов:

- Всегда отображаются поверх других окон.
- Перечислены в меню **Transient | Tools** и **AC | Tools**.

Инструментальные панели документа (**Document Toolbars**) обеспечивают быстрый доступ к часто используемым командам, относящимся к активному окну документа. Существует 3 типа панелей документа:

- Schematic toolbar 
- Transient toolbar 
- AC toolbar 

В любой момент времени видна только одна панель, относящаяся к активному окну документа.



## Другие окна

«Другие» окна не являются частью многодокументного интерфейса, однако они могут оставаться открытыми все время, и их не обязательно закрывать, чтобы перейти в другое окно. Эти окна всегда показывают информацию, относящуюся к текущему активному документу. Переключение между окнами документов автоматически обновляет информацию в окнах. К таким окнам относятся:


- Окно Компонентов (Components Window) (**Window | Components**, или **F3**)
- Окно Переменных (Variables Window) (**Window | Variables**, или **F4**)
- Данные Переходного процесса и АС (Transient and AC Data) (**Transient/Data, AC/Data**)
- Инструменты (Tools) (**Tools | Script, Tools | Sweep**, и другие)

## Окна диалогов

В отличие от просто «окон», диалоговое окно должно быть закрыто, чтобы вернуться в основное окно. Обычно окна диалогов имеют кнопки **OK** и **Cancel**, а некоторые и кнопку **Close**. Вот примеры диалоговых окон:

- Предпочтения (Preferences) (**Edit | Preferences**).
- Инструменты Схемы (Schematic Tools) (**Schematic | Tools**).
- Установки Переходного Процесса (Transient Settings) (**Transient | Settings**).
- ...и другие.

## Помощь (Help)

Файл Помощи (Help) NL5 `n15.chm` должен находиться в одной директории с файлом `n15.exe`. Файл содержит лишь краткую справочную информацию об операторах, функциях, командах скрипта, моделях и параметрах компонентов. За подробной информацией обращайтесь к этому Руководству. Чтобы открыть **Help**, выберите команду **Help | Help** Основного Меню. Для контекстной помощи нажмите клавишу **F1** или кнопку **Help** , которая доступна в некоторых окнах программы.

## Горячие клавиши

- **F1** – Помощь (Help)
- **F2** – Показать окно схемы
- **F3** – Показать/скрыть Окно Компонентов
- **F4** - Показать/скрыть Окно Переменных
- **F5** – Показать окно переходного процесса (transient)
- **F6** – Запустить расчет переходного процесса
- **F7** – Продолжить расчет переходного процесса
- **F8** - Показать окно частотной характеристики (АС)
- **F9** - Запустить расчет частотной характеристики (АС)

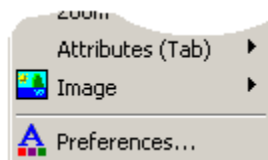
## Предпочтения (Preferences)

Предпочтения NL5 используются для индивидуальной настройки элементов пользовательского интерфейса, таких как шрифты, цвета, форматы, для предопределения параметров (default), обслуживания памяти и т.п. Предпочтения применяются ко всему приложению, а не только к отдельному документу (схеме). Предпочтения не влияют на результаты симуляции.

Предпочтения хранятся в файле с именем `n15.nlp`, находящимся в одной директории с `n15.exe`. Предпочтения сохраняются в файл каждый раз, когда нажимается кнопка **Apply** или **OK** в диалоговом окне **Preferences**, и при закрытии программы. При запуске NL5 загружает последние сохраненные предпочтения из файла.

Предпочтения можно сохранять и в индивидуальном файле предпочтений (с расширением «`nlp`»), а затем загружать их из файла. Эта возможность позволяет иметь разные профили для разных задач и возможность легкого переключения между ними.

Откройте диалоговое окно **Preferences** командой **Edit | Preferences** Основного Меню. Многие контекстные меню также имеют команду **Preferences**, обычно находящуюся внизу списка:



Выбор команды **Preferences** в контекстном меню обычно открывает диалоговое окно **Preferences** на странице, относящейся к данному месту программы (контексту).




Диалоговое окно **Preferences** состоит из нескольких страниц. Выбор страницы осуществляется щелчком по имени страницы на панели выбора страниц (слева). Когда любой из параметров меняется, активизируется кнопка **Apply**. Затем можно щелкнуть по кнопкам:

- **OK** – принять изменения и закрыть диалоговое окно.
- **Cancel** – отменить изменения и закрыть диалоговое окно.
- **Apply** – принять изменения и закрыть диалоговое окно.




## Preferences (предпочтения)

Сохранить/открыть предпочтения в/из файла, выбрать цветовую схему.

### Preferences

-  • **Save preferences.** Сохранить предпочтения в файл.
-  • **Open preferences.** Прочитать предпочтения из файла.
-  • **Reset preferences to default.** Установить предопределенные (default) предпочтения.

**Color scheme.** Цветовая схема применяется ко всем окнам документов (Schematic, Transient, Transient Tools, AC, AC Tools). Изменение цветовой схемы также изменит цвета имеющихся кривых переходного процесса и AC.

-  • **Color with black background.** Цветная графика на черном фоне.
-  • **Color with white background.** Цветная графика на белом фоне.
-  • **Black and white.** Черная графика на белом фоне. Эта цветовая схема может быть использована временно для сохранения черно-белого изображения схемы или графиков в файл или копирования в буфер обмена.

## Application (приложение)

Установка опций приложения.

- **Automatically check for updates (Автоматическая проверка обновлений).** Может быть установлена в диапазоне «Never»...«Every 90 days» («Никогда»...«Каждые 90 дней»). NL5 может автоматически проверять обновления на Web-сайте. NL5 не загружает и не устанавливает обновления: программа только сообщает о наличии доступных обновлений. Если на вашем компьютере активна антивирусная программа или/и защитный экран, от вас может потребоваться разрешение на доступ NL5 к Интернету. Если версия NL5, выпуск и редакция текущие, сообщение об этом появится в строке состояния. Если на Web-сайте обнаружена новая версия, будет открыто окно с информацией «release notes».
- **Most Recently Used files (недавно использованные файлы).** Может быть установлено в диапазоне 0...10. Это максимальное число недавно использованных файлов, отображаемых в меню **File**.
- **Subcircuit Library path.** Путь к директории “Subcircuit Library” (“Библиотека Подсхем”). Если файл подсхемы (модель **SubCir**) находится в этой директории, имя файла подсхемы может быть задано как «краткое»: без пути.
- **Beep on errors and messages.** Издавать звуковой сигнал при выводе окон ошибок или сообщений. Не распространяется на звуковые сигналы, генерируемые при выводе системных сообщений (такими как «файл не найден», «такой файл уже есть» и т.д.).

## Document (документ)

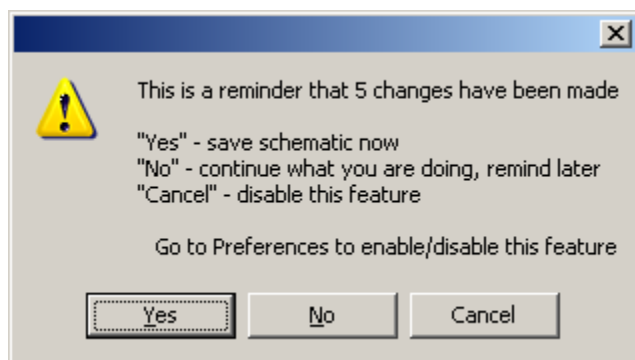
Установка predetermined свойств новой схемы и опций автосохранения/резервных копий.

**Properties (свойства).** Это информация по умолчанию будет установлена для свойств новой схемы. Чтобы увидеть и отредактировать свойства схемы, выберите команду **File | Properties**.

- **Author (автор).**
- **Organization (организация).**

### Autosave and backup (автосохранение и резервная копия)

- **Create backup when saving first time** (Создавать резервную копию при первом сохранении). Если схема была загружена из файла, отредактирована, а затем впервые сохраняется, файл, из которого схема была загружена, будет конвертирован в файл резервной копии с расширением «nl5~». Это предохраняет от случайной потери оригинала при ошибочной перезаписи файла.
- **Save automatically when analysis starts** (Автоматически сохранять, когда запускается анализ). Если выбрано, NL5 автоматически сохраняет схему при каждом запуске анализа переходного процесса или АС.
- **Show a reminder when NNN changes are made** (Показывать напоминание, когда сделано NNN изменений). Если выбрано, NL5 будет показывать напоминание после заданного количества сделанных изменений схемы:



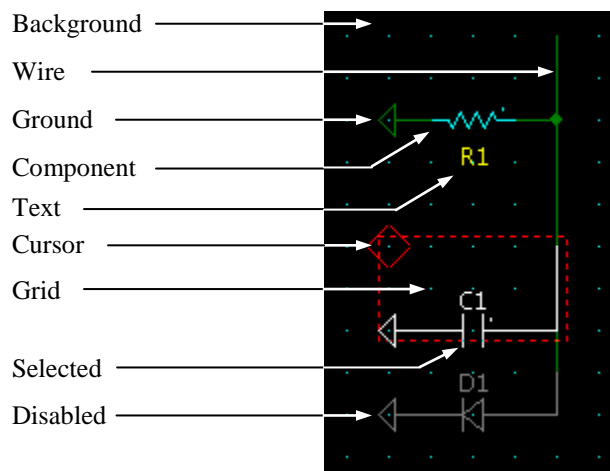
### Undo (отмена)

- **Max number of Undo steps** (максимальное количество шагов отмены). Хотя размер буфера отмены неограничен, может быть задано максимальное количество запоминаемых операций. Если в буфер добавляется новая информация, а размер буфера превышает заданную величину, самые старые данные будут выброшены из буфера.
- **Clear Undo buffer on schematic save** (очищать буфер отмены при сохранении схемы). Если выбрано, буфер отмены будет очищен при сохранении схемы в файл. Иначе будут помниться все операции с момента создания схемы или загрузки схемы из файла.

## Schematic (схема)

Устанавливает свойства отображения схемы. Новые свойства будут применены ко всем новым и существующим элементам схемы, за исключением элементов с пользовательскими (отформатированными) свойствами.

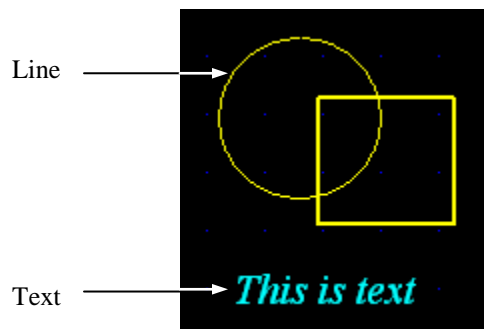
- A** • **Font size.** Установить размер шрифта имени и значения компонента.
- abc** **A** • **Font.** Выбрать шрифт имени и значения компонента.
- ≡** • **Line width.** Установить ширину линии проводников и изображения компонентов.
- >** • **Cursor width.** Установить ширину линии изображения курсора.
- **Attributes grid.** Выбрать размер сетки для размещения атрибутов компонента (относительно размера сетки схемы).
- ⋮** • **Show grid points.** Если выбрано, показывать точки сетки схемы.
- 123** • **Show node numbers.** Если выбрано, показывать номера узлов схемы.
- 📄** • **Show hints.** Если выбрано, показывать имя компонента, модель и параметры в окне подсказки, когда указатель мышки расположен на компоненте.
- **Colors.** Дважды щелкните по пункту в списке, чтобы изменить цвет:



## Drawings (рисунки)

Установка свойств новых «рисунков» (текст, линии, прямоугольники, овалы). Свойства будут применены только к новым рисункам.

- A** • **Font size.** Установить размер шрифта текста.
- abc** **A** • **Font.** Выбрать шрифт текста.
- ≡** • **Line width.** Установить ширину линии для картинок.
- **Drawings grid.** Выбрать размер сетки для размещения картинок (относительно размера сетки схемы)..
- **Colors.** Дважды щелкните по пункту в списке, чтобы изменить цвет:



## Mouse (Schematic) (мышь, схема)

Установить функции колесика мышки.

**Mouse wheel action.** Выбрать функцию колесика мышки при вращении, в зависимости от состояния клавиш **Ctrl** и **Shift**.

- **no key held** - только вращение колесика, клавиши не нажаты.
- **Ctrl** - вращение колесика мышки при нажатой клавише **Ctrl**.
- **Shift** - вращение колесика при нажатой клавише **Shift**.
- **Ctrl+Shift** - вращение колесика при нажатых клавишах **Ctrl** и **Shift**.

Выбрать функцию из:

- **None** – ничего
- **Zoom** – уменьшение/увеличение
- **Hor scroll** – горизонтальная прокрутка
- **Vert scroll** – вертикальная прокрутка
- **Invert Zoom** – инверсия операций Zoom In/Out (увеличение, уменьшение).



## Components (компоненты)

Установка свойств окна компонентов и окна переменных, и параметров по умолчанию для новых компонентов

### Components and variables window (окна компонентов и переменных).

- A** • **Font size.** Установить размер шрифта.
- A** <sup>abc</sup> • **Font.** Выбрать шрифт.
- **Show units with NNN color.** Выбрать цвет для отображения единиц параметра в окне компонентов:
  - **None** – не показывать единицы
  - **Grey** – показывать серым цветом
  - **Black** – показывать черным цветом

**New component.** Установка значения по умолчанию нового компонента:

- **R, Ohm** – резистор
- **C, F** – конденсатор
- **L, H** – индуктивность
- **Vd (Diode), V** – прямое падение напряжения на диоде

## Symbols (символы компонентов)

Выбрать символ для изображения некоторых типов компонентов.

- **Voltage source.** Источники напряжения (включая управляемые)
- **Current source.** Источники тока (включая управляемые).
- **Controlled source.** Управляемые источники напряжения и тока.
- **Resistor.** Резистор, потенциометр, управляемые резисторы.

## Warnings (предупреждения)

Выбор предупреждений показываемых при проверке схемы.

- **Warnings.** Снимите флажок, чтобы убрать все предупреждения.
- **Floating pins.** Проверка любых «плавающих» выводов компонента.
- **Non-connected components.** Проверка компонентов со всеми не присоединенными выводами.
- **Overlapping components and wires.** Проверка перекрытия компонента другим компонентом и компонента проводником.
- **Possibly floating schematic.** Проверка, имеет ли схема хотя бы одну землю или этикетку (label) с моделью источника напряжения.

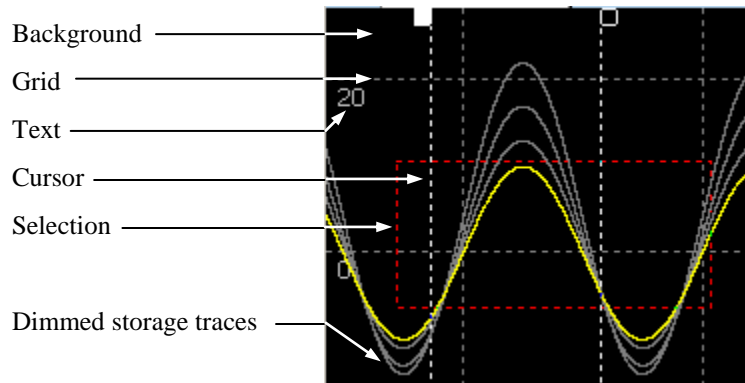
## Graphs (графики)

Установка свойств окон Transient, Transient Tools, AC, и AC Tools.

- A** • **Font size.** Установить размер шрифта разметки осей.
- abc A** • **Font.** Выбрать шрифт разметки осей.
- ≡** • **Default trace width.** Установить ширину линии новых кривых.
- ≡** • **Markers width.** Установить ширину маркеров.

### Gridlines interval (pixels) (интервал линий сетки, в пикселях)

- ⋮** • **Vertical gridlines.** Установить приблизительный интервал между вертикальными линиями (в пикселях).
- ⋮** • **Horizontal gridlines.** Установить приблизительный интервал между горизонтальными линиями (в пикселях).
- **Numbers alignment.** Установить положение чисел вертикальной разметки.
- **Colors.** Дважды щелкните по пункту в списке, чтобы изменить цвет:



## Table (таблица)

Установка свойств таблицы данных переходного процесса и АС анализа.

**Text.** Устанавливает свойства текста в таблице (кроме фазы в таблице данных АС).

- A** • **Font size.** Установить размер шрифта текста.
- A** <sup>abc</sup> • **Font.** Выбрать шрифт текста.

**Phase.** Устанавливает свойства текста фазы в таблице данных АС.

- A** • **Font size.** Установить размер шрифта текста.
- A** <sup>abc</sup> • **Font.** Выбрать шрифт текста.
- **Alignment.** Установить положение (выравнивание) текста.

**Significant digits.** Задаёт количество значащих цифр для данных в таблице и Markers Tool (маркеры).

- **Time/Frequency.** Время/частота.
- **Data.** Данные.
- **Colors.** Дважды щелкните по пункту в списке, чтобы изменить цвет:

|                  |         | left       |
|------------------|---------|------------|
|                  | Cursors | 319.702e-3 |
| Text             | V(C42)  | -83.5759   |
| Phase text       |         | 0          |
| Phase background | V(C37)  | -61.2802   |
|                  |         | -180       |

## Legend (условные обозначения, список кривых на графике)

Установка свойств окна Legend (списка кривых на графике).

- A** • **Legend font size.** Установить размер шрифта текста.
- **Max width (максимальная ширина).** Если не выбрано, то ширина окна Legend будет автоматически подстраиваться, чтобы имена всех кривых были показаны полностью. Если выбрано, то ширина окна будет также подстраиваться, но только до тех пор, пока не превысит заданное значение ( $NNN * \text{высоту шрифта текста}$ ).

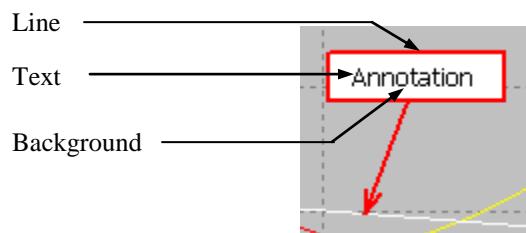
## Annotation (аннотация)

Устанавливает свойства аннотаций в окнах переходного процесса и АС анализа.

- A** • **Font size.** Установить размер шрифта текста.
- abc** **A** • **Font.** Выбрать шрифт текста.
- ≡** • **Line width.** Установить ширину линии для указателя и рамки.
  - **Arrow.** Если выбрано, рисовать стрелку указателя.
  - **Draw line with trace color.** Если выбрано, для указателя и рамки используется цвет кривой.
  - **Draw text with trace color.** Если выбрано, используется цвет кривой для текста.

**Significant digits.** Задаёт количество значащих цифр для отображаемого времени, частоты и данных.

- **Time/Frequency.** Выводить время/частоту.
- **Data.** Выводить данные.
- **Colors.** Дважды щелкните по пункту в списке, чтобы изменить цвет:



## Text (текст)

Устанавливает свойства текста в окнах переходного процесса и АС анализа.

- A** • **Font size.** Установить размер шрифта текста.
- abc** **A** • **Font.** Выбрать шрифт текста.
- ≡** • **Line width.** Установить ширину линии для указателя и рамки.
  - **Colors.** Дважды щелкните по пункту в списке, чтобы изменить цвет:



## Mouse (Graphs) (мышь, графики)

Установить функции колесика мышки.

**Mouse wheel action.** Выбрать функцию колесика мышки при вращении, в зависимости от состояния клавиш **Ctrl** и **Shift**.

- **no key held** – только вращение колесика, клавиши не нажаты.
- **Ctrl** – вращение колесика мышки при нажатой клавише **Ctrl**.
- **Shift** – вращение колесика при нажатой клавише **Shift**.
- **Ctrl+Shift** – вращение колесика при нажатых клавишах **Ctrl** и **Shift**.

Select action from:

- **None** – ничего
- **Zoom** - уменьшение/увеличение по горизонтали и вертикали одновременно
- **Hor zoom** - уменьшение/увеличение по горизонтали
- **Vert zoom** - уменьшение/увеличение по вертикали
- **Hor scroll** - горизонтальная прокрутка
- **Vert scroll** - вертикальная прокрутка
  
- **Invert Zoom** – инверсия операций Zoom In/Out (увеличение, уменьшение)

## Transient (переходный процесс)

Установка опций симуляции переходного процесса и памяти.

- **Status update interval, ms.** Обновлять информацию о статусе переходного процесса с заданным интервалом.

### Memory (память).

- **Max memory per trace, MB.** Установка максимального размера памяти, отведенной для одной кривой. Если лимит памяти исчерпан, небольшая начальная часть кривой будет удалена и появится предупреждающее сообщение на панели статуса переходного процесса «*One or more traces have been truncated*» (одна или более кривых были усечены).
- **Max memory per delay and transmission line components, MB (warning).** Установка максимального размера памяти, отведенной для компонента «задержка» (delay) и компонентов линии передачи. Если расчетная требуемая память превышает заданный лимит, появится предупреждающее сообщение с опциями «продолжить» или «остановить симуляцию».

### Export traces (экспорт кривых).

- **Approximate number of points (приблизительное количество точек).** Когда открывается диалоговое окно **Transient Export/View**, значение шага времени в таблице выбирается автоматически таким образом, что количество точек в таблице приблизительно равно этому числу.


- **Max number of points (максимальное количество точек).** Количество точек для **Transient Export/View** не может превышать этого значения и будет автоматически ограничено. Также это значение определяет максимальное количество точек, передаваемое в команде скрипта “*tracename*”.

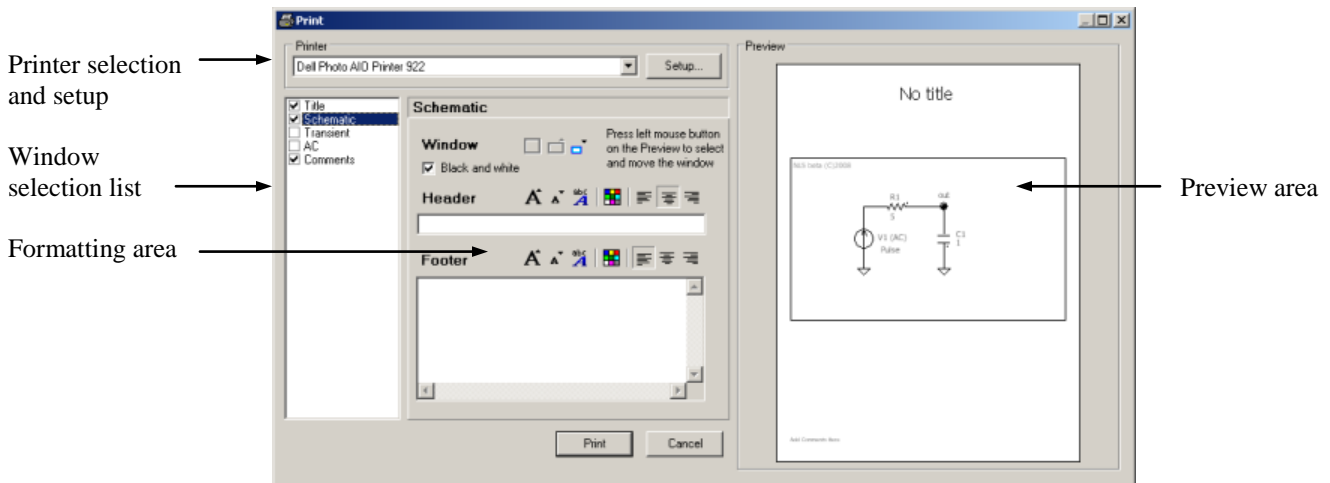
## HTTP Server

Установка опций HTTP Сервера.

- **Max number of log lines.** Введите максимальное число строк лога, или выберите из выпадающего списка. Когда количество строк превысит заданное, первые строчки будут стираться.

## Печать

Щелкните по **Preview and print** кнопке инструментальной панели  или выберите команду **File | Preview** в Основном Меню, чтобы открыть диалоговое окно **Print (Печать)**. Типичный вид диалогового окна и его основных компонентов показан ниже:



- Выберите принтер из выпадающего списка. Щелкните по кнопке **Setup**, чтобы сделать установки принтера.
- Выберите окна, которые будут распечатаны, в списке выбора окон. Список содержит: Title (заголовок), Comments (комментарии) и окна активного документа, доступные для печати (показанные на экране в данный момент):
  - **Schematic (схема).**
  - **Transient (переходный процесс).**
  - **Transient Tools (инструменты анализа переходного процесса).**
  - **AC (частотная характеристика).**
  - **AC tools (инструменты анализа частотной характеристики).**

**Обратите внимание:** если окно схемы, переходного процесса или анализа AC максимизировано, тогда только это окно будет доступно для печати. В этой ситуации появится предупреждающее сообщение ниже списка выбора.

- Выделите имя окна в списке, чтобы задать формат печати этого окна в области форматирования (Formatting area).
- Задайте расположение окна в области **Preview**. Щелкните по изображению окна для его выбора. Щелкните по изображению окна и перетащите его, чтобы переместить окно на листе
- Нажмите кнопку **Print** для печати или **Cancel**, чтобы закрыть диалоговое окно.

## Форматирование печати

Обычно при печати заголовок отображается в верхней части листа, а комментарии в нижней. Однако они могут быть перемещены в любое место на листе. Окна могут отображаться в любом месте листа, размер окна выбирается в области форматирования окна. Окна могут иметь индивидуальный заголовок (**Header**) и сноску (**Footer**). Заголовок – это одна строка текста, задаваемая в окне Header. Сноска может иметь много строк, и задается в разделе Footer.

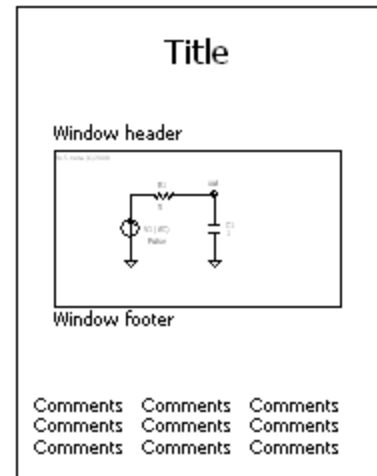
Доступны следующие кнопки форматирования

### Window (окно)

- **Maximize.** Максимизировать.
- **Larger.** Увеличить.
- **Smaller.** Уменьшить.
- **Black and white.** Печатать в черно-белом формате.

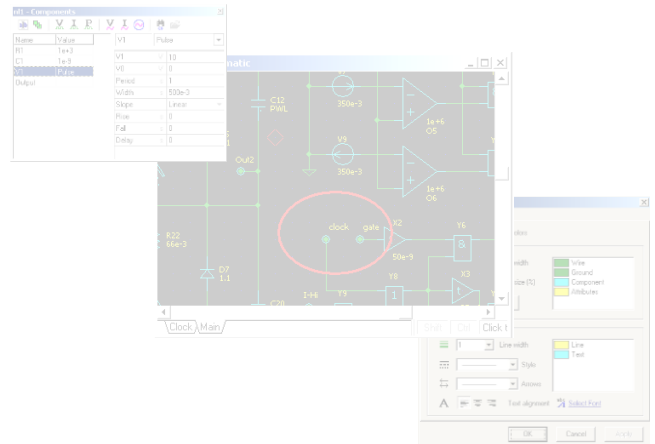
### Title, Comments, Header and Footer

- A** • **Larger font.** Увеличить шрифт.
- A** • **Smaller font.** Уменьшить шрифт.
- A** • **Select font.** Выбрать шрифт.
- A** • **Select color.** Выбрать цвет.
- ≡** • **Align left.** Выравнивание по левому краю.
- ≡** • **Center.** Выравнивание по центру.
- ≡** • **Align right.** Выравнивание по правому краю.





# III. Схема



Ниже приведена упрощенная диаграмма работы со схемой:

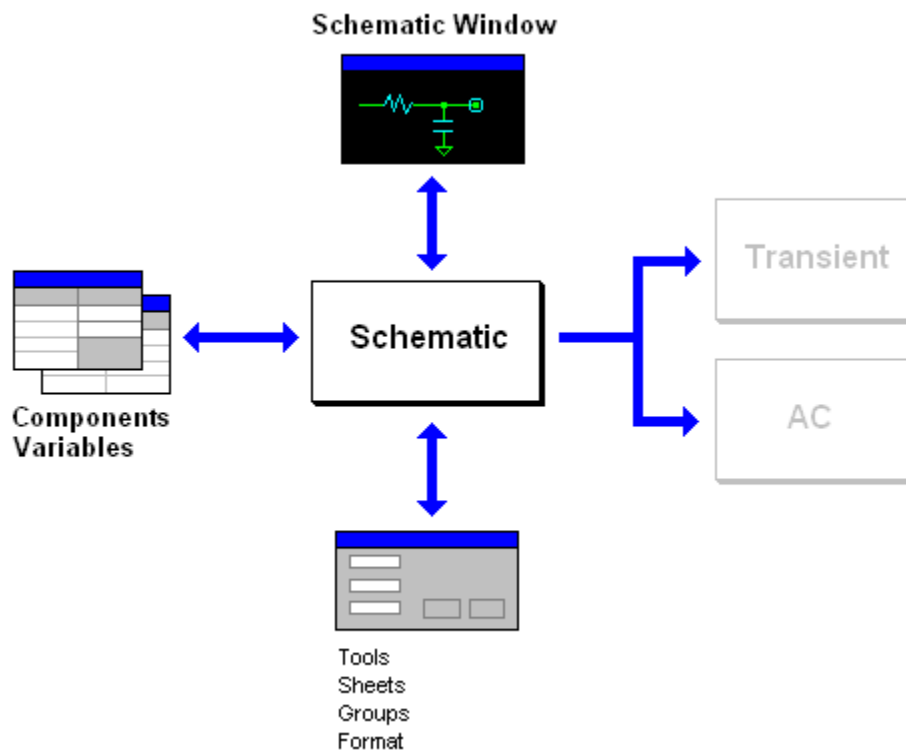
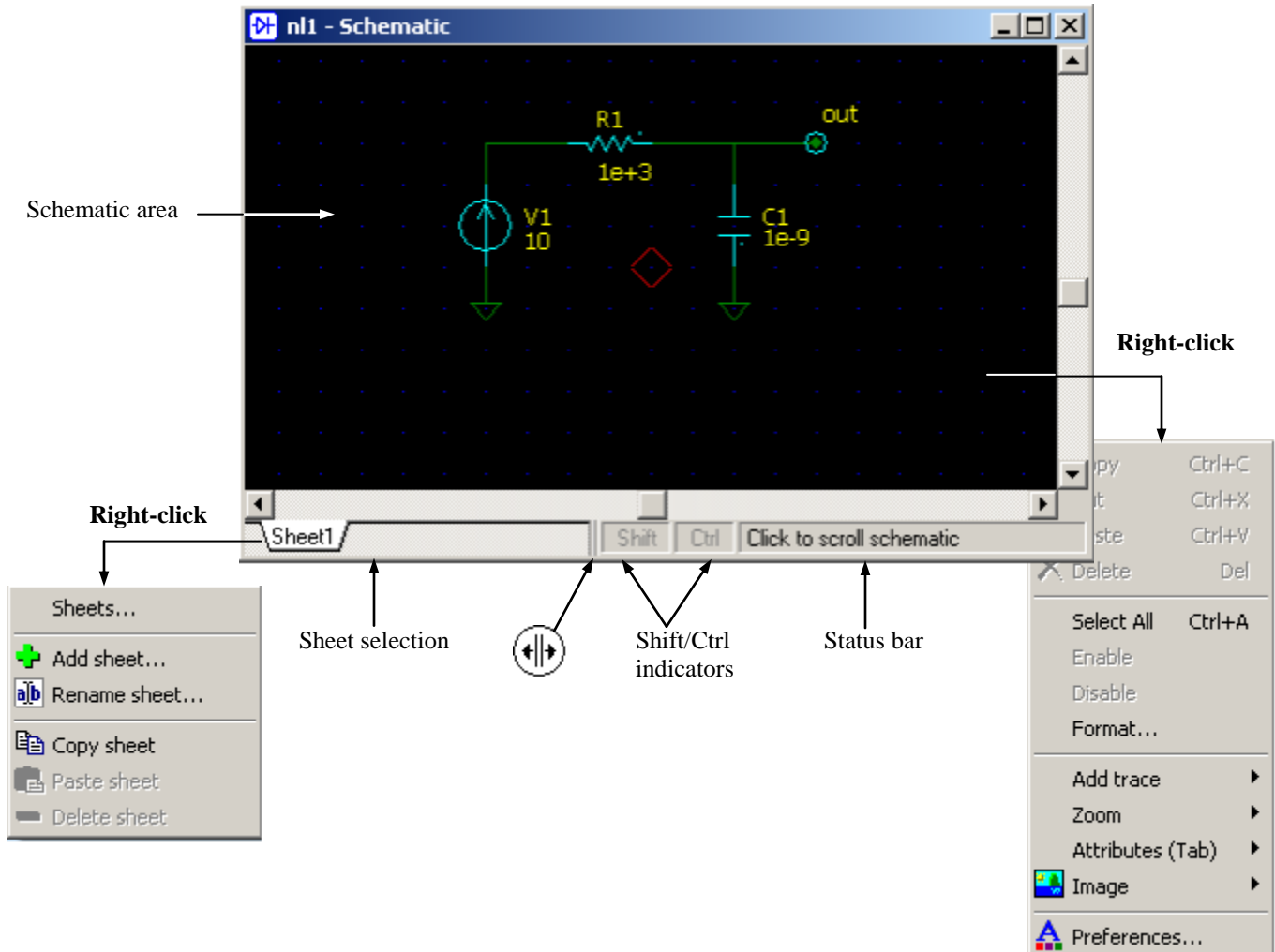



Схема отображается и может быть отредактирована в **Schematic window (окне схемы)**. Любой документ должен иметь окно схемы, закрывание окна схемы автоматически закрывает весь документ. Компоненты и переменные отображаются и могут редактироваться в **Components Window (окне компонентов)** и **Variables Window (окне переменных)**. Несколько диалоговых окон, таких как **Tools (инструменты)**, **Sheets (листы)**, **Groups (группы)**, **Format (формат)**, используются для выполнения других операций над схемой и окном схемы. Данные схемы используются для анализа переходного процесса и вычисления частотной характеристики (Transient и AC).

## Окно схемы

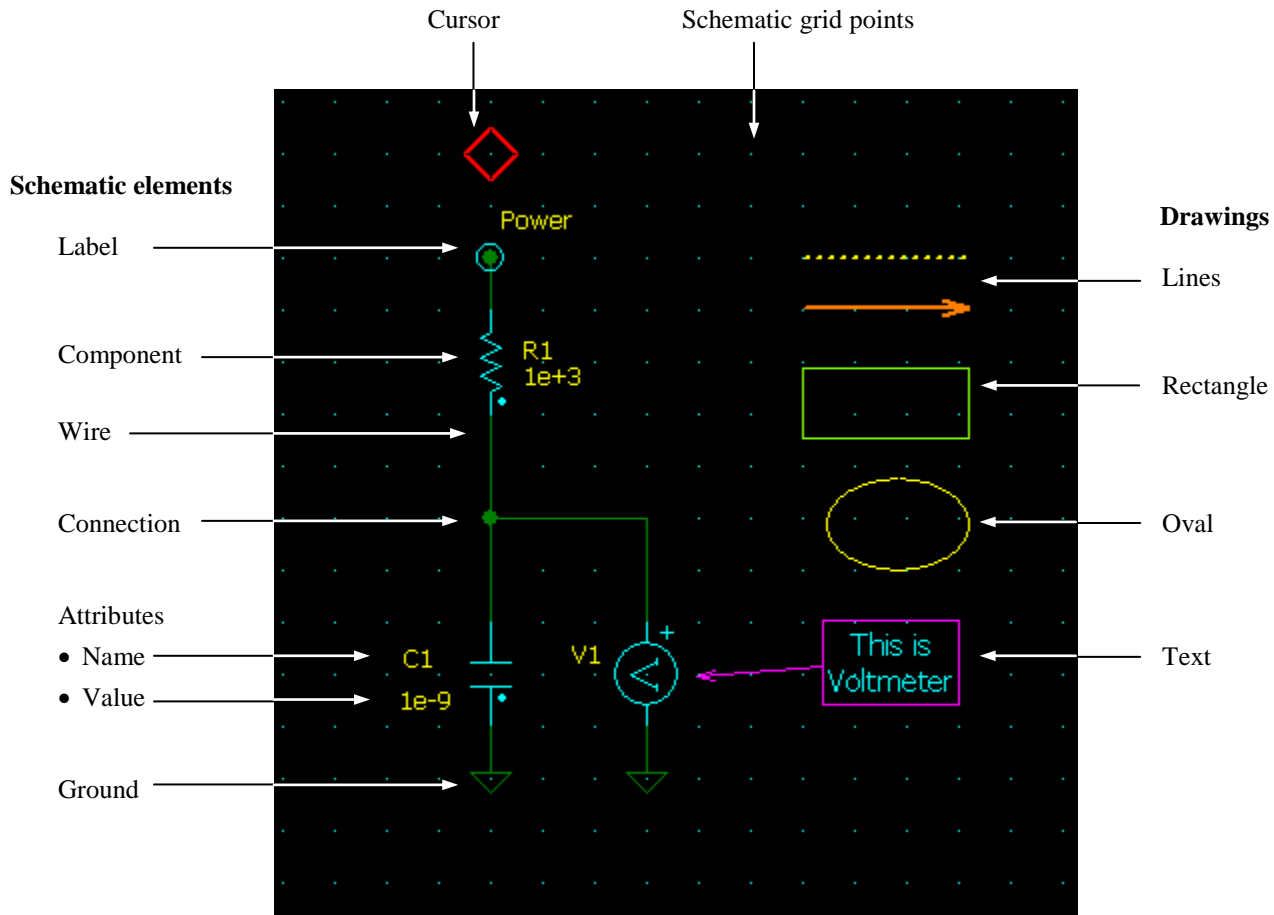
Типичный вид окна схемы и его основных компонентов показан ниже:



- Схема показана в области схемы (**Schematic area**).
- Область выбор листа (**Sheet selection**) содержит закладки листов. Щелкните по закладке для выбора листа.
- Щелчок правой кнопкой мышки по области выбора листа вызывает контекстное меню с соответствующими командами.
- Индикаторы **Shift/Ctrl** подсвечиваются, когда клавиша **Shift** и/или **Ctrl** нажаты.
- Строка состояния (**Status bar**) показывает подсказки, относящиеся к текущей позиции указателя мышки и состоянию **Shift/Ctrl**.
- Поместите указатель мышки поверх области разделителя (“splitter”) , затем нажмите левую клавишу мышки и перетащите его для изменения размера области выбора листа .

- Щелкните правой клавишей мышки по схеме, чтобы увидеть контекстное меню с соответствующими командами.

Область схемы содержит элементы схемы, рисунки, точки сетки и курсор.



- Элементы схемы** включают в себя: проводник, соединение, землю, этикетку (label) и компонент. Элементы схемы описывают «электрическую» часть схемы, используемую для симуляции.
- Рисунки (Drawings)** включают в себя: линию, прямоугольник, овал и текст. Рисунки применяются для добавления комментариев и примечаний.
- Точки сетки (Grid points)** — это опорные точки для курсора и элементов схемы.
- Курсор (Cursor)** используется для размещения/выделения элементов схемы и может располагаться только на точках сетки.

Большая часть команд редактирования применима и к элементам схемы, и к рисункам. Если специально не оговорено, слово «рисунки» опускается в описании этих команд

Все элементы изначально размещаются на схеме с предустановленными свойствами (цвет, ширина, стиль, шрифт и т.д.), заданными на странице **Schematic** диалогового окна **Preferences**.







Свойства любого элемента могут меняться пользователем с помощью форматирования. Изменение predetermined свойств элементов схемы сказывается на всех существующих элементах схемы, исключая элементы со свойствами, определенными пользователем («форматированные»). Изменение predetermined свойств рисунков не сказывается на существующих рисунках

## Редактирование схемы

Редактирование схем и навигация могут быть выполнены командами доступными в Основном Меню, контекстных меню, на основной инструментальной панели, инструментальной панели схемы, с помощью горячих клавиш, клавиш клавиатуры и мышки. NL5 поддерживает много команд и горячих клавиш, которые обычно используются в приложениях Windows (таких как **Edit | Copy (Ctrl-C)**, **Edit | Paste (Ctrl-V)** и т.д.), использует клавишу **Ctrl** с мышкой для операций выделения/копирования, использует полосы прокрутки и т.п. Другие команды настолько интуитивно понятны, что начало работы со схемой не займет много времени.

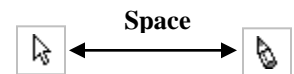
Очень часто одни и те же операции могут быть выполнены разными способами. Например, выбор и размещение нового компонента в схеме может быть сделано с использованием только клавиатуры, только мышки или с помощью того и другого. Пользователю предлагается самому выбрать наиболее эффективный и подходящий для него способ. Полный список команд прилагается.

Есть 6 режимов работы с редактированием схемы:

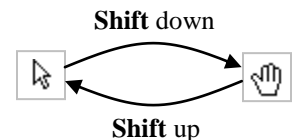
-  • **Selection.** Выделение (Выбрать элемент, блок; поставить компонент.)
-  • **Wire.** Соединение (Нарисовать проводник, поставить компонент.)
-  • **Scrolling.** Прокрутка схемы.
-  • **Line.** Рисование линии.
-  • **Rectangle.** Рисование прямоугольника или квадрата.
-  • **Oval.** Рисование овала или круга.

Режим редактирования может быть выбран щелчком по кнопке инструментальной панели схемы. Также есть несколько быстрых способов переключения между часто используемыми режимами:

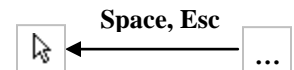
- Нажмите пробел (**Space**) для переключения между режимами **Selection** и **Wire**:



- Установите указатель мышки на пустое место, нажмите и удерживайте **Shift**, щелкните и перетащите мышку для прокрутки схемы. Отпустите **Shift**, чтобы вернуться в режим **Selection**:



- Нажмите **Esc** или пробел (**Space**) в любом режиме для переключения в режим **Selection**:



## Курсор

Курсор используется как маркер при размещении нового элемента схемы: проводника, земли, соединения, компонента или этикетки. Курсор работает в двух режимах, *Selection* (выделение) и *Wire* (соединение):

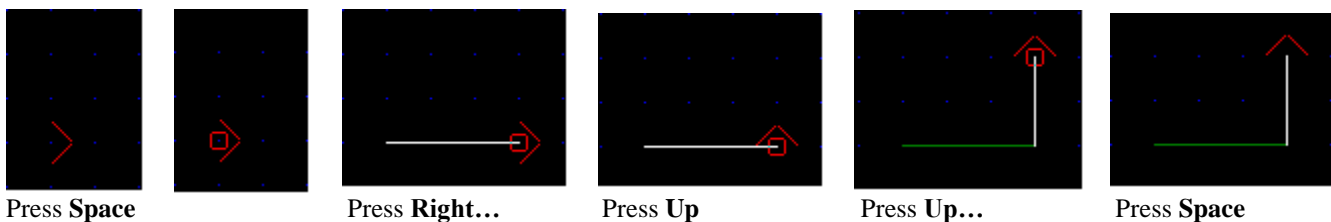




- Используйте клавиши **Left, Up, Right, Down** курсорной части клавиатуры для изменения направления и перемещения курсора.
- Чтобы изменить направление курсора с помощью мышки, щелкните рядом с углом, указывающим на новое направление.
- Щелкните по схеме, чтобы переместить курсор в новую точку.
- Нажмите **Home**, чтобы отцентровать курсор на экране.
- Нажмите пробел (**Space**) для переключения между режимами *Selection* и *Wire*.
- В режиме *Selection* поместите курсор на элемент для его выделения.
- В режиме *Wire* переместите курсор, чтобы нарисовать новый проводник.
- Когда курсор достигает края окна схемы, окно прокручивается автоматически.

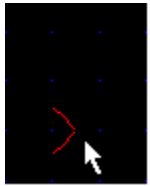
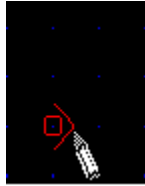
## Проводник

Следующие примеры показывают, как нарисовать проводник, используя клавиатуру и мышь.

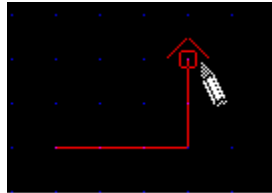
**Клавиатура.** Поместите курсор в начальную точку, используя клавиши (**Left, Up, Right, Down**). Нажмите пробел (**Space**), чтобы переключиться в режим *Wire*, затем перемещайте курсор с помощью стрелок клавиатуры. Новый проводник появится в режиме выделения. Чтобы завершить «прокладку» проводника, нажмите пробел (**Space**), этим вы переключитесь в режим *Selection*, или поменяйте направление курсора и продолжайте проводник в другом направлении:



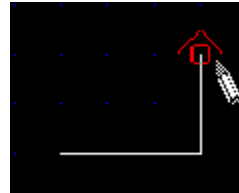
**Мышка.** Щелкните по кнопке **Wire**  для переключения в режим соединения **Wire**. Щелкните по начальной точке соединения, удержите клавишу мышки и перетащите курсор к конечной точке соединения, где и отпустите клавишу мышки. Вы можете проводить два ортогональных отрезка проводника за один раз. Щелкните по кнопке **Selection** , чтобы вернуться в режим **Selection**:

Click **Wire** button

Click and drag right, then up

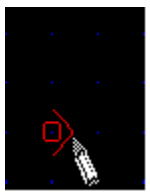
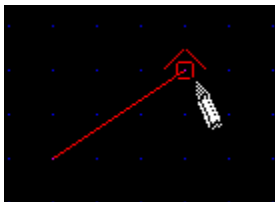


Release left button

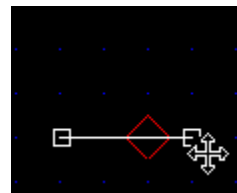
Click **Selection** button

При проведении соединений вы можете использовать совместно клавиатуру и мышь. Например, использовать клавишу пробел (**Space**) для переключения между режимами **Selection** и **Wire**, и использовать мышь для рисования проводников.

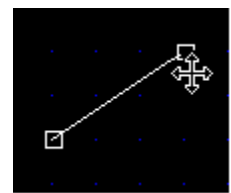
Для проведения диагональных проводников удерживайте клавишу **Ctrl**, пока рисуете, затем отпустите клавишу мышки. Другой способ провести диагональный проводник — выбрать существующий проводник, затем щелкнуть и переместить конец проводника:

Hold **Ctrl** key, click and drag


Click to select

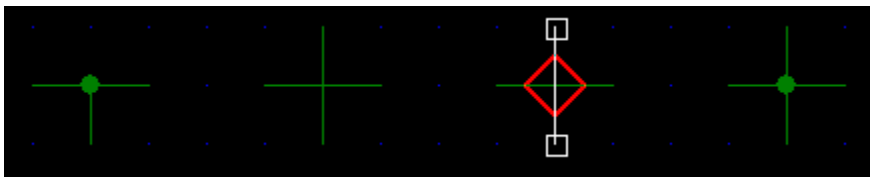


Click and drag the end

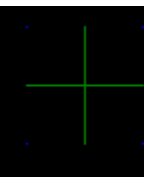


## Соединение (Connection)

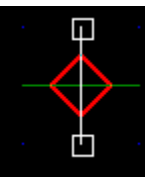
Три проводника, приходящие в одну точку, всегда соединены. Точка соединения автоматически устанавливается в процессе проверки схемы. Два перекрещивающихся проводника не соединяются по умолчанию. Чтобы соединить эти проводники, добавьте точку соединения: поместите курсор на пересечение и нажмите клавишу «.» (точка), или поместите курсор на пересечение и щелкните на панели выбора (**Selection Bar**) по иконке **Connection** (  ).



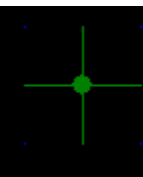
Automatic connection



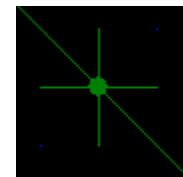
No connection



Press '.'



Connection


**Diagonal wire not connected!**



Все ненужные точки соединения будут автоматически удалены в процессе проверки схемы.

**Внимание:** *диагональные проводники могут быть не соединены с другими, даже если точка соединения помещена на пересечении. Постарайтесь избежать соединения двух диагональных проводников в одной точке.*

## Земля

Для размещения земли нажмите клавишу 'G' или щелкните по иконке земли  на панели выбора (**Selection Bar**). Земля является общей для всего документа, включая все листы и все подсхемы.

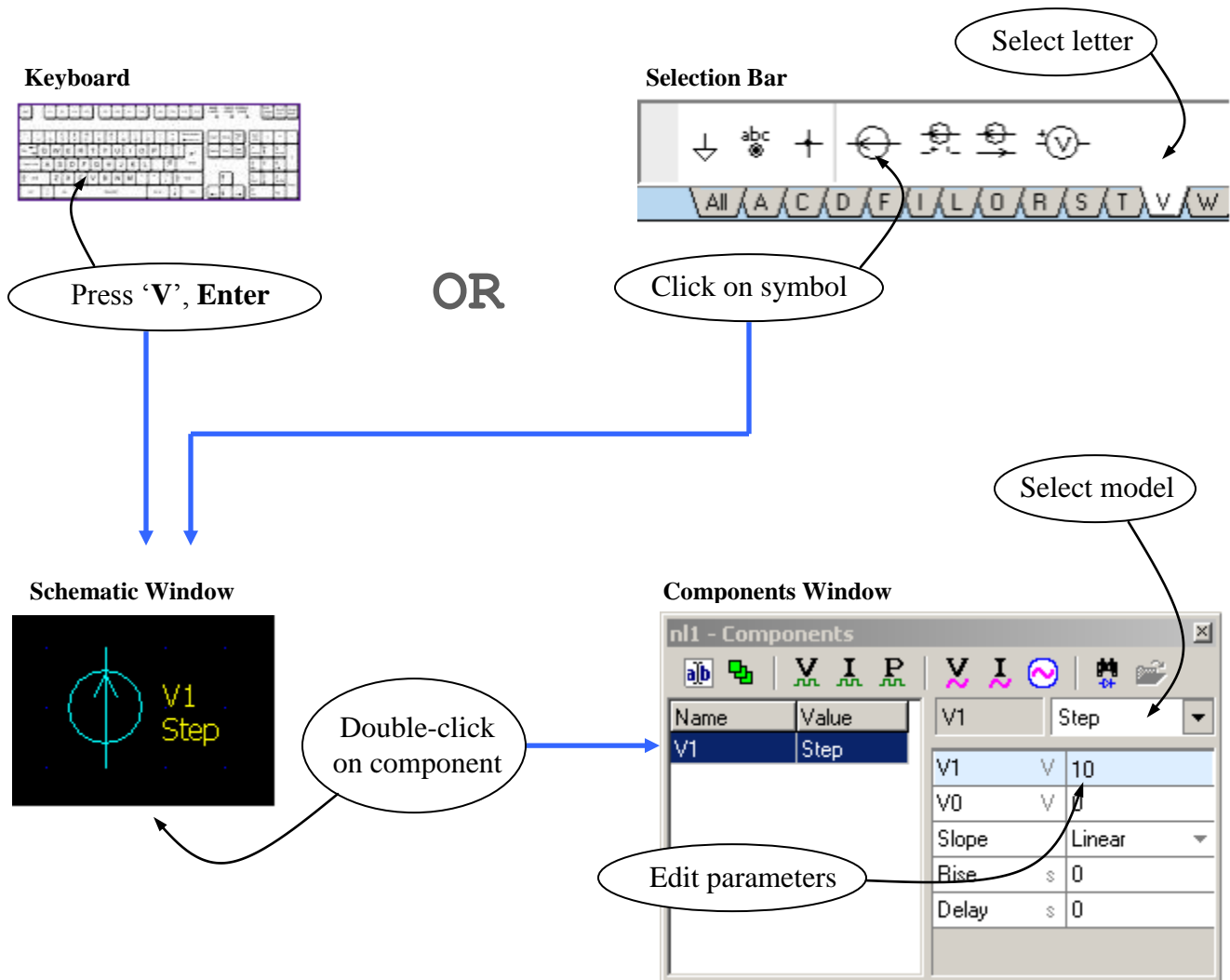
## Компонент

Каждый тип компонента имеет назначенную букву (**letter**) и символ (**symbol**). Например, все типы компонентов с буквой **S** — это ключи. Панель выбора (**Selection Bar**) имеет закладку, относящуюся к каждой букве. Когда при размещении компонента используется клавиатура, буква на клавиатуре используется для выбора типа компонента. При установке имя компонента начинается с этой буквы.

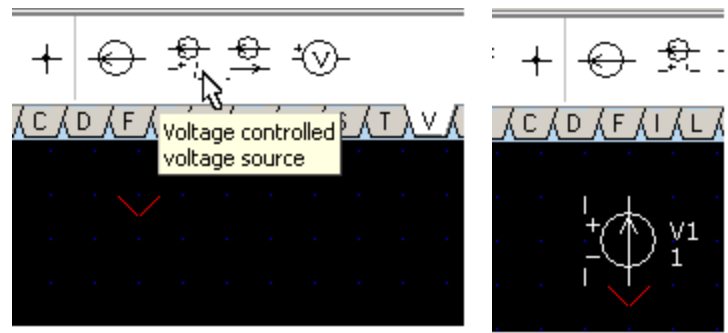
**Символ** — это изображение типа компонента: то, как компонент отображается в схеме. Панель выбора (**Selection Bar**) содержит символы всех доступных типов компонентов.

Некоторые компоненты являются “customized” (конфигурируемые пользователем): изображение этих компонентов (размер компонента, количество, положение и названия выводов) может быть сконфигурировано в диалоговом окне **Edit Component**. Диалоговое окно откроется автоматически в момент постановки компонента, а также может быть открыто в любое время после того, как компонент установлен. См. раздел *Редактирование конфигурируемого компонента* для подробностей.

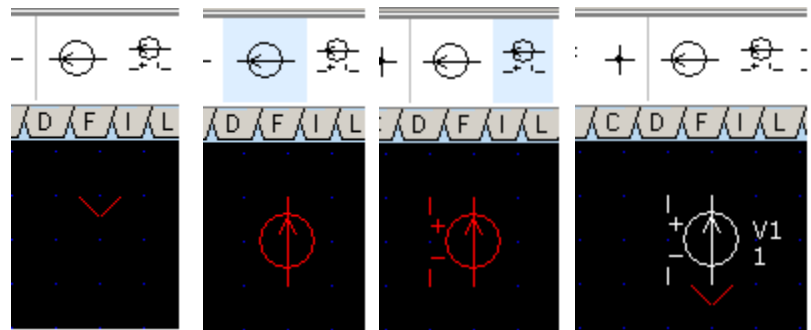
Следующая диаграмма и пример показывают процесс размещения компонента.



**Панель выбора (Selection Bar).** Выберите закладку с требуемым типом компонента (V), затем щелкните по символу компонента, чтобы поместить его («Voltage controlled voltage source»). Новый компонент появится в выделенном виде, так что вы можете сразу повернуть его, отразить, перевернуть или выбрать нужный вид.

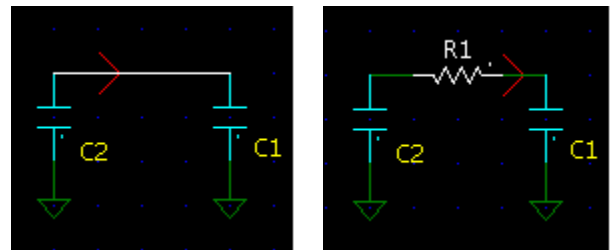


**Клавиатура.** Нажмите клавишу с буквой нужного типа компонента (V). Если несколько типов компонентов доступны для этой буквы, нажмите клавишу с буквой вновь, пока нужный тип компонента не появится. Используйте клавиши стрелок для перемещения нового компонента или щелкните и перетащите новый компонент мышкой. Чтобы поместить компонент, нажмите **Enter**. Чтобы отказаться, нажмите **Delete** или **Esc**.



О схеме с еще не установленным новым компонентом, показанным цветом курсора, иногда говорится, что она находится в режиме нового компонента (*New component*).

Когда компонент размещен над существующим проводником, часть проводника под компонентом автоматически вырезается, так что редактировать проводники нет нужды.



Новый компонент имеет автоматически сгенерированное имя. Имя начинается с буквы заданного типа, сопровождаемой уникальным номером. Позже компонент может быть переименован в окне компонентов (**Components window**).

Когда компонент размещается в схеме, его изображение может быть модифицировано (отражено, отображено, повернуто). Некоторые типы компонентов могут иметь несколько видов (**view**). Команды, модифицирующие изображение и вид, также доступны в процессе размещения компонента с помощью клавиатуры, до нажатия на клавишу **Enter**.

Наряду со схемой новый компонент будет показан в окне компонентов (**Components Window**). В этом окне вы можете увидеть все модели (**models**), доступные для этого типа компонентов. Когда модель выбрана, появятся параметры (**parameters**) модели и станут доступны для редактирования. См. раздел *Окно компонентов* для подробностей.

Для переключения в окно компонентов из схемы поместите курсор поверх компонента и нажмите **Enter** или дважды щелкните по компоненту. В этом случае, если вы закончили редактировать параметры, нажав **Enter** или **Esc**, вы вернетесь в окно редактора схемы.

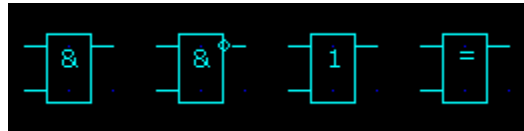
## Вид компонента (View)

типы компонентов имеют несколько видов: почти одинаковые изображения с небольшими отличиями. Разные виды могут иметь разное расположение выводов или некоторое функциональное различие. Несколько примеров отличий разного вида компонента:

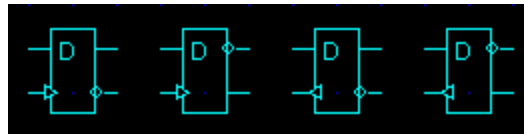
- Полярность управляющего сигнала:



- Функция логического элемента:



- Инверсия входов/выходов:



Чтобы изменить вид компонента, используйте следующие кнопки и горячие клавиши:

- **Next view.** Выбор следующего вида компонента с несколькими возможными видами.
- '+', '-' клавиши. Выбор следующего/предыдущего изображения компонента, включая изменение вида, отражение или отображение компонента.

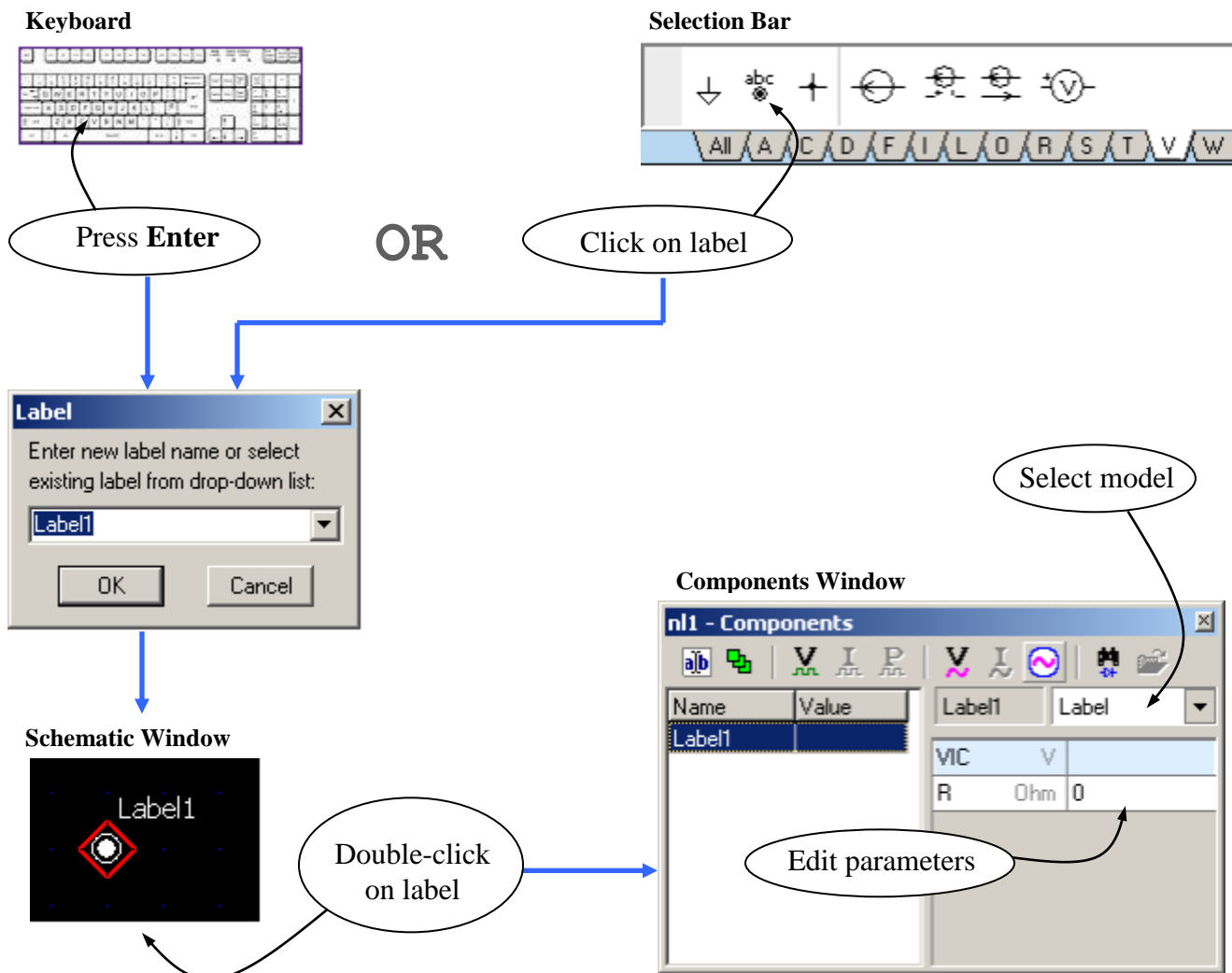
Эти команды также могут быть выполнены над новым компонентом, пока он размещается с использованием клавиатуры до нажатия на клавишу **Enter**.


## Этикетка (Label)

Этикетка похожа на компонент, за исключением того, что в схеме может быть много этикеток с одним и тем же именем. Все этикетки с одинаковыми именами электрически соединены. (Этикетки в подсхемах являются локальными для подсхемы и не соединены с основной схемой). Этикетки могут быть использованы:

- Для соединения разных точек схемы без проводников.
- Для соединения частей схемы, расположенных на разных листах.
- Как «пробник» симуляции (V trace - напряжение).
- Как источник напряжения.

Следующая диаграмма и пример показывают процесс размещения этикетки.



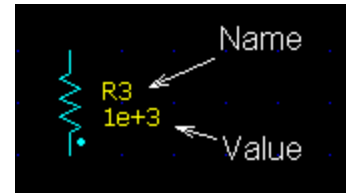
Чтобы поместить этикетку, нажмите **Enter** на проводнике или пустом месте или щелкните по иконке **Label** (  ) на панели выбора (**Selection Bar**). Появится диалоговое окно **Label**. Введите новое имя этикетки или выберите существующее из выпадающего списка и нажмите **OK**. Этикетка будет немедленно показана на схеме и в окне компонентов. Чтобы задать модель и

параметры этикетки, поместите курсор на этикетке и нажмите **Enter** или дважды щелкните по этикетке для переключения в окно компонентов.

## Атрибуты

Атрибуты компонента или этикетки включают имя (**Name**) и значение (**Value**). Доступны следующие режимы отображения атрибутов:

- Нет атрибутов
- Только **Name**
- Только **Value**
- **Name** и **Value**



Нажмите клавишу **Tab** для переключения режима отображения атрибута или выберите атрибуты в Основном Меню или контекстном меню схемы.

Атрибуты могут размещаться с разрешением выше, чем сетка схемы. «Сетка атрибутов» может устанавливаться в диапазоне от «1/1» до «1/32» относительно сетки схемы. Сетку атрибутов можно изменить на странице **Schematic** диалогового окна **Preferences**.

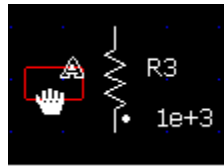
Чтобы переместить атрибут, щелкните по нему и перетащите:



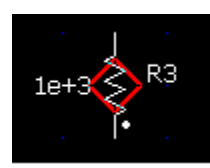
Click...



Drag...




Release...

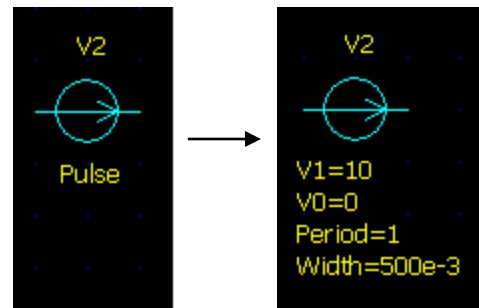
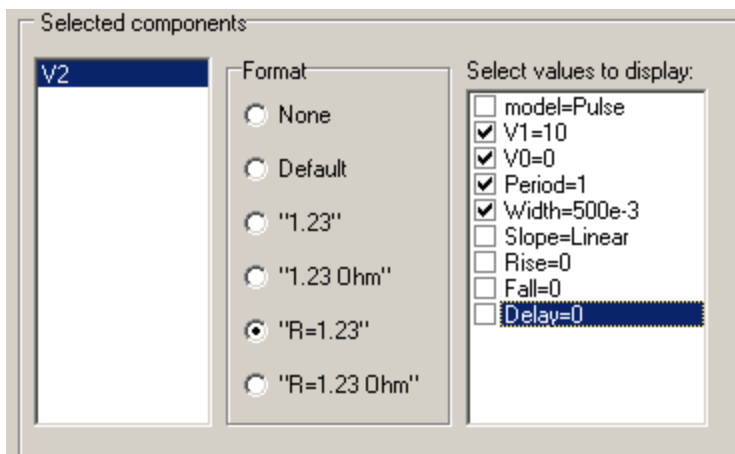


Unselect






Чтобы изменить ориентацию атрибута выберите компонент и щелкните по кнопке **Rotate attributes**  или нажмите **Ctrl-T**.

По умолчанию **Value** — это либо первый параметр модели компонента, либо имя модели. Список параметров, отображаемых для заданного компонента, может быть изменен пользователем в диалоговом окне **Format**. Щелкните правой клавишей мышки по компоненту, выберите команду **Format** в контекстном меню, затем выберите страницу **Attributes**. Для компонента, подсвеченного в списке, выберите формат и параметры для отображения.

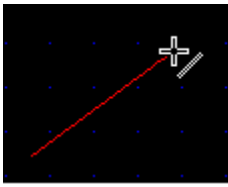


## Рисунки (линия, прямоугольник, овал)

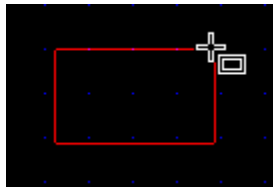
Рисунки (drawings) не оказывают влияние на функционирование схемы и используются исключительно для «декорирования». Рисунки могут размещаться с разрешением выше, чем сетка схемы. «Сетка рисования» может быть задана в диапазоне от «1/1» до «1/32» относительно сетки схемы. Сетка рисования может быть изменена на странице **Drawings** диалогового окна **Preferences**.

- Чтобы нарисовать линию, щелкните по кнопке **Line**  , чтобы переключиться в режим **Line**.
- Чтобы нарисовать прямоугольник, щелкните по кнопке **Rectangle**  , чтобы переключиться в режим **Rectangle**.
- Чтобы нарисовать овал, щелкните по кнопке **Oval**  , чтобы переключиться в режим **Oval**.

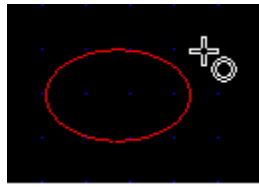
Щелкните по начальной точке, удержите клавишу мышки и перетащите курсор к конечной точке рисунка, затем отпустите клавишу мышки:



Line




Rectangle (square)

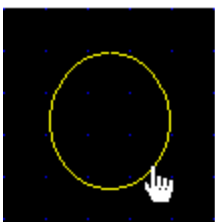


Oval (circle)

Чтобы нарисовать квадрат или окружность, держите клавишу **Ctrl** нажатой, когда рисуете и отпускаете клавишу мышки.

Щелкните кнопку выделения **Selection**  , чтобы переключиться назад в режим **Selection**.

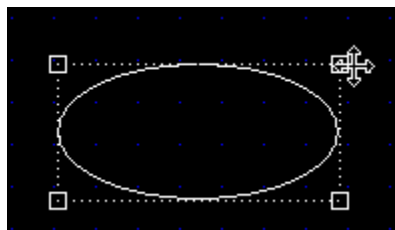
Чтобы изменить размер и/или вид рисунка, выделите рисунок, а затем щелкните по квадратному маркеру и растащите рисунок:



Click to select



Click and drag

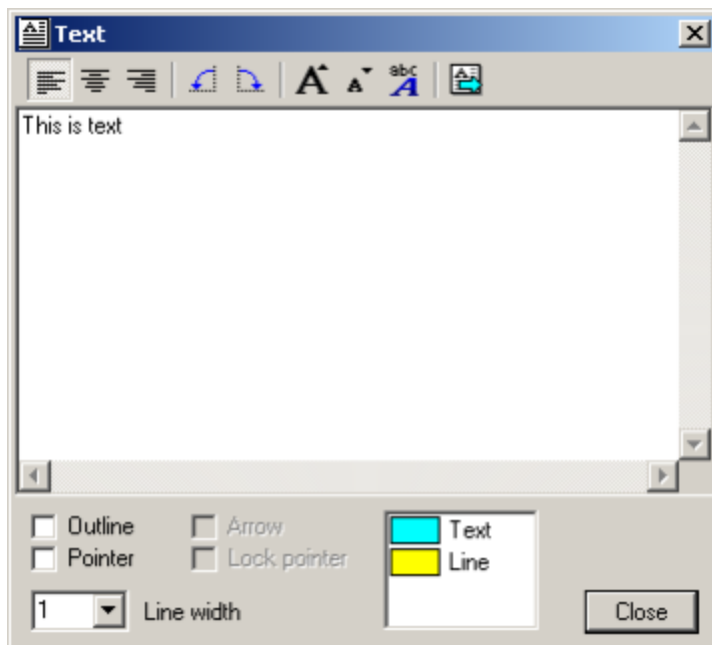


При размещении рисунок появляется в формате по умолчанию. Чтобы изменить цвет, ширину линии и тип рисования, выберите рисунок, затем команду **Edit | Format** основного меню. Или щелкните правой клавишей мышки по рисунку и выберите команду **Format** из контекстного меню. Чтобы отформатировать только один элемент рисунка, просто дважды щелкните по нему.

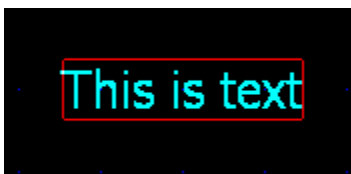


## Текст и Переменные

Чтобы вставить текст, щелкните по кнопке **Insert text** . Появится диалоговое окно **Text**:






Введите текст в текстовое окно. Текст одновременно будет показан на схеме:





Текст может быть отформатирован с использованием кнопок инструментальной панели:




**Alignment.** Устанавливает выравнивание для многострочного текста.

-  • **Align left.** Выравнивание по левому краю.
-  • **Center.** Выравнивание по центру.
-  • **Align right.** Выравнивание по правому краю.

**Orientation.** Изменить ориентацию текста.

-  • **Rotate left.** Повернуть влево.
-  • **Rotate right.** Повернуть вправо.

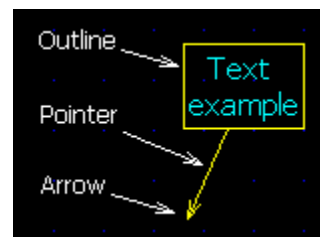
**Font.** Изменить размер шрифта или выбрать шрифт.

-  • **Larger font.** Увеличить размер шрифта.
-  • **Smaller font.** Уменьшить размер шрифта.
-  • **Select font.** Выбрать шрифт.

-  • **Run script.** Выполнить скрипт (если текст является выполняемым скриптом).

### Опции указателя и рамки

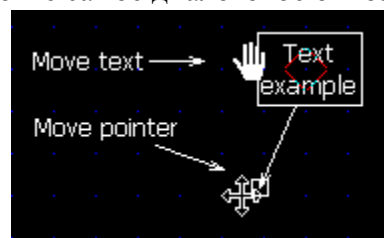
- **Outline.** Рисовать прямоугольную рамку.
- **Pointer.** Рисовать линию указателя от текста в заданную точку.
- **Arrow.** Рисовать линию указателя со стрелкой.
- **Lock pointer.** Закрепить конец указателя: конец указателя не будет перемещаться, даже если текст перемещается.
- **Line width.** Задаёт ширину линии для рамки и указателя.
- **Color.** Двойной щелчок по пункту в списке для изменения цвета.




Щелкните по кнопке **Close** когда закончите, чтобы закрыть диалоговое окно.

Для редактирования текста дважды щелкните по нему или щелкните правой клавишей мышки по нему и выберите команду **Edit** из контекстного меню. Появится то же самое диалоговое окно.

Чтобы переместить текст, щелкните по тексту и перетащите его. Если указатель прикреплен (locked), перемещен будет только текст. Чтобы переместить только указатель, щелкните по тексту, чтобы выделить его, затем щелкните и перетащите квадратик маркера на конце указателя



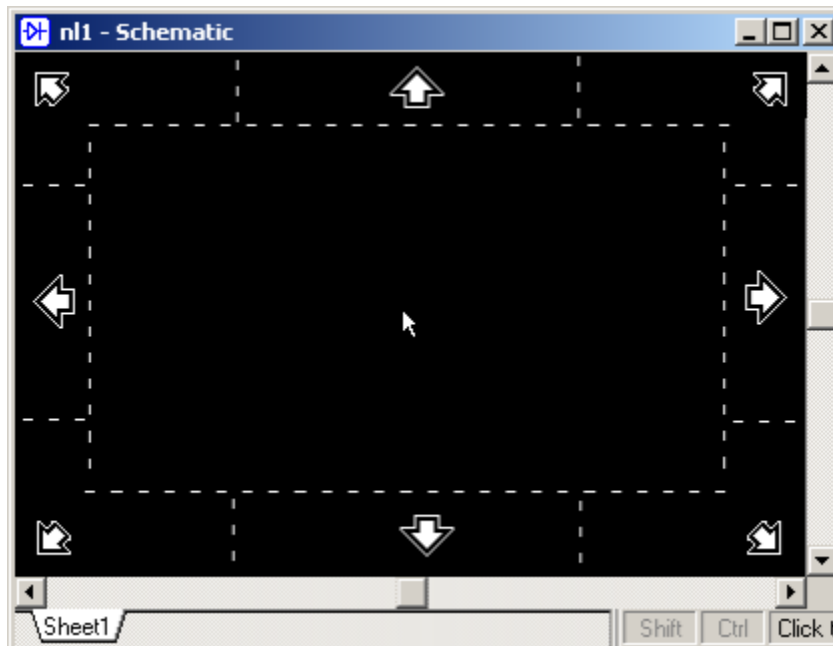
Чтобы вывести в качестве текста список переменных схемы, щелкните по кнопке **Insert variables**



 . Переменный и их значения будут показаны как обычный текст, за исключением того, что этот текст нельзя редактировать вручную: он будет автоматически обновляться при любых изменениях переменных и их значений.

## Прокрутка и масштабирование

Чтобы прокрутить схему, используйте любой из следующих методов:

- Переместите схемный курсор к краю окна схемы, окно будет автоматически прокручиваться.
- Переместите указатель мышки к краю окна схемы. Курсор мышки примет вид «большой стрелки». Щелкните или удержите левую клавишу мышки, чтобы прокрутить схему:



- В режиме *Selection*  установите указатель мышки на пустое место, нажмите и удержите клавишу **Shift**, затем щелкните и перетащите схему. Клавиша **Shift** временно переключит работу в режим *Scrolling* (прокрутка).
- Нажмите и удержите клавишу **Ctrl** и поверните колесико мышки для горизонтальной прокрутки.
- Нажмите и удержите клавишу **Shift** и поверните колесико мышки для вертикальной прокрутки.
- Используйте горячие клавиши клавиатуры **Shift-Up**, **Shift-Down**, **Shift-Right**, **Shift-Left**.
- В режиме *Scrolling*  щелкните и перетащите схему, или нажмите клавиши **Up**, **Down**, **Right**, **Left**.
- Нажмите клавишу **Home**, чтобы отцентровать курсор на экране
- Укажите и дважды щелкните мышкой по схеме, чтобы установить курсор и отцентровать его на экране.

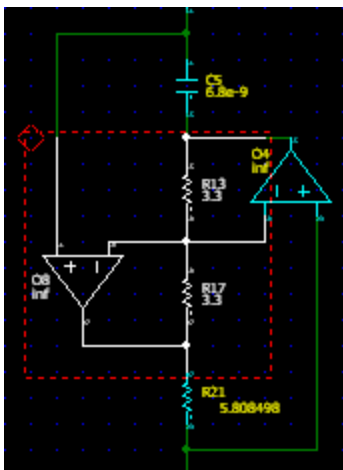
Для масштабирования схемы используйте любой из следующих методов:

- Поверните колесико мышки для увеличения и уменьшения масштаба.
- Щелкните по кнопке инструментальной панели или используйте горячие клавиши клавиатуры:

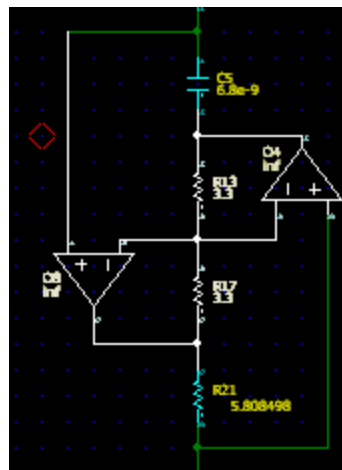
- 🔍 ○ **PgUp** - увеличение
- 🔍 ○ **PgDn** - уменьшение
- 🖱️ ○ **Ctrl-Home** – вывести схему на весь экран
- 🖱️ ○ **Shift-Home** - вывести выделенную часть схемы на весь экран
- Щелкните правой кнопкой мышки и выберите строчку контекстного меню **Zoom**, затем выберите масштаб в процентах (25%...250%).

## Выделение и сброс выделения

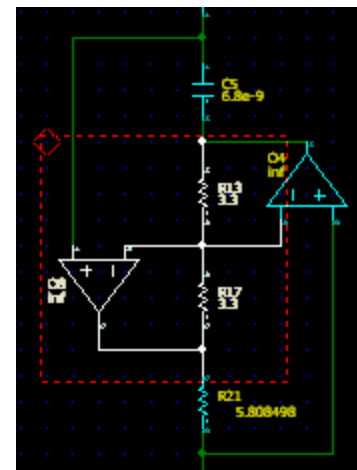
- Чтобы выделить один схемный элемент, укажите на него и щелкните по нему.
- Чтобы выделить блок, установите курсор на пустом месте, щелкните и растяните прямоугольник выделения. В зависимости от состояния клавиш **Ctrl** и **Shift** в момент отпускания клавиши мышки, может быть сделано следующее выделение:
  - Клавиши не нажаты. Только компоненты, полностью помещенные в область выделения; только часть соединений, попавшая в область выделения. Выделение ограничено прямоугольником.
  - Клавиша **Ctrl** нажата. Все компоненты и соединения с любой частью в выделенной области. Выделение не ограничено прямоугольником.
  - Клавиша **Shift** нажата. Только компоненты и соединения, полностью попавшие в область выделения. Выделение ограничено прямоугольником.



No keys depressed



Ctrl key depressed




Shift key depressed

- Чтобы добавить новое выделение к уже существующему, нажмите и удержите клавишу **Ctrl**, выделите новый элемент или новый блок.
- Для выделения всех элементов нажмите **Ctrl-A**.
- Щелкните правой клавишей мышки по выделенному элементу и откройте контекстное меню.
- Выбор команды **Select Net** в контекстном меню выделяет элемент схемы со всеми приходящими к нему проводниками, а также части схемы, подсоединенные к нему посредством этикеток (labels), включая другие листы.

- При перемещении схемного курсора автоматически выделяется элемент под курсором.
- Чтобы снять выделение, щелкните на пустом месте схемы или нажмите **Esc**.
- Чтобы снять выделение блока, переместитесь и щелкните по пустому месту вне блока или нажмите **Esc** дважды. Первое нажатие удаляет прямоугольник блока, второе снимает выделение всех элементов.

О схеме с выделением, ограниченным прямоугольником, также говорят, что схема находится в режиме *выделения блока*.

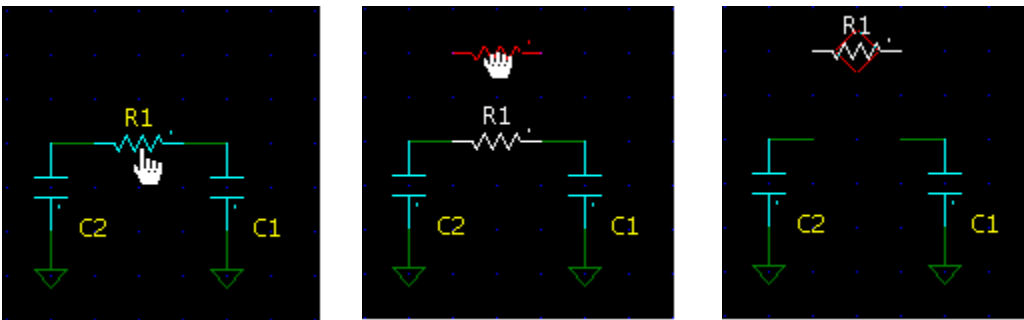
## Удаление

Чтобы удалить элементы, выделите элементы или блок, затем нажмите **Del**, или щелкните по кнопке **Delete** .

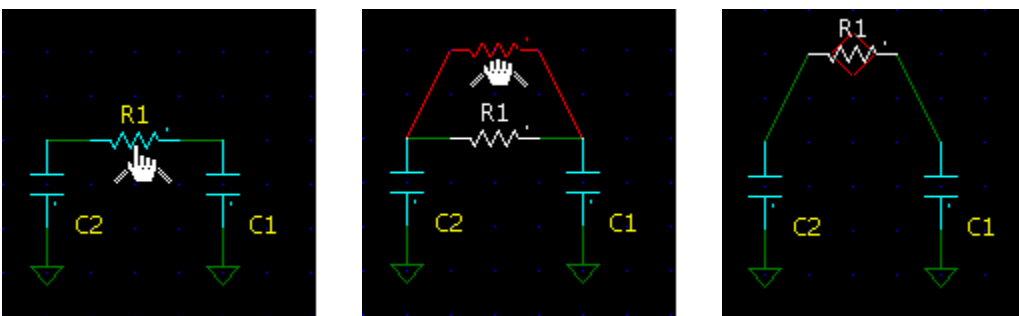
Чтобы удалить весь лист, щелкните правой клавишей по закладке **Sheets selection**, затем выберите команду **Remove sheet** в контекстном меню. Схема должна иметь хотя бы один лист, так что последний лист не может быть удален.

## Перемещение и копирование

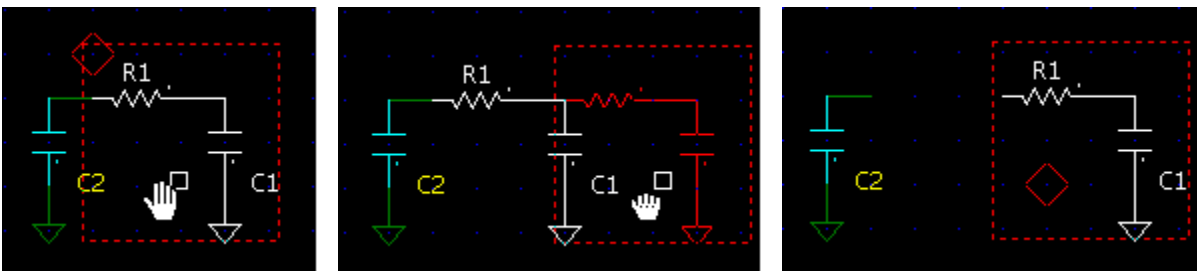
- Чтобы переместить элемент, щелкните по элементу и перетащите его в новое место:



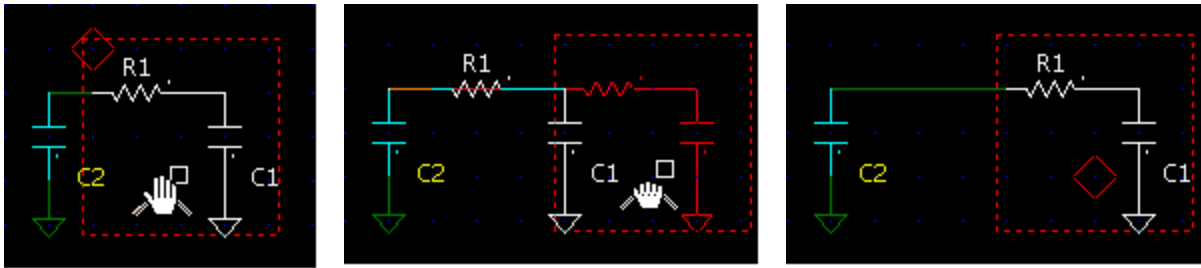
- Чтобы переместить элемент схемы с «резиновыми связями», удерживайте клавишу **Shift**, щелкните по элементу и перетащите в новое место:



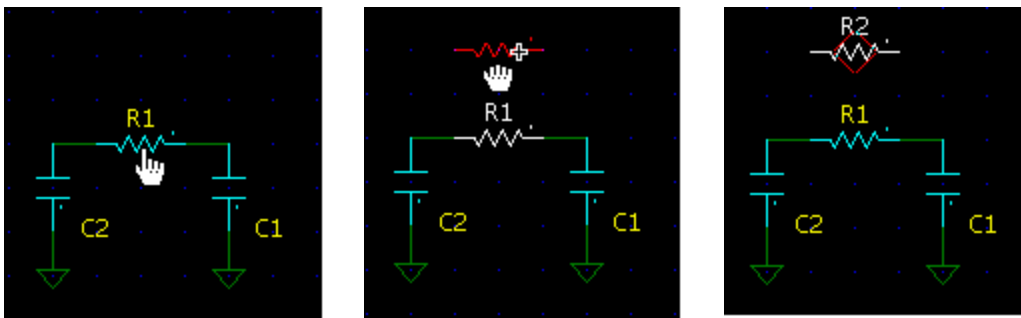
- Чтобы переместить блок, щелкните по выделению и перетащите в новое место.



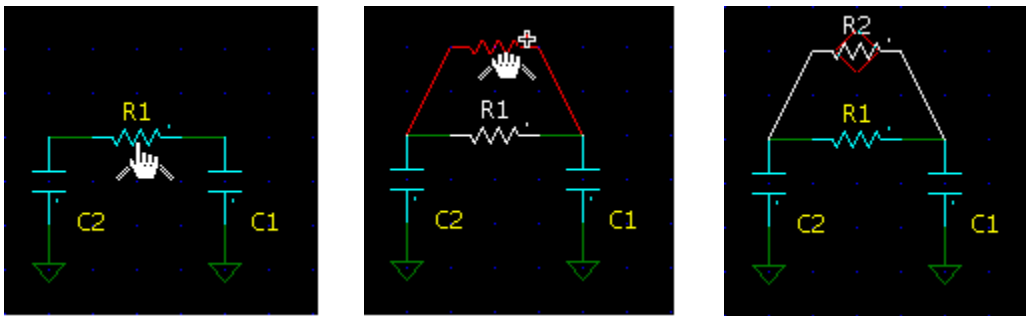
- Чтобы переместить блок с «резиновыми связями», нажмите и удержите клавишу **Shift**, щелкните по выделению и перетащите в новое место.



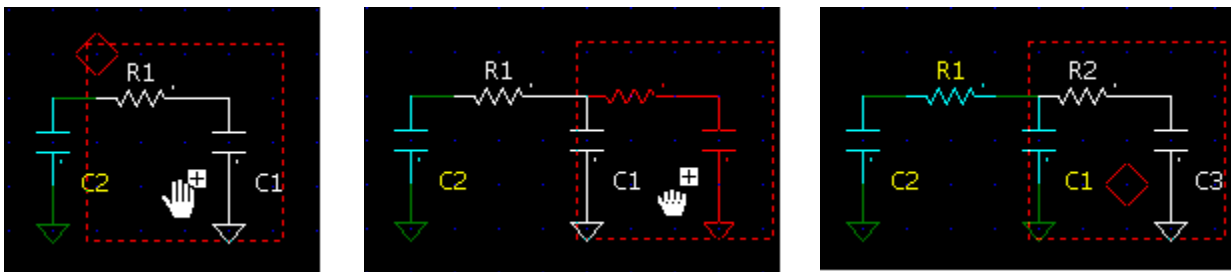
- Чтобы скопировать элемент, удержите клавишу **Ctrl**, щелкните по нему и перетащите в новое место.



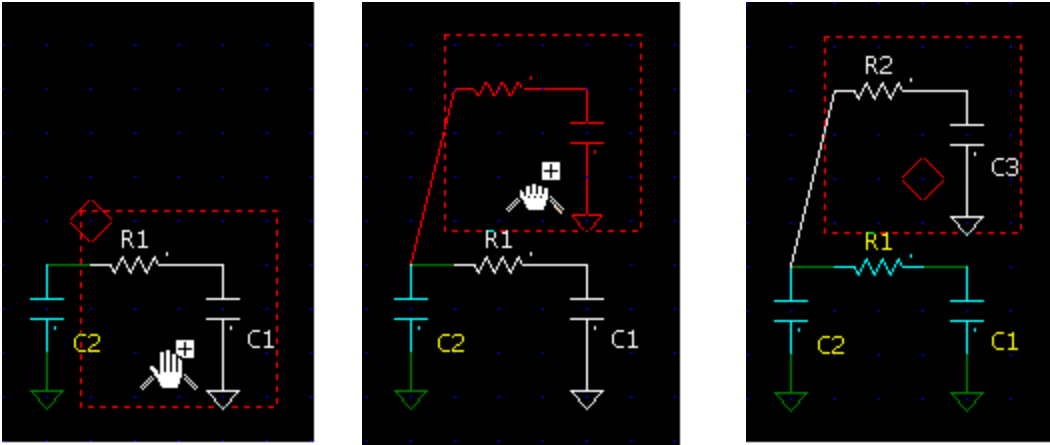
- Чтобы скопировать элемент с «резиновыми связями», удержите клавиши **Ctrl** и **Shift**, щелкните по элементу и перетащите его в новое место.



- Чтобы скопировать блок, удержите клавишу **Ctrl**, щелкните по выделению и перетащите его в новое место.

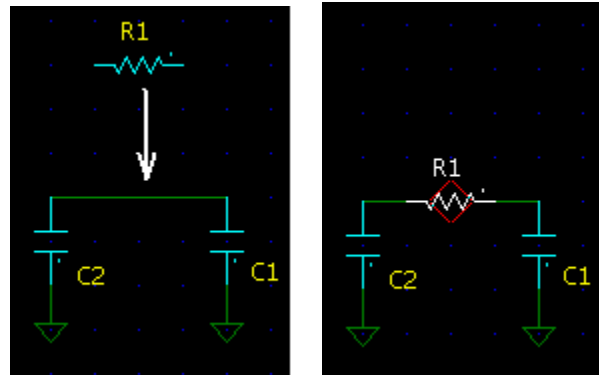


- Чтобы скопировать блок с «резиновыми связями», удержите клавиши **Ctrl** и **Shift**, щелкните по элементу и перетащите его в новое место.



Операции перемещения/копирования могут также выполняться с помощью стандартных команд меню и горячих клавиш: **Edit | Copy (Ctrl-C)**, **Edit | Cut (Ctrl-X)** и **Edit | Paste (Ctrl-V)**. Также возможно использование этих команд для перемещения/копирования элементов на другую страницу, в другой документ или NL5 приложение.

Когда компонент перемещается или копируется поверх существующего проводника, часть проводника под компонентом удаляется, так что удалять проводник вручную не требуется:



## Отключить и включить (Disable and Enable)

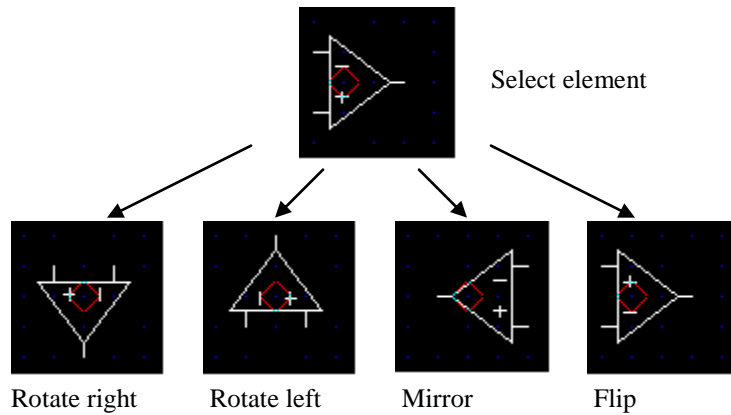
Элементы схемы могут быть «включены» или «выключены». Выключенные элементы показаны «disabled» цветом и не используются при симуляции. Возможность сделать элемент «выключенными» позволяет временно исключить его из симуляции без удаления из схемы. Чтобы выключить элемент, щелкните правой клавишей мышки по выделению и выберите команду **Disable** из контекстного меню. Чтобы включить элемент, выберите команду **Enable**.



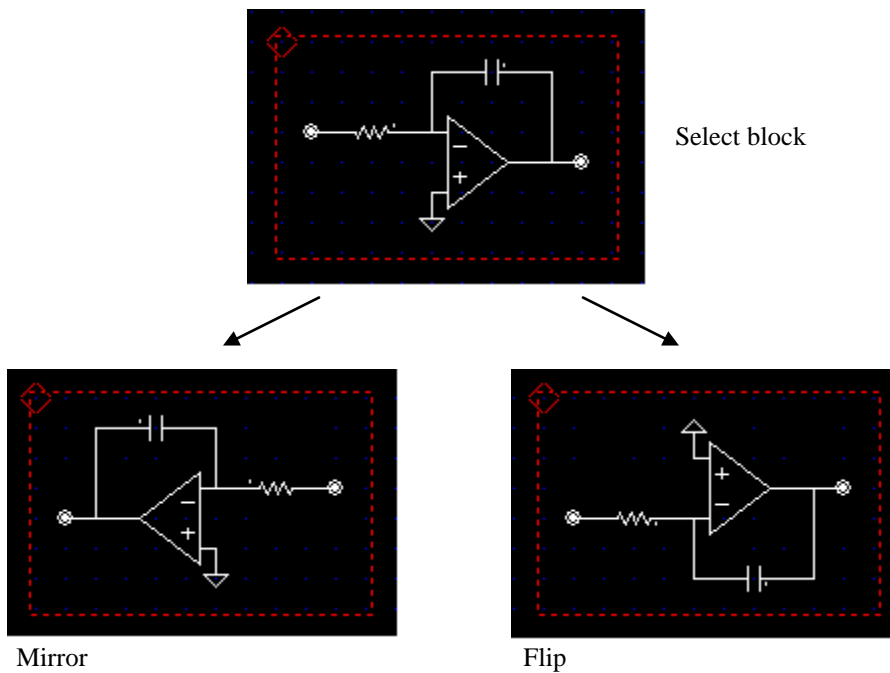
## Повернуть, отразить, перевернуть (Rotate, Mirror, Flip)

Чтобы изменить ориентацию элемента схемы, используйте команды **Rotate** (повернуть), **Mirror** (отобразить) и **Flip** (перевернуть):

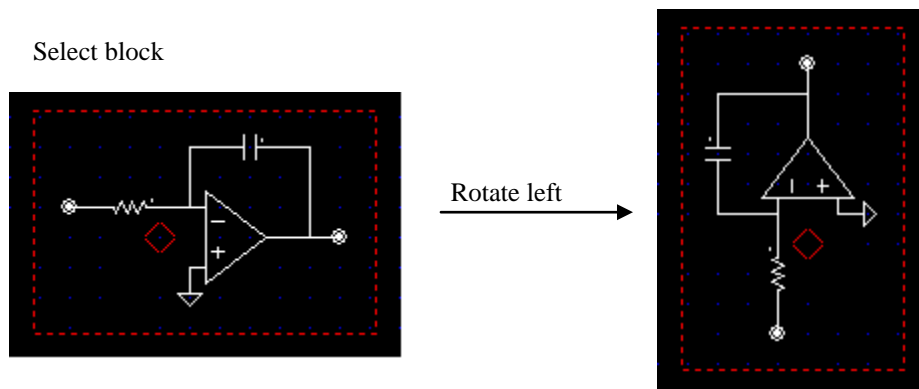
- **Rotate right (Ctrl-R)**. Повернуть вправо (по часовой стрелке).
- **Rotate left (Ctrl-L)**. Повернуть влево (против часовой стрелки).
- **Mirror (Ctrl-M)**. Зеркально отобразить (вокруг вертикальной оси).
- **Flip (Ctrl-F)**. Перевернуть (вокруг горизонтальной оси)



Эти команды могут быть применены к выделенному блоку:

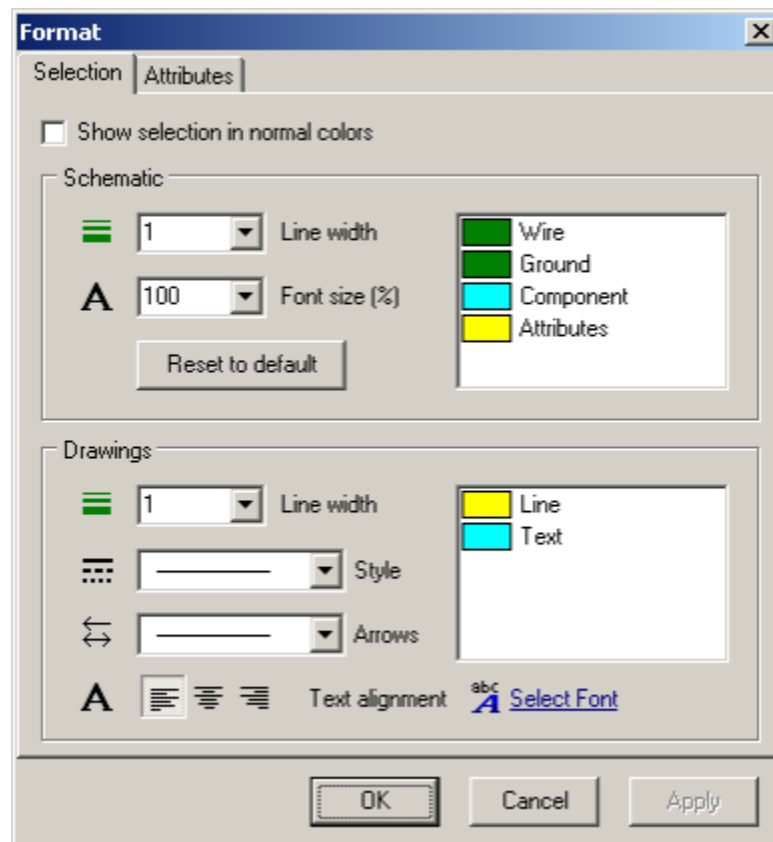


Чтобы повернуть блок, выделите его, затем поместите курсор на центр вращения внутри блока:



## Формат


Все элементы сначала размещаются на схеме с предустановленными свойствами (color - цвет, width - ширина, style - стиль, font - шрифт и т.д.). Чтобы отформатировать выделенный элемент, щелкните правой клавишей мышки по выделению и выберите команду **Format** из контекстного меню, или выберите команду **Edit | Format** в основном меню. Появится диалоговое окно **Format**:







Доступны будут только свойства, применимые к выделенным элементам. Например, если выделены только проводники, все поля, исключая **Schematic Line width** (ширина линий схемы) и **Wire color** (цвет проводника), будут не доступны. Если выбраны элементы, имеющие разные значения одного и того же свойства, соответствующие поля будут доступны, но останутся пустыми. Оставив их пустыми, вы оставите индивидуальные значения без изменений, иначе они примут одно и то же значение.

- **Show selection in normal colors.** Когда открывается диалоговое окно **Format**, все выделенные элементы имеют одинаковый, «selected» цвет. Установите этот флажок, чтобы видеть все элементы с их обычным цветом.



### Schematic (схема).

-  • **Line width.** Ширина линий выделенных элементов схемы: проводников, компонентов, этикеток и земли.
- A** • **Font size.** Размер шрифта атрибутов компонента.
- **Color.** Дважды щелкните по пункту в списке, чтобы изменить цвет.
- **Reset to default.** Вернуть все свойства к предустановленным.

### Drawings (рисунки).

-  • **Line width.** Ширина линий выделенных рисунков: линий, прямоугольников, овалов и текста.
-  • **Style.** Стилль линий для рисунков.
-  • **Arrows.** Установить стрелки для линий.
- A** • **Text alignment.** Выравнивание для многострочного текста.
-  • **Select font.** Выбрать шрифт для текста.
- **Color.** Дважды щелкните по пункту в списке, чтобы изменить цвет.
- **ОК.** Применить изменения и закрыть окно.
- **Cancel.** Игнорировать изменения и закрыть окно.
- **Apply.** Применить изменения не закрывая окна.

## Отменить и выполнить повторно (Undo, Redo)

Чтобы отменить изменения схемы или параметров компонентов, нажмите **Ctrl-Z** или щелкните кнопку **Undo** button . Чтобы выполнить отмененную операцию повторно, нажмите **Ctrl-Y**, или щелкните кнопку **Redo** . Количество отмененных операций, которое можно выполнить повторно, не ограничено.


















Размер буфера отмен неограничен, и предопределенной установкой является хранение всех изменений, сделанных с момента создания схемы или загрузки схемы из файла. Буфер очищается в момент сохранения схемы в файле. Однако для оптимальной производительности размер

буфера может быть ограничен на странице **Document** диалогового окна **Preferences**. Там же можно изменить опцию “Clear Undo buffer on schematic change” чтобы не очищать буфер во время сохранения файла.



В настоящее время операция отмены не распространяется на изменения параметров компонентов сделанных из скрипта, при оптимизации и при выполнении операции «Sweep».



























## Команды редактирования схемы

Следующие команды редактирования схемы, кнопки и горячие клавиши, доступны в Основном Меню, на основной инструментальной панели, инструментальной панели схемы и контекстных меню схемы.

-  • **Show schematic window (F2).** Показать окно схемы.
-  • **Schematic Tools.** Инструменты работы со схемой.
-  • **Check schematic.** Проверить схему.
  - **Sheets.** Открыть диалоговое окно «Sheets» (Листы).
-  • **Groups.** Открыть диалоговое окно «Groups» (Группы).
-  • **Parts list.** Показать список компонентов в окне **Properties**.
  - **Attributes** ►
    - **Name.** Переключение отображения имени атрибута.
    - **Value.** Переключение отображения величины атрибута.
-  • **Image** ►
  -  ○ **Copy to clipboard.** Скопировать изображение окна схемы в буфер обмена.
  -  ○ **Save as BMP.** Сохранить изображение окна схемы в файле формата «bmp».
  -  ○ **Save as JPG.** Сохранить изображение окна схемы в файле формата «jpg».
-  • **Cut (Ctrl-X).** Вырезать выделение (скопировать в буфер обмена и удалить)
-  • **Copy (Ctrl-C).** Скопировать выделение в буфер обмена.
-  • **Paste (Ctrl-V).** Вставить из буфера обмена.
-  • **Delete (Del).** Удалить выделение
-  • **Undo edit (Ctrl-Z).** Отменить операцию.
-  • **Redo edit (Ctrl-Y).** Выполнить повторно отмененную операцию.
  - **Select All (Ctrl-A).** Выделить все элементы схемы.
  - **Select Net.** Выделить элемент и все проводники, подсоединенные к нему непосредственно и через этикетки (включая другие листы).
  - **Format.** Форматировать выделенные элементы.
-  • **Preferences.** Открыть диалоговое окно **Preferences**.
-  • **Help (F1).** Помощь об окне схемы. Для помощи о компоненте, выделите компонент и нажмите **F1**.

## Панель инструментов (Toolbar) и контекстное меню

-  • **Selection.** Включить режим *Selection*.
-  • **Draw wire.** Включить режим *Wire*.

-  • **Scrolling.** Включить режим *Scrolling*.
-  • **Draw line.** Включить режим *Line*.
-  • **Draw rectangle.** Включить режим *Rectangle*.
-  • **Draw oval.** Включить режим *Oval*.
-  • **Insert text.** Ввести и разместить на схеме текст.
-  • **Insert variables.** Разместить на схеме список переменных схемы.
-  • **Zoom-in (PgUp).** Установить курсор в центр и увеличить схему.
-  • **Zoom-out (PgDn).** Установить курсор в центр и уменьшить схему.
-  • **Schematic to the screen (Ctrl-Home).** Вывести схему во весь экран.
-  • **Selection to the screen (Shift-Home).** Вывести выделенную часть схемы во весь экран.
-  • **Rotate right (Ctrl-R).** Повернуть выделенный элемент, блок, или новый компонент направо (по часовой стрелке).
-  • **Rotate left (Ctrl-L).** Повернуть выделенный элемент, блок, или новый компонент налево (против часовой стрелки).
-  • **Mirror (Ctrl-M).** Отразить выделенный элемент, блок, или новый компонент (вокруг вертикальной оси).
-  • **Flip (Ctrl-F).** Перевернуть выделенный элемент, блок, или новый компонент (вокруг горизонтальной оси).
-  • **Next view.** Выбрать следующий вид выделенного или нового компонента.
-  • **Rotate attributes (Ctrl-T).** Повернуть атрибуты выделенного компонента.
-  • **Edit text.** Редактировать выделенный текст.
-  • **Run script.** Выполнить скрипт из текста.
-  • **Edit component.** Открыть диалоговое окно **Edit component**. Доступно только для “customized” компонентов (конфигурируемых пользователем).
-  • **Open subcircuit.** Открыть файл, в котором находится подсхема компонента. Доступно только для компонентов с моделью **SubCir**, если задано имя файла подсхемы.
- **Enable.** Включить выделенные элементы.
- **Disable.** Выключить выделенные элементы.
- **Add trace ►**
  -  • **Voltage.** Добавить кривую напряжения переходного процесса для выделенного компонента. Если выделен проводник, сначала будет добавлена новая этикетка.
  -  • **Current.** Добавить кривую тока переходного процесса для выделенного компонента..
  -  • **Power.** Добавить кривую мощности для выделенного компонента..
  -  • **AC Voltage.** Добавить кривую AC напряжения для выделенного компонента. Если выделен проводник, сначала будет добавлена новая этикетка.
  -  • **AC Current.** Добавить кривую AC тока для выделенного компонента.
  -  • **Set AC Source.** Определить выделенный компонент как источник для AC анализа.

- **Zoom** ►
  - **25%...250%**. Масштабировать схему в заданных процентах.

## Клавиатура и горячие клавиши

Следующие клавиши клавиатуры и горячие клавиши можно использовать при редактировании схемы.

- **Пробел (Space)**.
  - В режиме *Selection*: переключиться в режим *Wire*.
  - В других режимах: переключиться в режим *Selection*.
- **Home**. Центрировать курсор на экране.
- **Tab**. Переключить отображение атрибутов (имя и значение).
- **Enter**.
  - В режиме *New component*: поставить компонент.
  - На компоненте или этикетке: переключиться в окно компонентов для редактирования параметров.
  - На проводнике или пустом месте: поставить новую этикетку (label).
- **Esc**.
  - В режиме *New component*: отменить.
  - В режиме *Block selected*: удалить рамку блока.
  - При рисовании проводника, линии, прямоугольника или овала: отменить.
  - При выделении мышкой: отменить.
  - Все режимы кроме *Selection*: переключиться в режим *Selection*.
  - Снять выделение.
- **Right, Up, Left, Down** (клавиши курсорной части клавиатуры)
  - В режиме *New component*: переместить компонент.
  - В режиме *Block selected*: переместить выделение.
  - В режиме *Selection*: изменить направление курсора, переместить курсор.
  - В режиме *Wire*: изменить направление курсора, нарисовать проводник.
  - В режиме *Scrolling*: прокрутить схему.
- **Shift-Right, Shift-Up, Shift-Left, Shift-Down**. (клавиши курсорной части клавиатуры при нажатой клавише **Shift**). Прокрутка схемы.
- **‘.’ (точка)**. Поставить соединение.
- **‘G’, ‘g’**. Поставить землю.
- **‘A’...’Z’, ‘a’...’z’**. Выбор нового компонента по типу буквы.
- **‘+’, ‘-’**. Выбор следующего/предыдущего изображения компонента изменением вида, отражения и переворачивания, когда это применимо.

## Операции с мышкой

Следующие операции с мышкой можно использовать при редактировании схемы.

- **Щелчок** (левая клавиша).
  - На пустом месте: снять выделение, установить курсор.
  - На выделенном блоке: установить курсор.
  - На элементе: выделить элемент.
  - В режиме *Wire*: установить курсор.
- **Ctrl-click**. То же, что и щелчок, но не снимает выделения (добавление к выделению).
- **Right-click**. То же, что и щелчок, плюс открывает контекстное меню.
- **Двойной щелчок (Double-click)**.
  - На компоненте или этикетке: выделить и переключиться в окно компонентов.
  - На линии, прямоугольнике или овале: форматировать элемент.
  - На пустом месте: установить курсор и центрировать экран.
- **Щелчок и перетаскивание (Click and drag)**.
  - На пустом месте: выделить блок.
  - На атрибуте: переместить атрибут.
  - На выделении: переместить выделение.
  - В режиме *Wire*: рисовать проводник.
- **Ctrl-click and drag**.
  - На пустом месте: добавить блок к выделению.
  - На выделении: скопировать выделение.
  - В режиме *Wire*: рисовать диагональный проводник.
- **Shift-click and drag**.
  - На пустом месте: прокрутить схему.
  - На выделении: переместить выделение с «резиновыми связями».
- **Ctrl-Shift-click and drag**.
  - На выделении: скопировать выделение с «резиновыми связями».
- **Колесико мышки**. Увеличить/уменьшить масштаб. Схема масштабируется относительно позиции указателя мышки.
- **Ctrl- колесико мышки**. Прокрутить горизонтально.
- **Shift- колесико мышки**. Прокрутить вертикально.



## Компоненты

NL5 поддерживает около 60 типов компонентов. Каждый тип компонента имеет назначенную букву, символ и модели.

**Буква** идентифицирует функциональную группу компонента. Например, все типы компонентов с буквой **S** — это ключи (переключатели). Панель выбора компонентов (Selection Bar) имеет закладку, предназначенную для каждой буквы. Когда при размещении компонента используется клавиатура, клавиша буквы определяет тип компонента. Имя компонента по умолчанию начинается с этой буквы

**Символ** — это изображение типа компонента: то, как компонент отображается в схеме. Панель выбора содержит символы всех доступных типов компонентов

Некоторые компоненты являются “**customized**” (конфигурируемые пользователем): изображение этих компонентов (размер компонента, количество, положение и названия выводов) может быть сконфигурировано в диалоговом окне **Edit Component**. Диалоговое окно откроется автоматически в момент постановки компонента, а также может быть открыто в любое время после того, как компонент установлен. См. раздел *Редактирование конфигурируемого компонента* для подробностей


**Модель** определяет функционирование компонента. Например, источник напряжения имеет модели **Pulse** (импульс), **Sin** (синусоида), **Step** (ступенька), **SubCir** (подсхема) и другие. Каждая модель имеет набор параметров, свойственных этой модели. Модель и параметры компонента могут быть установлены в окне компонентов (**Components Window**).

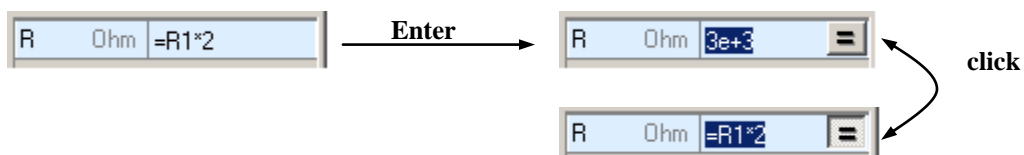
В этой главе представлена общая информация обо всех компонентах и более детальная информация о некоторых типах компонентов и параметрах. См. **Приложение 1** для детального описания всех типов, моделей и параметров.

## Формулы


Большая часть параметров компонента (числа) и все схемные переменные могут быть определены как **формулы**. Формула всегда начинается со знака равенства «=»:

```
=Var1*2
=R1/2
=max (R1, R2, R3)
```

Чтобы ввести формулу, напишите выражение, начинающееся со знака «=» и нажмите клавишу **Enter**. Формула будет рассчитана и ее текущее числовое значение будет отображено. Щелкните по изображенной рядом кнопке , чтобы посмотреть или редактировать формулу:



Если выражение формулы содержит ошибку и не может быть вычислено, будет отображен текст #VAL вместо числа.

Чтобы очистить формулу, введите новое выражение или число без знака равенства вместо числового значения, или щелкните по кнопке , чтобы перейти в режим редактирования, сотрите формулу и нажмите **Enter**.

Если формула содержит имя компонента, а компонент был переименован, формула будет автоматически обновлена. Формула не может содержать время и значения, которые меняются в процессе вычисления, такие как напряжение, ток и мощность. Циклические ссылки (когда некоторые из параметров в формуле указывают на эту же формулу) не разрешены, и вызовут сообщение об ошибке.

## Функции

Некоторые параметры компонентов могут быть **функцией**. Функция — это выражение, которое перерасчитывается при каждом шаге вычисления переходного процесса или АС анализа. Кроме чисел и имен, некоторые функции могут использовать следующие переменные:

$t$  – текущее время переходного процесса, s.

$f$  – текущая частота АС, Hz

$w$  – текущая угловая частота АС,  $w = 2\pi f$ .

$s$  или  $p$  – параметр Лапласа,  $s = p = j\omega$ .

$x, y$  – входной сигнал для модели **Function**.

$V(name)$  – напряжение на компоненте  $name$ . Кривая **V** должна быть разрешена для компонента.

$I(name)$  – ток через компонент  $name$ . Кривая **I** должна быть разрешена для компонента.


$P(name)$  – мощность на компоненте  $name$ . Кривая **P** должна быть разрешена для компонента.

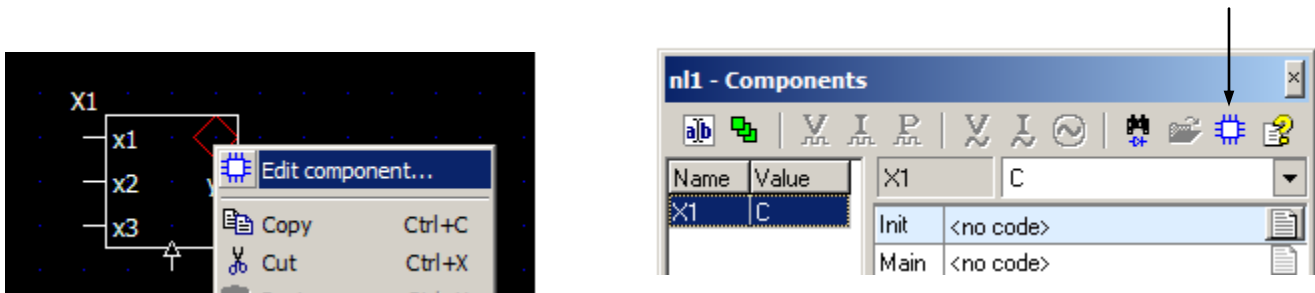
Функция вводится как выражение без знака равенства. Например:

```
sin(t*1000)*(1+cos(t*10))
(t%2>1)?1:-1
mag(x,y)
sq(V(r1))/r1
1/(1+s*R1*C1)
1000*f
```

## Редактирование конфигурируемого компонента (customized)

Когда ставится компонент конфигурируемого типа (**customized**), сначала появляется диалоговое окно **Edit Component**. В этом окне можно задать размер компонента, количество и расположение выводов, имена выводов. Задайте параметры и нажмите **ОК**, чтобы поставить компонент.

Чтобы редактировать уже поставленный компонент, щелкните на компоненте правой кнопкой мышки и выберите команду **Edit component** из контекстного меню, или щелкните кнопку **Edit component**  в окне компонентов (**Components window**):

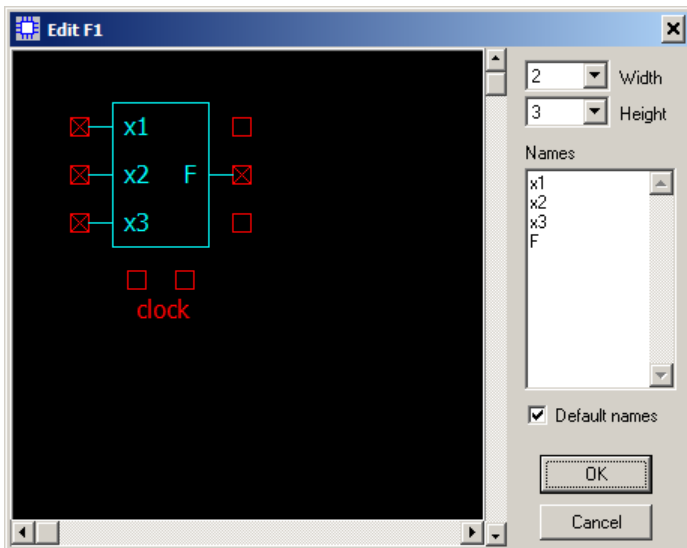


Окно **Edit Component** выглядит по-разному для разных компонентов, так как параметры доступные для разных компонентов могут быть разными. Для всех компонентов:

- Выберите новые размеры компонента из выпадающих списков **Width (ширина)** и **Height (высота)**.
- Щелкните на квадратных маркерах, чтобы добавить/удалить выводы. Разные компоненты могут иметь разное доступное количество и положение выводов.
- Нажмите **ОК**, чтобы применить изменения и закрыть окно, или **Cancel**, чтобы закрыть окно без принятия изменений.

Существуют следующие типы окна для разных компонентов:

## F - Custom function

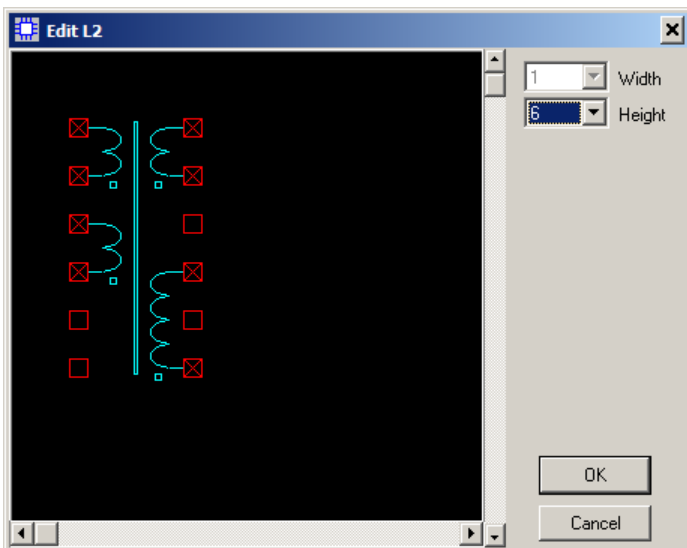


- Maximum width: 32.
- Maximum height: 8.
- Number of inputs: 0...8.
- Number of outputs: 1.
- Number of clocks: 0...1.

- Установите флажок **Default names** (предустановленные имена) чтобы генерировать имена выводов автоматически.
- Снимите флажок **Default names**, чтобы вводить произвольные имена выводов в окне **Names**.

## L - Custom coupled inductors

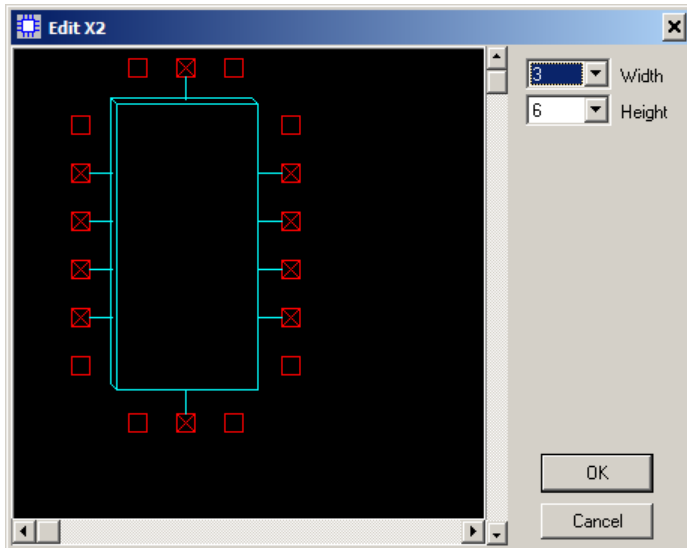
### W - Custom transformer



- Maximum width: 1.
- Maximum height: 32.
- Number of windings: 1...9.

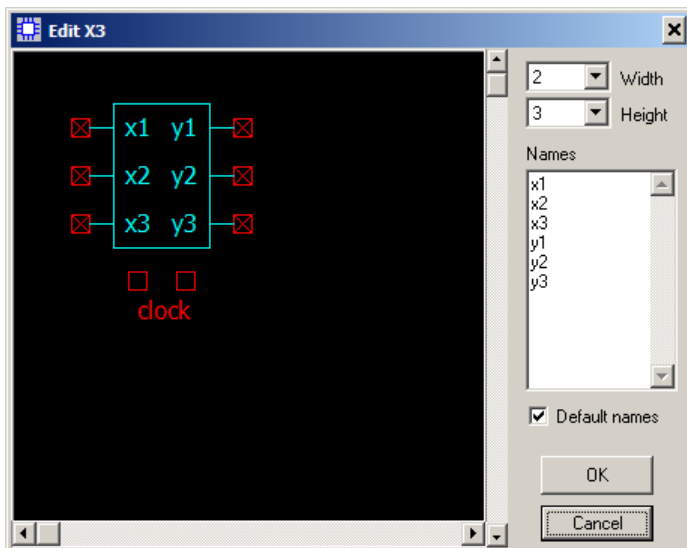
- Щелкните на квадратных маркерах, чтобы выбрать начало и конец обмотки.

## X - Custom block (Subcircuit)



- Maximum width: 32.
- Maximum height: 32.
- Number of pins: 1...128.

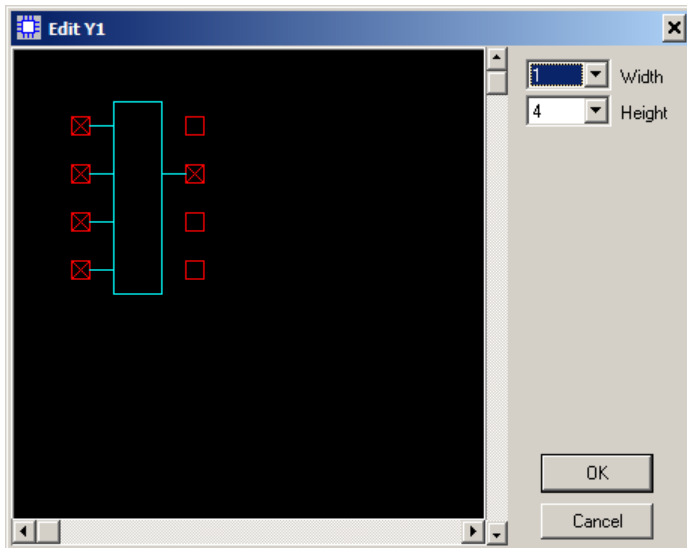
## X – Code




- Maximum width: 32.
- Maximum height: 32.
- Number of inputs: 0...32.
- Number of outputs: 0...32.
- Number of clocks: 0...1.

- Установите флажок **Default names (предустановленные имена)** чтобы генерировать имена выводов автоматически.
- Снимите флажок **Default names**, чтобы вводить произвольные имена выводов в окне **Names**.

## Y – Custom logic



- Maximum width: 32.
- Maximum height: 32.
- Number of inputs: 1...32.
- Number of outputs: 1.


Логическая функция компонента, а также инвертирующий-неинвертирующий выход выбирается с помощью Вида компонента (используйте команду **Next view** или кнопку ).

## Работа с подсхемами

Модель подсхемы (**SubCir**) позволяет создавать простые и хорошо читаемые схемы с помощью замены некоторой части схемы одним компонентом. Когда начинается симуляция, компонент с моделью **SubCir** заменяется фактической схемой, загружаемой из файла подсхемы. **SubCir** модель доступна почти для всех компонентов. Для некоторых компонентов типа **X** эта модель является единственно доступной.

**SubCir** модель имеет одинаковые параметры для всех типов компонентов:

| Model  | Parameter | Units | Description                                    |
|--------|-----------|-------|--|
| SubCir | File      |       | Имя файла подсхемы                             |
|        | Pin1      |       | Имя этикетки подсхемы, соединенной с выводом 1 |
|        | ...       |       | ...  |
|        | PinN      |       | Имя этикетки подсхемы, соединенной с выводом N |
|        | Cmd       |       | Стартовая командная строка подсхемы            |
|        | IC        |       | Строка начальных условий подсхемы              |

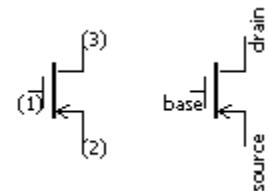
**File** — это имя файла подсхемы. Введите имя файла вручную или щелкните по кнопке  и выберите имя в диалоговом окне. Расширение файла “.nl5” может быть опущено.

Можно использовать полный путь к файлу, например:

```
C:\Program files\NL5\Projects\Modulator\clock_gen.nl5
```

Однако если файл подсхемы расположен в той же директории, что и основной файл схемы, или в директории определенной как «Директория Подсхем» (страница **Application** окна **Preferences**), то можно использовать краткое имя файла без пути и расширения (например, «clock\_gen»). Краткое имя также отображается на схеме. Рекомендуется держать файлы подсхем в той же директории, что и основной файл схемы, или в «Директории Подсхем», так как это позволяет использовать краткие имена и делает проект «переносимым» (**portable**).

Параметры «Pin1»...«PinN» используются для назначения выводам компонента этикеток, определенных в подсхеме. Если имена этикеток не введены для выводов, на изображении компонента отображаются номера выводов в скобках. Когда имя этикетки определено, оно отображается на изображении компонента. Любой параметр «PinN» может быть пустым, то есть соответствующий вывод компонента не связан с этикеткой подсхемы.



**Cmd** — это командная строка, применяемая к подсхеме, после того как она загружена из файла (перед началом анализа). Строка состоит из команд в формате «**name=value**», разделенных точкой с запятой «;». Это позволяет использовать один и тот же файл подсхемы с модифицированными значениями для разных компонентов. Например:


```
R1=1k;R2=12k;C1=5n
```

где R1, R2 и C1 — компоненты подсхемы.

**IC** — это текстовая строка, аналогичная строке **Cmd**, но содержащая только начальные условия (initial conditions) для компонентов подсхемы. Например:

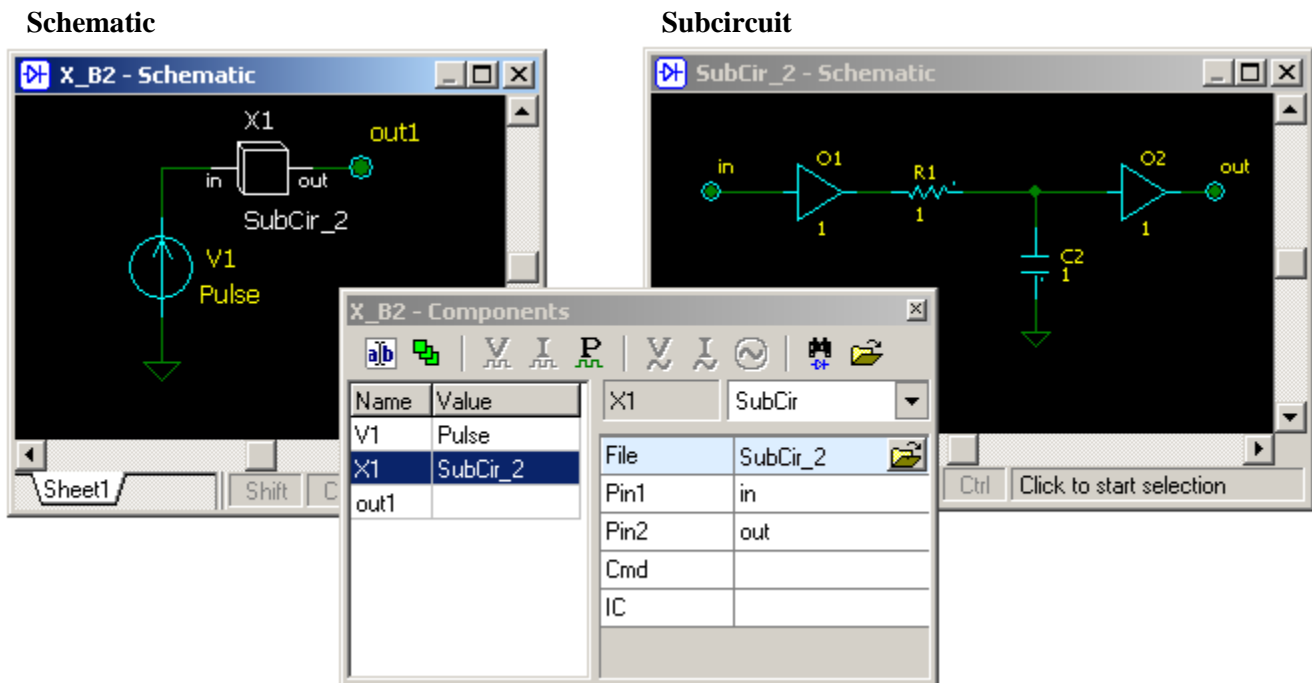
```
C1.IC=10;O2.IC=0
```

где C1 и O2 — компоненты подсхемы. В отличие от параметра **Cmd**, строка **IC** может быть автоматически модифицирована некоторыми командами. Команда **Transient | Save IC** заполняет строку текущими начальными условиями всех компонентов в подсхеме. Команда **Schematic | Tools**, страница **Initial Condition**: при установленном флажке **Set subcircuits to empty (no IC)** эта строка будет очищена.

Параметры **Cmd** и **IC** могут быть отредактированы прямо в строке параметра, или в специальном диалоговом окне, если щелкнуть кнопку  .

Подсхема всегда загружается из файла при старте симуляции. Если подсхема была модифицирована, она должна быть сохранена в файл перед запуском симуляции. Исключением будет случай, когда подсхема и основная схема открыты в той же копии (instance) приложения NL5. В этом случае подсхема будет взята прямо из памяти NL5, так что сохранения изменений в файл не требуется

Пример использования подсхемы:





## Работа с PWL моделью

Модель **PWL** (“PieceWise Linear”— кусочно-линейная функция) описывает нелинейную характеристику компонента кусочно-линейным приближением, а также описывает некоторые компоненты с параметром, управляемым напряжением или током.

Для **нелинейных компонентов** параметр “pwl” – это «ступенчатая» функция, задающая **дифференциальное** значение параметра как функцию напряжения на компоненте или тока через компонент. Например, параметр “pwl” для нелинейного резистора задает дифференциальное сопротивление как функцию напряжения на резисторе. Результирующая кусочно-линейная вольт-амперная характеристика резистора вычисляется автоматически. Аналогично, параметр “pwl” для нелинейного конденсатора задает дифференциальную емкость как функцию напряжения на конденсаторе, а результирующая кусочно-линейная вольт-кулонная характеристика вычисляется автоматически. В следующей таблице показано, какие «ступенчатые» параметры и кусочно-линейные характеристики применяются для нелинейных компонентов в NL5:

| “Step-like” parameter | PWL characteristic | Component   |
|-----------------------|--------------------|---|
| R(V)                  | I(V)               | Resistor, diode, zener                                    |
| R(I)                  | V(I)               | Resistor  |
| C(V)                  | Q(V)               | Capacitor   |
| L(I)                  | H(I)               | Inductor  |
| K(V)                  | V(V)               | Voltage controlled voltage source, operational amplifiers |
| K(I)                  | V(I)               | Current controlled voltage source                         |
| K(V)                  | I(V)               | Voltage controlled current source                         |
| K(I)                  | I(I)               | Current controlled current source                         |

Для компонентов, **управляемых напряжением или током**, параметр “pwl” – это «ступенчатая» функция, задающая **абсолютное** значение параметра как функцию внешнего управляющего напряжения ( $V_{in}$ ) или тока ( $I_{in}$ ). Например, параметр “pwl” для резистора управляемого напряжением задает абсолютное сопротивление как функцию управляющего напряжения.

В следующей таблице показаны «ступенчатые» параметры для компонентов, управляемых напряжением или током:

| “Step-like” parameter | Component                    |
|-----------------------|------------------------------|
| R( $V_{in}$ )         | Voltage controlled resistor  |
| R( $I_{in}$ )         | Current controlled resistor  |
| C( $V_{in}$ )         | Voltage controlled capacitor |
| C( $I_{in}$ )         | Current controlled capacitor |
| L( $V_{in}$ )         | Voltage controlled inductor  |
| L( $I_{in}$ )         | Current controlled inductor  |
| K( $V_{in}$ )         | Voltage controlled amplifier |
| K( $I_{in}$ )         | Current controlled amplifier |

PWL модель резистора (**R**) описана ниже в примере; PWL модели других компонентов также похожи (с небольшими изменениями).

«Ступенчатый» параметр «pwl» является строкой с разделенными запятыми значениями:

R0, V1, R1, V2, R2, ..., VN, RN

где:

R0 - сопротивление, пока абсолютное значение напряжения на резисторе меньше v1.

R1 - сопротивление, пока абсолютное напряжение на резисторе между v1 и v2.

...

RN - сопротивление, когда абсолютное значение напряжения на резисторе больше vN.

Величины v1...vN должны быть заданы в возрастающем порядке. Результирующая кусочно-линейная характеристика для нелинейных элементов вычисляется автоматически и **всегда** проходит через начало координат (точка 0,0).

Параметр «pwl» может быть симметричной или несимметричной функцией.

**Симметричный** параметр определен на интервале от нуля до плюс бесконечности,

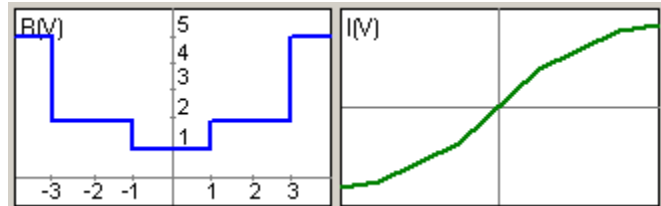
отрицательная часть всегда симметрична

положительной. Для симметричного

параметра все напряжения v1...vN

**положительные**. Следующие графики R(V) и I(V) представляют параметр и характеристику, определенные строкой:

1, 1, 2, 3, 5



**Несимметричный** параметр определен во всем диапазоне: от минус до плюс бесконечности. Для

несимметричного параметра напряжения

v1...vN могут быть как положительными, так и

отрицательными, но при этом обязательно

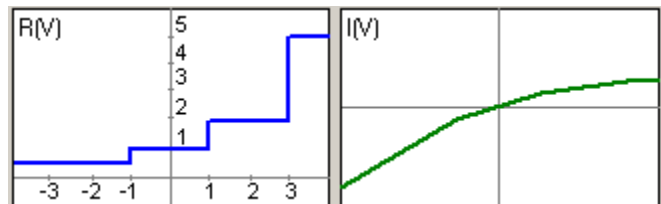
должно быть напряжение, равное **нулю**.


Нулевое напряжение является признаком

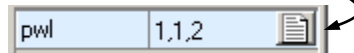
несимметричного параметра. Следующие

графики R(V) и I(V) представляют параметр и характеристику, определенные строкой:

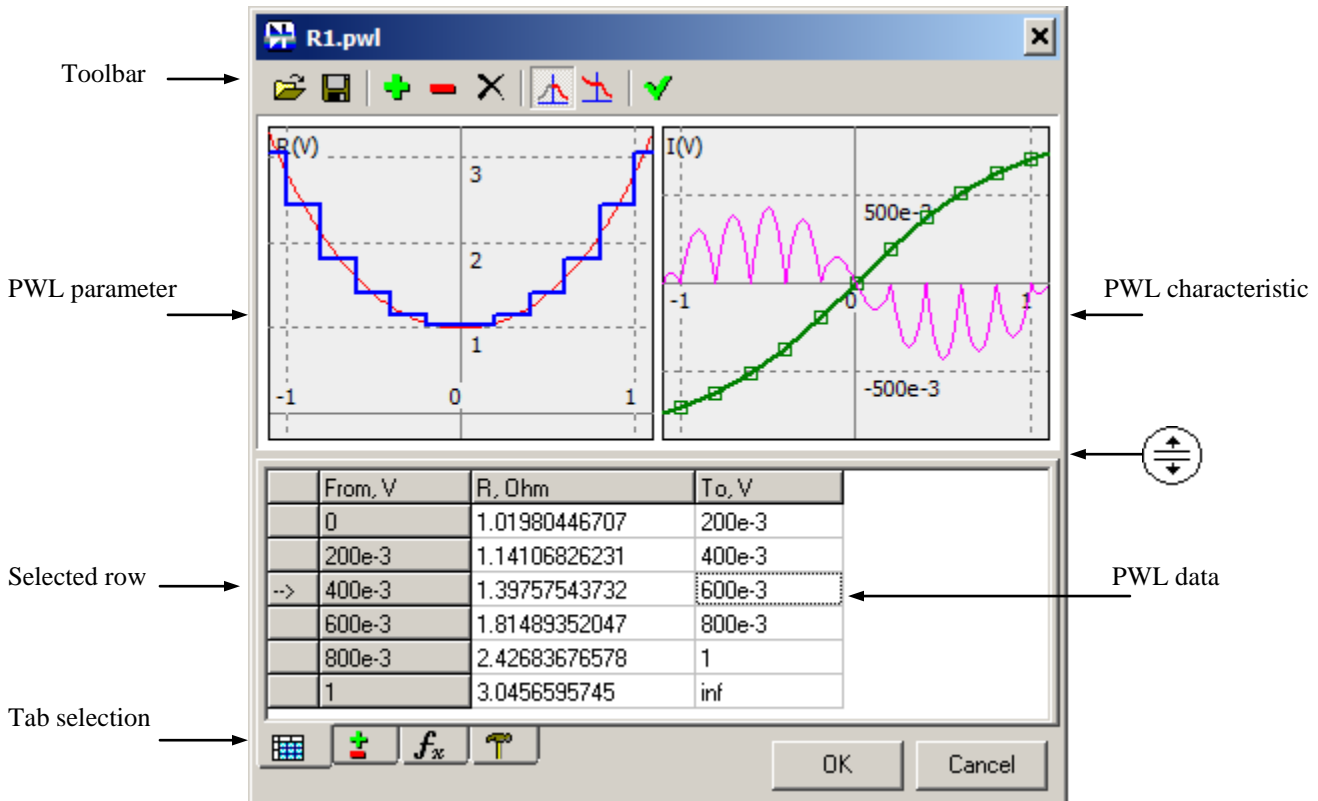
.5, -1, 1, 0, 1, 1, 2, 3, 5






Параметр “pwl” может быть отредактирован в диалоговом окне **PWL**. Щелкните по кнопке  справа от параметра «pwl», чтобы открыть диалоговое окно:











Типичный вид окна **PWL** и его основные компоненты показаны ниже:

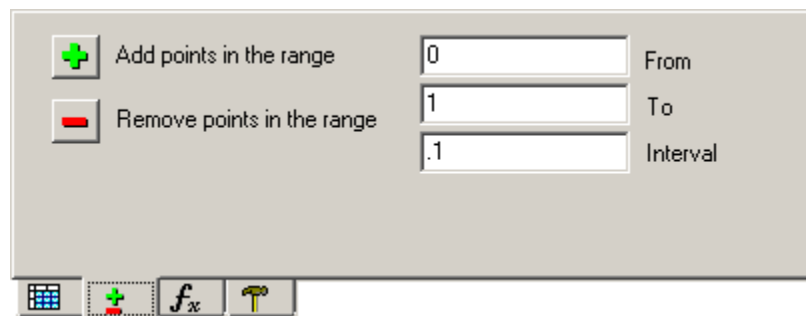



В верхней части окна находятся графики «ступенчатого» параметра “pwl” и кусочно-линейной характеристики (для нелинейных элементов). В нижней части окна находятся 4 закладки для ввода данных и конфигурации.

- Переместите курсор поверх области «разделителя» , затем нажмите левую кнопку мышки и перетащите разделитель, чтобы изменить размер панелей.
- Щелчок правой кнопкой мышки на графике вызывает контекстное меню с соответствующими командами.
- Щелкните **OK** чтобы применить изменения и закрыть окно, или **Cancel** чтобы закрыть окно без изменений.
- Щелчком по кнопкам инструментальной панели выполняются следующие операции:
  -  ○ Прочитать данные из файла (в формате “csv”).
  -  ○ Сохранить данные в файл (в формате “csv”).

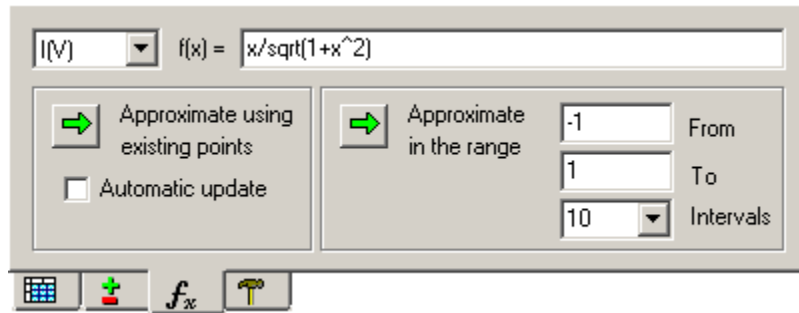
-  ○ **Split** – «расщепить» выделенную строку.
-  ○ **Remove** – удалить выделенную строку.
-  ○ **Clear** – очистить все данные.
-  ○ **Symmetrical** – симметричная характеристика.
-  ○ **Non-symmetrical** - несимметричная характеристика.
-  ○ **Обновить** графики.

-  • **“Table”**. Показывает “pwl” параметр в виде таблицы для редактирования.
  - Выберите симметричный или несимметричный тип PWL параметра, используя кнопки инструментальной панели. Заметьте, что первый параметр **From** равен нулю для симметричного, и «-inf» для несимметричного параметра.
  - Выделите ячейку таблицы для редактирования числа. Можно вводить и редактировать только значения **To**: соответствующие значения **From** обновляются автоматически.
  - **Arrow** (стрелка) в первой колонке показывает выделенную строку. Чтобы выделить строку, щелкните в области PWL данных.
  - Чтобы создать новую строку, щелкните кнопку **Split** чтобы «разщепить» выделенную строку, или введите число вместо “inf” в последней строке и нажмите **Enter**. Данные будут отсортированы автоматически.
  - Чтобы удалить строку, выделите строку и щелкните кнопку **Remove**.
  - Чтобы обновить графики, нажмите **Enter** или щелкните кнопку **Refresh**.
-  • **“Add/remove”**. Закладка используется для добавление и удаления строк в заданном диапазоне.




-  ○ Добавить строки в диапазоне **From...To** с интервалом **Interval**.
-  ○ Удалить строки в диапазоне **From...To**.

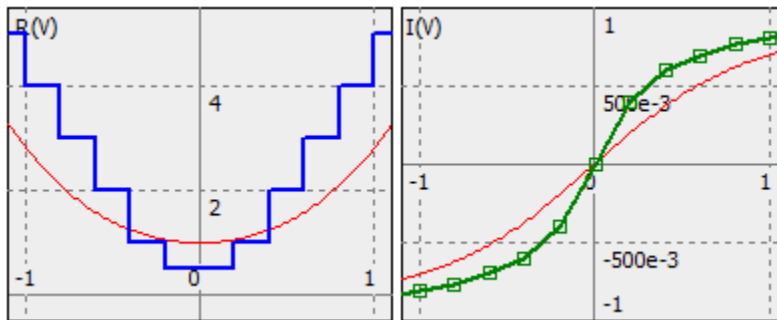
- **“Approximate”**. Аппроксимировать параметр «pwl» или кусочно-линейную характеристику заданной функцией.




Для этого:

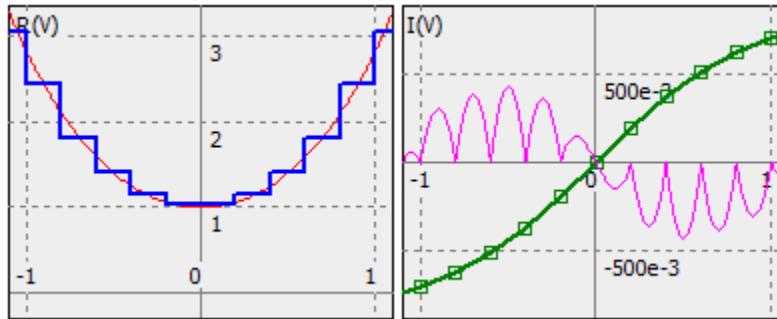
- Выберите функцию PWL параметра или характеристики (если доступна для данного компонента) из выпадающего списка. Например, для PWL модели резистора выберите R(V) или I(V).
- Введите  $f(x)$  как функцию параметра  $x$ . Например, для характеристики I(V) введите функцию:  

$$x/\sqrt{1+x^2}$$
- Нажмите **Enter** или щелкните кнопку **Refresh**  , чтобы обновить графики. Если выбрана функция характеристики I(V), соответствующая функция параметра R(V) будет вычислена автоматически; если выбрана функция параметра R(V), функция характеристики I(V) будет вычислена автоматически. Функции будут показаны на графиках.




- Чтобы аппроксимировать функцию на существующих точках (строках таблицы), введите точки вручную или на закладке **Add/remove**. Вы также сможете добавлять и изменять эти точки позднее для улучшения точности аппроксимации.

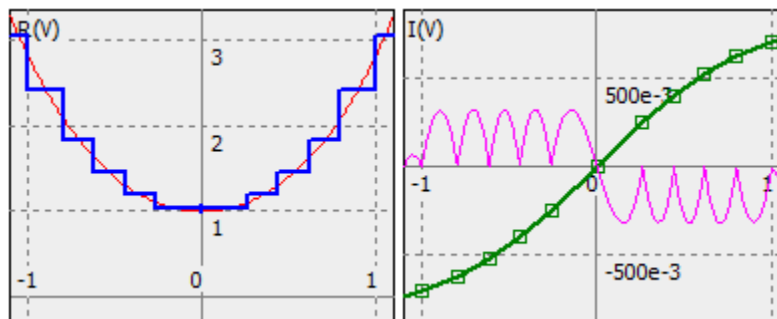
Щелкните кнопку **Approximate using existing points**  . Параметр «pwl» будет вычислен таким образом, чтобы PWL характеристика была максимально приближена к заданной функции. Результаты будут отображены на графиках.



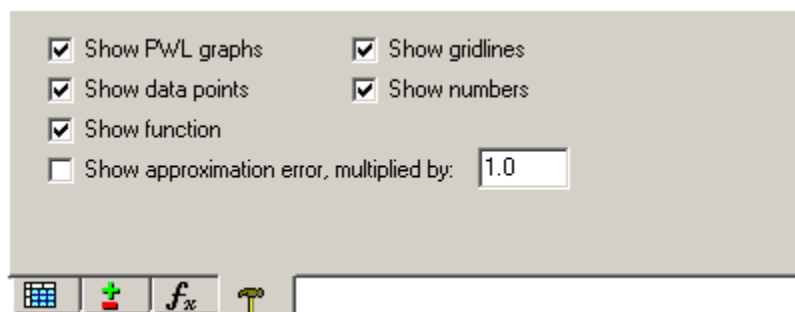
Подберите значения точек (величины **To** в таблице данных), или добавьте новые точки (строки таблицы) для необходимой точности аппроксимации. Установите флажок **Automatic update** для выполнения автоматической аппроксимации при любых изменениях в таблице данных или изменения аппроксимируемой функции.

- Чтобы аппроксимировать функцию с автоматическим выбором точек, задайте диапазон аппроксимации (**From, To**), и выберите количество интервалов в этом диапазоне (**Intervals**).

Щелкните кнопку **Approximate in the range** . Параметр “pwl» будет вычислен таким образом, чтобы PWL характеристика была максимально приближена к заданной функции. Одновременно интервалы между точками аппроксимации будут подобраны таким образом, чтобы минимизировать ошибку аппроксимации в заданном диапазоне. Результаты будут отображены на графиках.



- 🔧 • “Settings”. Закладка используется для конфигурации графиков.



Так как ошибка аппроксимации обычно очень маленькая, для вывода ошибки на графике используйте множитель.


## Работа с PWL источником

Модель **PWL** (“Piece-Wise Linear”) также используется для описания кусочно-линейного источника напряжения или тока.

Параметр “pwl” этой модели задает пары время/значение в следующем формате:

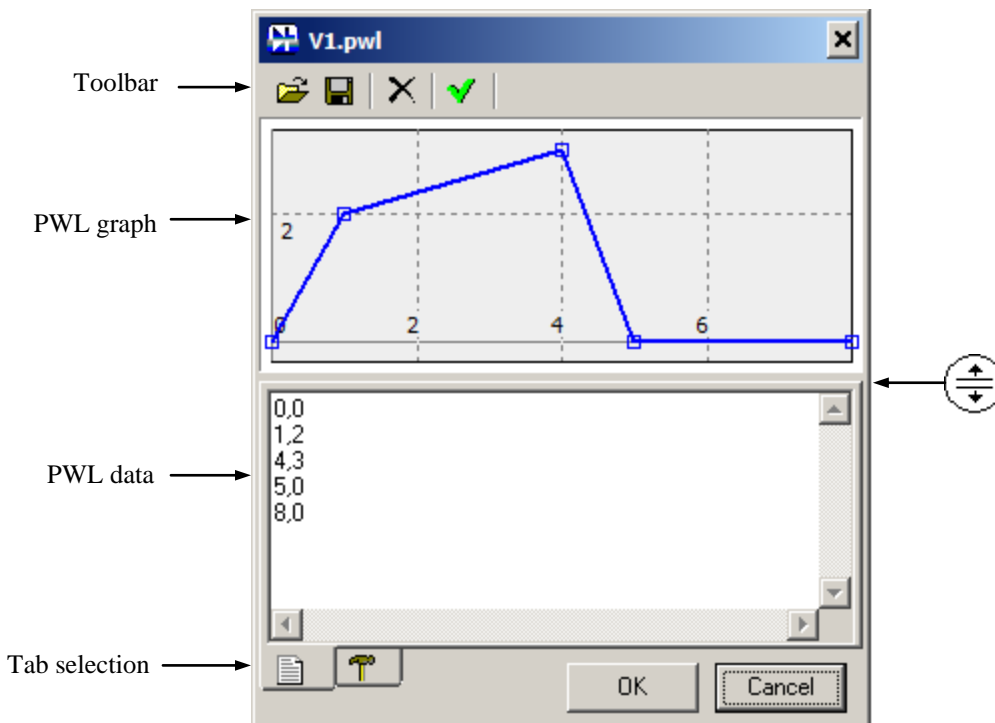
$$T1, V1, T2, V2, \dots, TN, VN$$

где пара  $t_i, v_i$  определяет величину сигнала (напряжение или ток) в заданное время. Величина сигнала между заданными точками интерполируется линейным образом. Значение сигнала до времени  $T1$  считается  $V1$ , после времени  $TN - VN$ . Величины  $T1...TN$  должны быть заданы в возрастающем порядке. Хотя сигнал задан на интервале  $T1...TN$ , он может быть задержан или циклически повторяться установкой параметров компонента “Delay” и “Cycle”.


Параметр “pwl” может быть отредактирован в диалоговом окне **PWL**. Щелкните по кнопке  справа от параметра «pwl», чтобы открыть диалоговое окно:









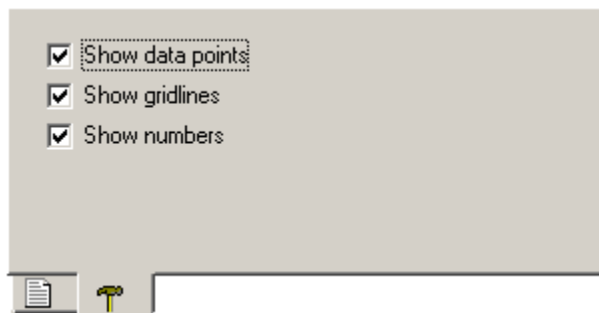
Типичный вид окна **PWL** для источников и его основные компоненты показаны ниже:



В верхней части окна находится график сигнала. В нижней части окна находятся 2 закладки для ввода данных и конфигурации.

- Переместите курсор поверх области «разделителя» , затем нажмите левую кнопку мышки и перетащите разделитель, чтобы изменить размер панелей.

- Щелчок правой кнопкой мышки на графике вызывает контекстное меню с соответствующими командами.
- Щелкните **ОК** чтобы применить изменения и закрыть окно, или **Cancel** чтобы закрыть окно без изменений.
- Щелчком по кнопкам инструментальной панели выполняются следующие операции:
  -  ○ **Прочитать** данные из файла (в формате “csv”).
  -  ○ **Сохранить** данные в файл (в формате “csv”).
  -  ○ **Очистить** все данные.
  -  ○ **Обновить** графики. Эта операция также сортирует введенные данные в возрастающем порядке и размещает каждую пару время/значение на отдельной строке.
-  • **“List”**. Показывает параметр “List” в виде текста для редактирования. Введите пары время/значение в произвольном порядке, затем щелкните кнопку **Обновить** для сортировки и отображения графика.
-  • **“Settings”**. Закладка используется для конфигурации графиков.





## Работа с моделью List

Модель **List** описывает последовательность включения компонента Switch и логический сигнал компонента Logic Generator.


Параметр “List” этой модели задает пары время/значение в следующем формате:

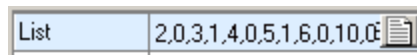
$$T1, S1, T2, S2, \dots, TN, SN$$

где пара  $t_i, s_i$  определяет состояние сигнала в заданное время:

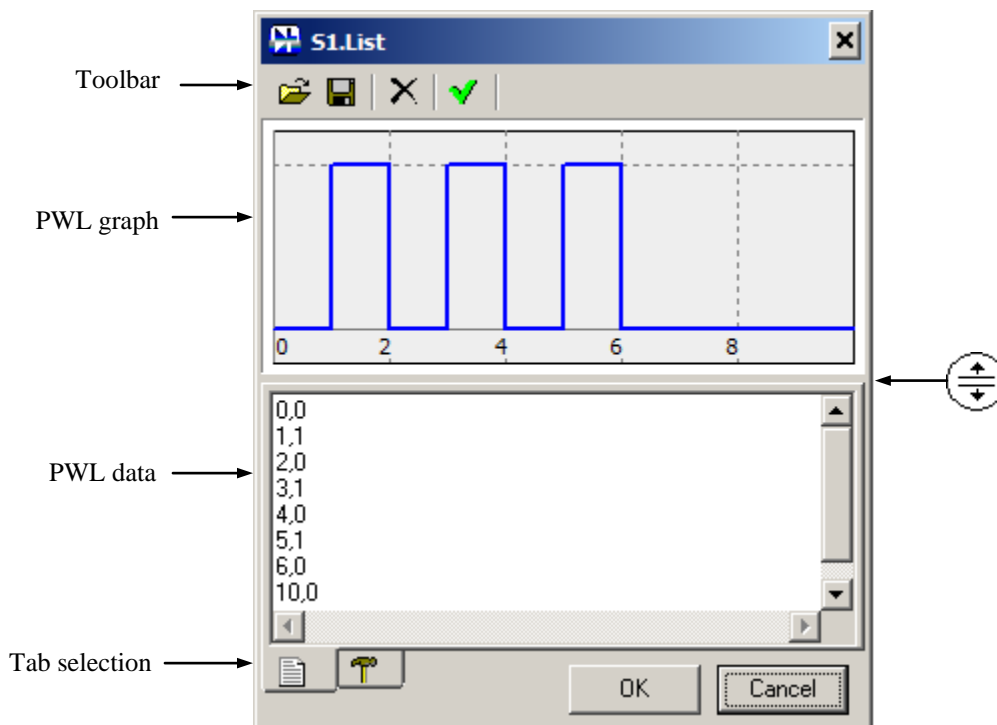
- Положительная величина соответствует состоянию On ключа или состоянию High логического генератора;
- Ноль или отрицательная величина соответствует состоянию Off ключа или состоянию Low логического генератора;

Значение сигнала до времени  $T1$  считается  $S1$ , после времени  $TN - SN$ . Величины  $T1...TN$  должны быть заданы в возрастающем порядке. Хотя сигнал задан на интервале  $T1...TN$ , он может быть задержан или циклически повторяться установкой параметров компонента “Delay” и “Cycle”.








Параметр “List” может быть отредактирован в диалоговом окне **List**. Щелкните по кнопке  справа от параметра «List», чтобы открыть диалоговое окно:

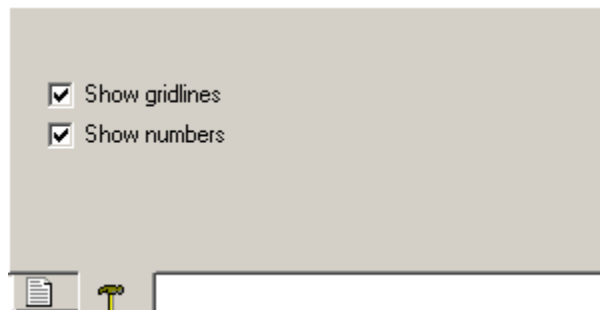


Типичный вид окна **List** и его основные компоненты показаны ниже:



В верхней части окна находится график сигнала. В нижней части окна находятся 2 закладки для ввода данных и конфигурации

- Переместите курсор поверх области «разделителя» , затем нажмите левую кнопку мышки и перетащите разделитель, чтобы изменить размер панелей.
- Щелчок правой кнопкой мышки на графике вызывает контекстное меню с соответствующими командами.
- Щелкните **ОК** чтобы применить изменения и закрыть окно, или **Cancel** чтобы закрыть окно без изменений.
- Щелчком по кнопкам инструментальной панели выполняются следующие операции:
  -  ○ **Прочитать** данные из файла (в формате “csv”).
  -  ○ **Сохранить** данные в файл (в формате “csv”).
  -  ○ **Очистить** все данные.
  -  ○ **Обновить** графики. Эта операция также сортирует введенные данные в возрастающем порядке и размещает каждую пару время/значение на отдельной строке.
-  • **“List”**. Показывает параметр «List» в виде текста для редактирования. Введите пары время/значение в произвольном порядке, затем щелкните кнопку **Обновить** для сортировки и отображения графика.
-  • **“Settings”**. Закладка используется для конфигурации графиков.




## Работа с моделью Table

Модель **Table** описывает look-up table для компонента Function.

Параметр “Table” этой модели задает пары время/значение в следующем формате:

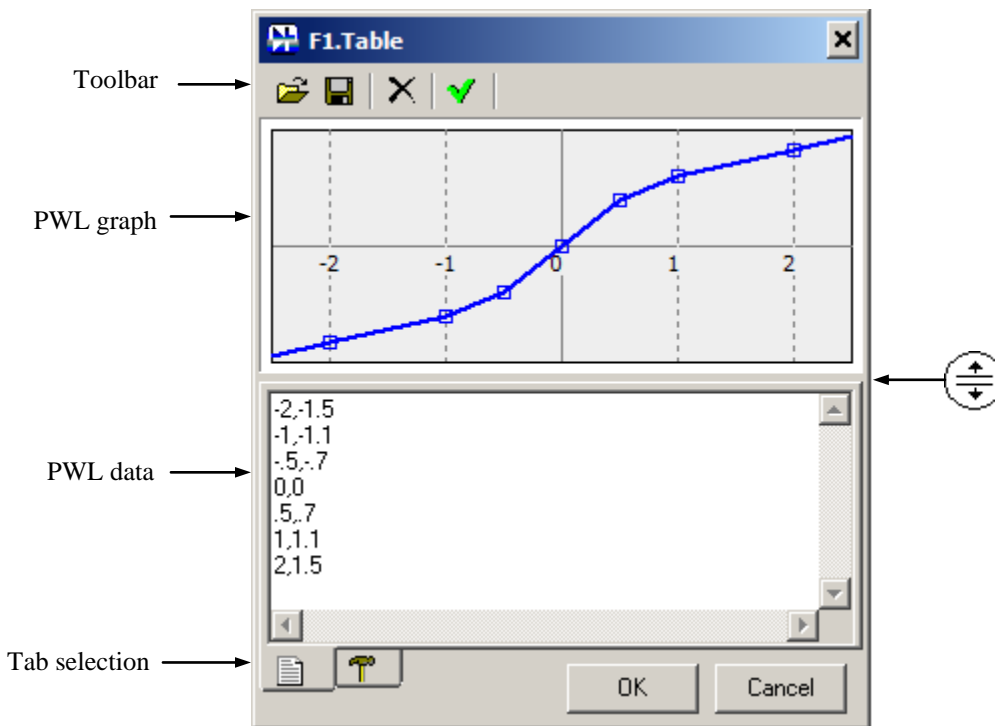
$$x_1, y_1, x_2, y_2, \dots, x_N, y_N$$

где пары  $x_i, y_i$  определяют входную величину ( $x$ ) и выходную величину ( $y$ ). Выходная величина между заданными точками  $X$  интерполируется линейным образом. Значение сигнала до  $x_1$  получается линейной экстраполяцией данных интервала  $x_1 \dots x_2$ , значение сигнала после  $x_N$  получается линейной экстраполяцией данных интервала  $x_{(N-1)} \dots x_N$ . Величины  $x_1 \dots x_N$  должны быть заданы в возрастающем порядке.


Параметр “Table” может быть отредактирован в диалоговом окне **Table**. Щелкните по кнопке  справа от параметра «Table», чтобы открыть диалоговое окно:









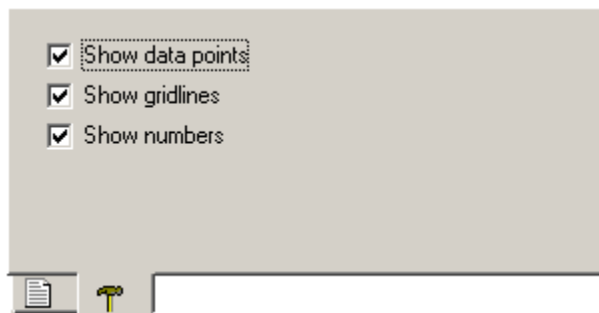
Типичный вид окна **Table** и его основные компоненты показаны ниже:



В верхней части окна находится график функции. В нижней части окна находятся 2 закладки для ввода данных и конфигурации.

- Переместите курсор вверх области «разделителя» , затем нажмите левую кнопку мышки и перетащите разделитель, чтобы изменить размер панелей.

- Щелчок правой кнопкой мышки на графике вызывает контекстное меню с соответствующими командами.
- Щелкните **ОК** чтобы применить изменения и закрыть окно, или **Cancel** чтобы закрыть окно без изменений.
- Щелчком по кнопкам инструментальной панели выполняются следующие операции:
  -  ○ **Прочитать** данные из файла (в формате “csv”).
  -  ○ **Сохранить** данные в файл (в формате “csv”).
  -  ○ **Очистить** все данные.
  -  ○ **Обновить** графики. Эта операция также сортирует введенные данные в возрастающем порядке и размещает каждую пару время/значение на отдельной строке.
-  • **“List”**. Показывает параметр “Table” в виде текста для редактирования. Введите пары X/Y в произвольном порядке, затем щелкните кнопку **Обновить** для сортировки и отображения графика.
-  • **“Settings”**. Закладка используется для конфигурации графиков.



## Работа с двухмерной (2D) моделью Table

Модель **Table** также описывает двухмерную (2D) look-up table для компонента Function-2.


Параметр “Table” этой модели задает выходное значение  $z$  как функцию входов  $x$  и  $y$  компонента в следующем формате:

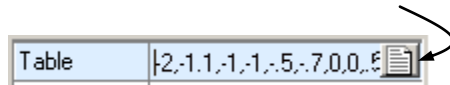
$$z_{11}, z_{12}, \dots, z_{1N}, z_{21}, z_{22}, \dots, z_{2N}, \dots, z_{M1}, z_{M2}, \dots, z_{MN}$$

где:

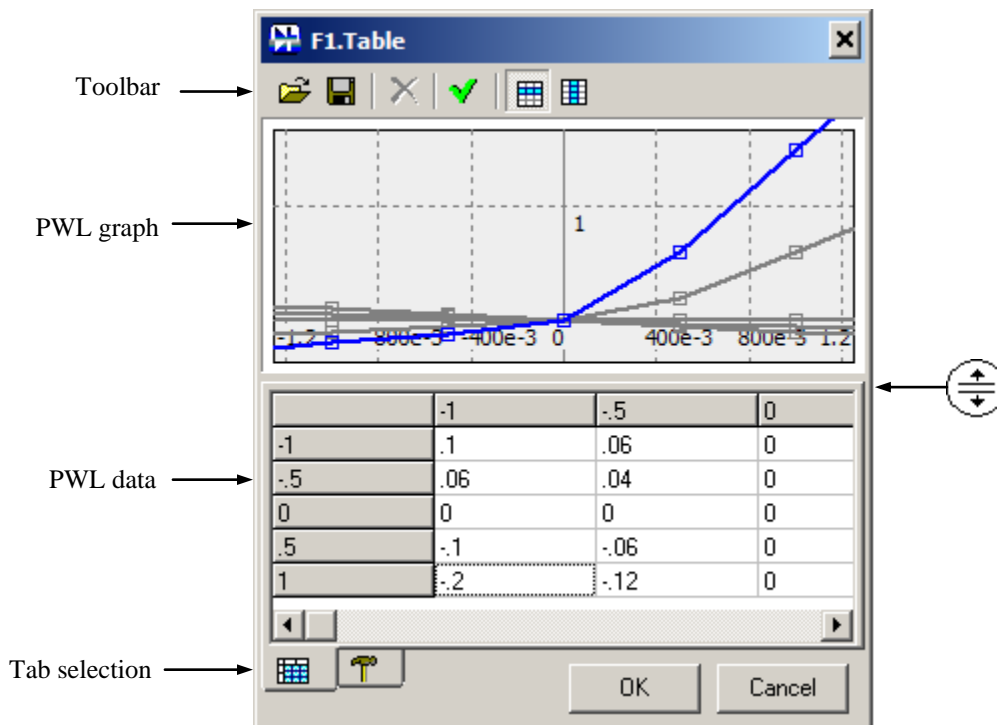
- $z_{ij}$  определяет выход функции для входных величин  $x_i$  и  $y_j$ ;
- $N$  – количество точек  $x$ , заданных параметром модели “X”;
- $M$  – количество точек  $y$ , заданных параметром модели “Y”.

Выходная величина между заданными точками  $x$  и  $y$  интерполируется линейным образом по обеим координатам. Значение сигнала до  $x_1$  получается линейной экстраполяцией данных интервала  $x_1 \dots x_2$ , значение сигнала после  $x_N$  получается линейной экстраполяцией данных интервала  $x_{(N-1)} \dots x_N$ . Аналогичное правило применяется к координате  $y$ .










Параметр “Table” может быть отредактирован в диалоговом окне **Table**. Щелкните по кнопке  справа от параметра «Table», чтобы открыть диалоговое окно:

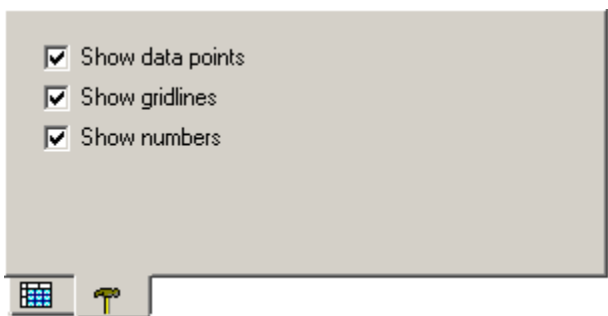


Типичный вид окна **Table** для двухмерной таблицы и его основные компоненты показаны ниже:



В верхней части окна находится график функции. В нижней части окна находятся 2 закладки для ввода данных и конфигурации.

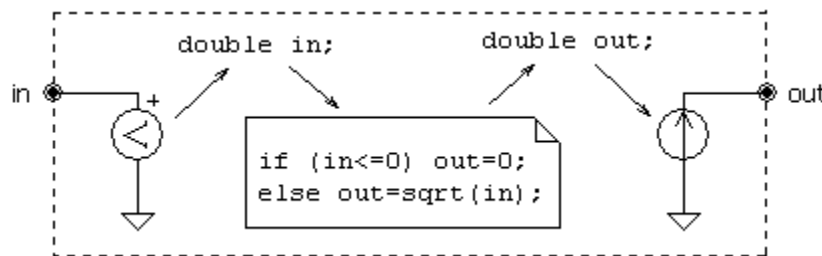
- Переместите курсор поверх области «разделителя» , затем нажмите левую кнопку мышки и перетащите разделитель, чтобы изменить размер панелей.
- Щелчок правой кнопкой мышки на графике вызывает контекстное меню с соответствующими командами.
- Щелкните **ОК** чтобы применить изменения и закрыть окно, или **Cancel** чтобы закрыть окно без изменений.
- Щелчком по кнопкам инструментальной панели выполняются следующие операции:
  -  ○ **Прочитать** данные из файла (в формате “csv”).
  -  ○ **Сохранить** данные в файл (в формате “csv”).
  -  ○ **Очистить** все данные.
  -  ○ **Обновить** графики.
  -  ○ Показать графики  $z(x)$ : каждая кривая представляет одну строку таблицы. Полное количество кривых равно  $m$ , выделенная кривая соответствует выделенной ячейке таблицы.
  -  ○ Показать графики  $z(y)$ : каждая кривая представляет один столбец таблицы. Полное количество кривых равно  $n$ , выделенная кривая соответствует выделенной ячейке таблицы.
-  • **“Table”**. Показывает параметр “Table” в виде таблицы для редактирования. Строки содержат данные для постоянной величины  $y$ , колонки – для постоянной величины  $x$ . Щелкните кнопку **Обновить** для отображения графика.
-  • **“Settings”**. Закладка используется для конфигурации графиков.



## Создание С-кода

Функционирование компонента **Code** с моделью **С** описывается программой, написанной на упрощенном языке **С (C-code)**. Программа интерпретируется симулятором во время анализа переходного процесса. Хотя интерпретация программы выполняется медленно, использование этой модели позволяет очень быстро делать изменения в программе. Когда программа окончательно отлажена, её можно откомпилировать и поместить в DLL для значительно более быстрой симуляции. См. главу *Создание DLL* для подробностей.

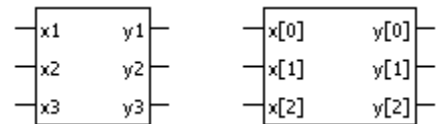
**Выполнение программы.** Пример выполнения программы приведен на следующей диаграмме:



Напряжение на входе компонента (“in”) измеряется вольтметром, и его величина присваивается переменной с тем же именем `in`. Затем интерпретируется код программы и вычисляется новое значение переменной `out`. И, наконец, напряжение, равное по величине значению переменной `out` выставляется на заземленный источник напряжения, подключенный к выходу компонента с таким же именем (“out”).

**Входы и выходы.** Как видно из диаграммы, входы имеют бесконечное входное сопротивление, а выходы являются заземленными источниками напряжения с нулевым выходным сопротивлением. Переменные, соответствующие входам и выходам, создаются автоматически во время инициализации анализа при  $t=0$ .

Значения входов и выходов доступны в программе через переменные, имеющие такое же имя, как и вывод компонента (например, `x1`, `x2`, `x3...`). Входы и выходы также могут быть связаны с элементами массива (`x[0]`, `x[1]`, `x[2]...`). В этом случае массив (`x[]`) соответствующего размера должен быть описан в инициализационной части программы.

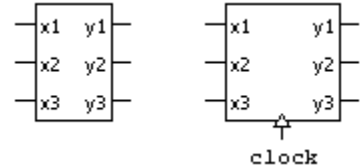


**Инициализационная часть программы** выполняется один раз в начале симуляции при  $t=0$ . Эта часть программы не является обязательной и может быть опущена (оставьте пустым параметр “Init”).

Инициализационная часть используется для присвоения начальных значений выходам (выходным переменным), описания и инициализации глобальных переменных и массивов. Глобальные переменные могут быть использованы для хранения глобальных параметров, вычисляемых только один раз в начале симуляции, и затем используемых в **основной части программы**. Также глобальные переменные могут хранить величины, вычисляемые на одном шаге симуляции, и затем используемые на следующем шаге. Если входы или выходы приписаны

к элементам массива, этот массив также должен быть описан в инициализационной части программы.

В **основной части программы** вычисляются выходные переменные, используя текущие значения входных переменных. Если у компонента не задан вывод *clock*, эта программа выполняется на каждом шаге симуляции. Если вывод *clock* существует, программа выполняется только при нарастающем фронте сигнала синхронизации на этом выводе.



Переменные, определенные в основной части программы, являются локальными, и существуют только во время выполнения основной программы. Наряду с глобальными и локальными переменными, в основной программе могут быть использованы следующие, доступные только для чтения (**read-only**), переменные:

Параметры компонентов (например,  $R1$ ,  $C2$ ,  $v.period$ , и т.п.),  
 $t$  – текущее время переходного процесса  
 $V(name)$  – напряжение на компоненте *name*  
 $I(name)$  – ток через компонент *name*  
 $P(name)$  – мощность на компоненте *name*

где *name* - это имя любого компонента в схеме. См. **Приложения 2, 3, 4** для подробного описания операторов, функций, и примеров языка С.


**Начальные условия (“IC”)** – это текстовая строка, которая может содержать код, присваивающий начальные значения выходным переменным и глобальным переменным, определенным в инициализационная части кода. Например:

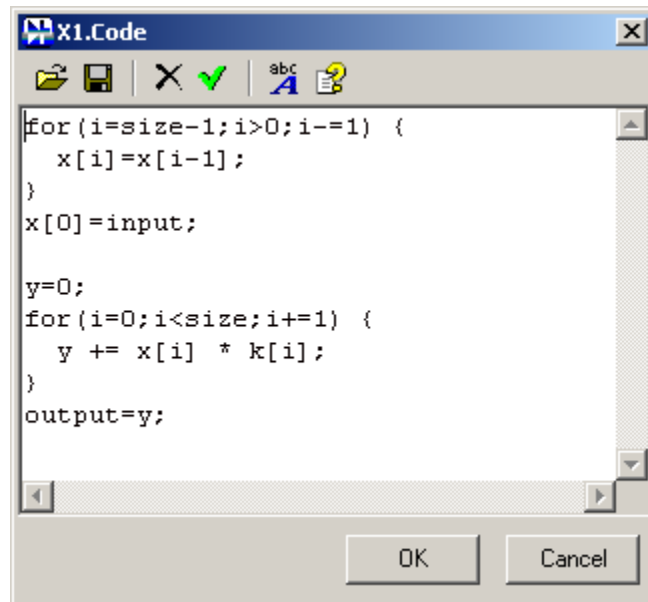
```
y1=1.2;y2=0;y3=2.345;integral=-4.19;counter=100
```

где  $y1$ ,  $y2$  и  $y3$  – выходные переменные;  $integral$  и  $counter$  – глобальные переменные. Строка **IC** может быть автоматически модифицирована некоторыми командами. Команда **Transient | Save IC** заполняет строку текущими значениями переменных.. Команда **Schematic | Tools**, страница **Initial Condition**: при установленном флажке **Clear C-code IC** эта строка будет очищена. Если строка не пустая, этот код будет исполнен после инициализационной части кода.











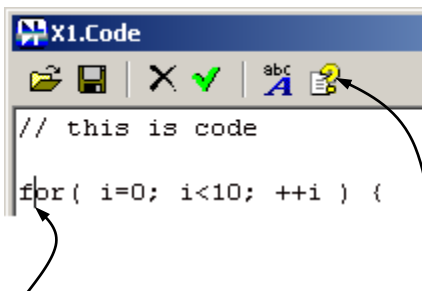
**Редактирование С-кода.** Для редактирования программы щелкните кнопку  в строке параметра. Появится следующее диалоговое окно:



Введите программу, затем щелкните кнопку **ОК**, чтобы применить изменения и закрыть окно, или **Cancel**, чтобы игнорировать изменения и закрыть окно.

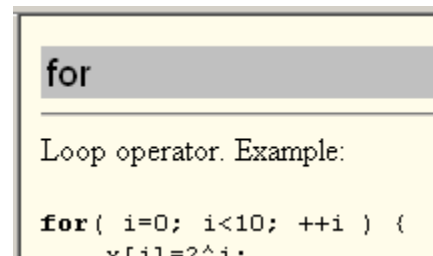
Следующие кнопки расположены на инструментальной панели окна:

-  • **Open code** – загрузить текст программы из файла.
-  • **Save code** – сохранить текст программы в файл.
-  • **Clear** –очистить текст.
-  • **Check** – проверить текст программы на наличие синтаксических ошибок.
-  • **Select font** – выбрать шрифт текста.
-  • **Help (F1)** – Помощь. Чтобы увидеть раздел помощи для определенного оператора, функции, или команды языка С, поместите курсор редактирования на соответствующее слово в программе и щелкните кнопку, или нажмите клавишу **F1**:



Place cursor on **for** operator

Click **Help** button  
or press **F1**

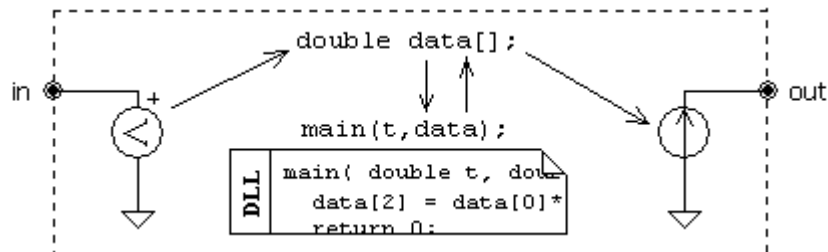


Help on **for** operator is displayed

## Работа с DLL

Функционирование компонента **Code** с моделью **DLL** описывается программой, написанной на стандартном языке **C**, скомпилированной любым доступным компилятором, и помещенной в DLL файл (Dynamic-linked library). Функции DLL будут вызываться во время симуляции переходного процесса. Программа, помещенная в DLL, будет выполняться значительно быстрее, чем написанная на языке **C** в модели **C**, однако любое изменение программы потребует создания новой **DLL**.

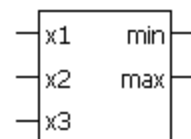
**Выполнение программы.** Пример выполнения программы из DLL приведен на следующей диаграмме:



Напряжение на входе компонента (“in”) измеряется вольтметром, и его величина присваивается соответствующему элементу массива `data`. Выполняется функция DLL `main` с указателем на этот массив в качестве параметра. При этом вычисляется новое значение выхода компонента “out”, которое помещается в соответственный элемент массива `data`. И наконец, напряжение, равное по величине этому значению, выставляется на заземленный источник напряжения, подключенный к выходу компонента.

**Входы и выходы.** Как видно из диаграммы, входы имеют бесконечное входное сопротивление, а выходы являются заземленными источниками напряжения с нулевым выходным сопротивлением. Массив `double data[]` используется для передачи входных/выходных величин в/из функции DLL. Размер массива равен `<число входов> + <число выходов>`. При вызове функции DLL, первые `<число входов>` элементов массива заполнены входными величинами, в том же порядке, как они показаны на изображении элемента (левая сторона, сверху-вниз). Выходные величины, вычисленные функцией DLL, помещаются в последующие `<число выходов>` элементов массива, в том же порядке, как они показаны на изображении элемента (правая сторона, сверху-вниз). Например, для компонента с тремя входами и двумя выходами, следующий код присвоит первому выходу минимальное значение на входах, и второму выходу – максимальное значение на входах:

```
data[3] = min( data[0], min(data[1],data[2]) );
data[4] = max( data[0], max(data[1],data[2]) );
```



**Функции.** Функции DLL должны быть определены следующим образом:

```
extern "C" __declspec (dllexport) int NAME(double t, double* data);
```

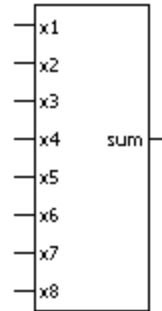
где:

*NAME* – произвольное имя функции, например `main`  
*t* – текущее время переходного процесса  
*data* – указатель на массив с входными/выходными данными

Функция возвращает ноль, если нет ошибок, или определенное пользователем ненулевое целое число – код ошибки. Код ошибки будет показан в сообщении об ошибке.

Например, следующая функция `sum` вычисляет сумму напряжений на восьми входах:

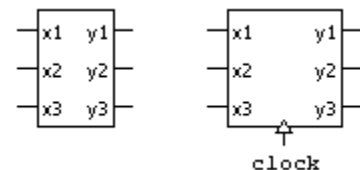
```
extern "C" __declspec (dllexport) int sum(double t, double* x)
{
    double y=0;
    for( int i=0; i<8; ++i ) {
        y += x[i];
    }
    x[8] = y;
    return 0;
}
```



**Функция инициализации** выполняется один раз в начале симуляции при  $t=0$ . Эта функция не является обязательной и может быть опущена (оставьте пустым параметр “Init”).

Функция инициализации используется для присвоения начальных значений выходам (выходным переменным), путем установки нужных значений соответствующим элементам массива `data[]`, а также описания и инициализации глобальных переменных DLL и массивов. Глобальные переменные могут быть использованы для хранения глобальных параметров, вычисляемых только один раз в начале симуляции, и затем используемых в **основной функции**. Также глобальные переменные могут хранить величины, вычисляемые на одном шаге симуляции, и затем используемые на следующем шаге. Следует иметь в виду, что если несколько компонентов используют одну DLL, то в память программы будет загружена только одна копия DLL, так что все компоненты будут использовать одни и те же глобальные переменные.

**Основная функция** вычисляет значения выходов, используя текущие значения входов. Если у компонента не задан вывод *clock*, эта функция выполняется на каждом шаге симуляции. Если вывод *clock* существует, функция выполняется только при нарастающем фронте сигнала синхронизации на этом выводе.



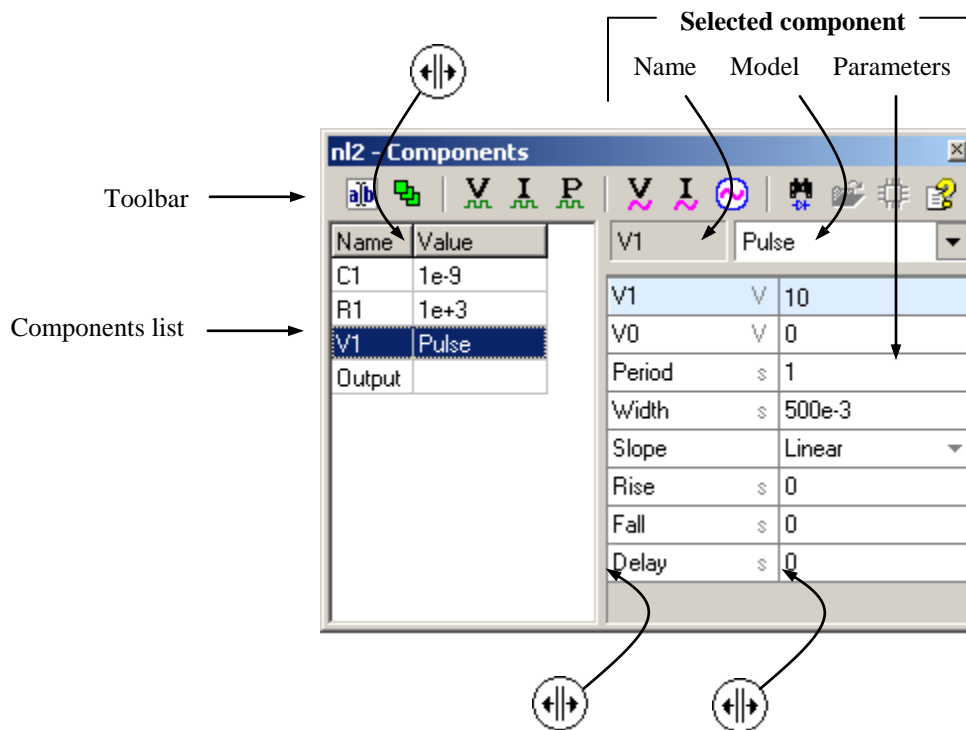
**Создание DLL.** Программа может быть скомпилирована и сохранена в виде DLL помощью любого доступного средства разработки C/C++. В настоящее время были протестированы Borland C++ Builder и Microsoft Visual C++. Примеры проектов для Borland C++ Builder 6 находятся в директории примеров `Examples/Components/x folder` полного загрузочного пакета NL5.


## Окно Компонентов (Components Window)

Чтобы открыть окно компонентов:

- Выберите команду меню **Window | Components**, или
- Нажмите горячую клавишу **F3**, или
- Дважды щелкните по компоненту в схеме.


Окно компонентов всегда показывает компоненты активной схемы. Обычный вид окна компонентов и его основные компоненты показаны ниже:

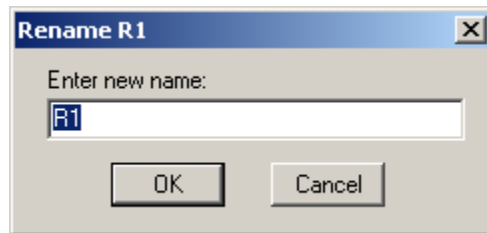


- **Инструментальная панель** дает быстрый доступ к часто используемым командам, относящимся к компонентам.
- **Список компонентов** показывает все компоненты и позволяет выбрать компонент для редактирования.
- **Область выделенного компонента** показывает **имя**, **модель** и **параметры** выделенного компонента и позволяет выбрать модели и редактировать параметры компонента.
- Поместите курсор мышки поверх области «разделителей» , затем нажмите левую кнопку мышки и перетащите разделитель, чтобы изменить размеры панелей и колонок.


## Инструментальная панель

Щелчок по кнопкам инструментальной панели выполняет следующие операции:

-  • **Rename a component.** Щелкните, чтобы переименовать выделенный компонент. Появится диалоговое окно **Rename**:







Введите новое имя компонента и щелкните по **OK**. Если такое имя уже существует, появится сообщение об ошибке: «*This name is used by another component*». Новое имя может состоять из любых букв и символов, однако рекомендуется использовать только буквы и цифры, и имя, начинающееся с буквы. В этом случае, если имя используется в формуле или функции, нет необходимости заключать его в кавычки. Имя не чувствительно к регистру. Если компонент был переименован, его имя будет автоматически модифицировано во всех случаях его появления: в имени компонента, в названии кривой, формуле или функции.

-  • **Set a group.** Щелкните, чтобы установить группу для выделенного компонента. Появится диалоговое окно **Group**:



Введите имя группы или выберите существующую группу из выпадающего списка и щелкните **OK**. Щелкните по кнопке **Groups**, чтобы открыть диалоговое окно **Groups** для расширенного обслуживания групп. Для удаления существующей группы («ungroup») откройте диалоговое окно **Group**, сотрите имя группы и щелкните по **OK**. См. Раздел *Группы* для подробностей.

-  • **Add transient voltage trace.** Добавить кривую напряжения переходного процесса для выделенного компонента.
-  • **Add transient current trace.** Добавить кривую тока переходного процесса для выделенного компонента.
-  • **Add transient power trace.** Добавить кривую мощности для выделенного компонента.
-  • **Add AC voltage trace.** Добавить кривую АС напряжения для выделенного компонента.

- **Add AC current trace.** Добавить кривую AC тока для выделенного компонента.
- **Set AC source.** Определить выделенный компонент как источник для AC анализа.
- **Find component.** Показать выделенный компонент на схеме. Компонент на схеме будет выделен (подсвечен) и отцентрован на экране.
- **Open subcircuit schematic file.** Открыть файл, в котором находится подсхема компонента. Доступно только для компонентов с моделью **SubCir**, если задано имя файла подсхемы.
- **Edit component.** Открыть диалоговое окно **Edit component**. Доступно только для “customized” компонентов (конфигурируемых пользователем).
- **Help (F1).** Показать раздел Помощи о выделенном компоненте.

## Список компонентов


Список компонентов показывает все компоненты схемы и позволяет выделить компонент для редактирования.

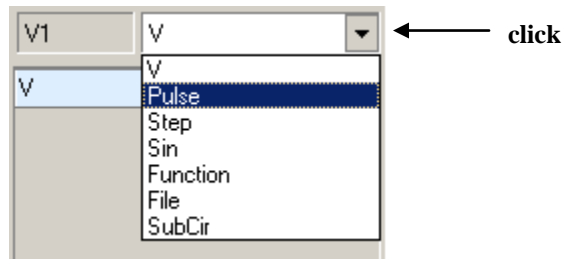
- Колонка **Name** показывает имя компонента.
- Колонка **Value** показывает либо первый параметр компонента, либо имя модели.
- Колонка **Group** автоматически отображается, если хотя бы один компонент имеет группу:

| Name | Value | Group   |
|------|-------|---------|
| R1   | 1e+3  | Group_R |
| R2   | 1e+3  | Group_R |
| R3   | 1e+3  |         |
| C1   | 1e-9  | Group_C |
| L1   | 1e-6  |         |
| V1   | 10    |         |

- Щелкните по списку, чтобы выделить компонент. Компонент, выделенный в этом списке, также будет выделен (подсвечен) на схеме.
- Нажмите **Enter** или дважды щелкните по компоненту для редактирования параметров компонента. В этом случае, если вы закончили редактировать параметры, нажав **Enter** или **Esc**, вы переключитесь обратно к списку компонентов.
- Нажимайте клавишу **Tab**, чтобы переключаться между списком компонентов и параметрами компонента.

## Выбор модели


Выберите **модель** компонента из выпадающего списка, щелкнув по кнопке 



## Редактирование параметров


Параметры редактируются в списке параметров. Щелкните по строке, чтобы выделить параметр, и используйте один из следующих методов для параметров разного типа.

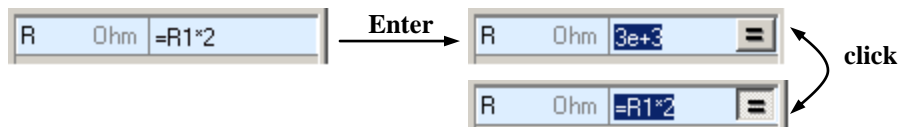
- **Число.**

- Введите численное значение: 

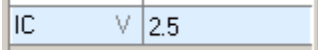
- Введите выражение и нажмите **Enter**. Выражение будет рассчитано и заменено числовым значением:



- Введите формулу и нажмите **Enter**. Формула будет вычислена и будет выведено ее численное значение. Щелкните кнопку , чтобы посмотреть и редактировать формулу и вернуться назад к выводу численного значения.



- **Начальные условия (Initial Conditions, IC)** (напряжение, ток, заряд).

- Введите численное значение: 

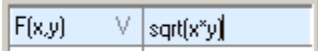
- Введите выражение и нажмите **Enter**. Выражение будет рассчитано и заменено числовым значением:




- Оставьте параметр пустым: 

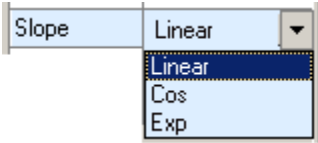
- **Функция.**

- Введите функцию, как выражение с допустимыми параметрами:




- **Список.**


- Щелкните по кнопке  и выберите значение из выпадающего списка:



- **Имя файла.**

- Введите имя файла: 

- Щелкните по кнопке  для выбора файла в диалоговом окне:





- **Текст.**


- Введите текст:

|     |              |
|-----|--------------|
| Cmd | R1=5k;C2=1n2 |
|-----|--------------|

- **PWL (кусочно-линейная функция).**


- Введите параметр вручную:

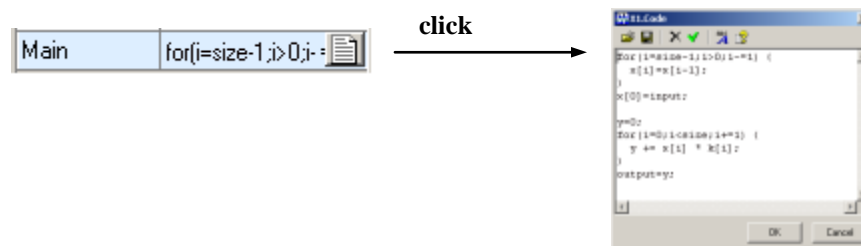
|     |       |
|-----|-------|
| pwl | 1,1,2 |
|-----|-------|

- Щелкните по кнопке , чтобы редактировать PWL в диалоговом окне (См. главу *Работа с PWL*):



- **С-программа.**

- Щелкните по кнопке , чтобы редактировать программу в диалоговом окне (См. главу *Редактирование С-кода*):

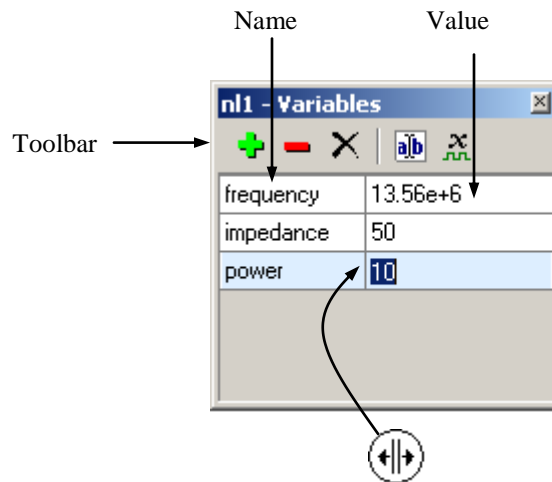



## Окно переменных (Variables Window)

Чтобы открыть/закрыть окно переменных:

- Выберите команду меню **Window | Variables Window | Variables**, или
- Нажмите горячую клавишу **F4**

Окно переменных откроется автоматически, когда открывается схема из файла, если схема имеет определенные переменные. Окно переменных всегда показывает переменные активной схемы. Обычный вид окна переменных и его основных компонентов показан ниже:

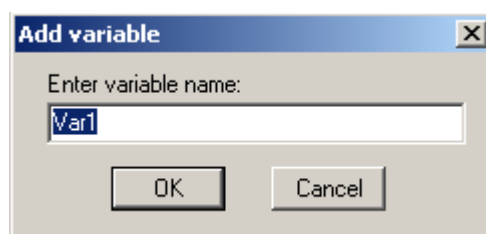


- **Инструментальная панель** предоставляет быстрый доступ к командам, относящимся к переменным.
- Переменные показаны в колонках Имя/Значение.
- Поместите мышку поверх «разделителя» , затем нажмите левую кнопку мышки и перетащите разделитель, чтобы изменить размер колонок.

## Панель инструментов

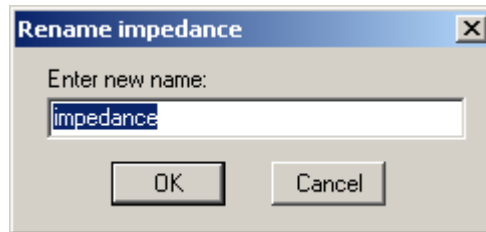
Щелкните по кнопкам панели, чтобы выполнить следующие операции:

- **Add a variable.** Щелкните, чтобы добавить переменную. Появится окно **Add variable**:



Введите имя переменной и щелкните **OK**. Если новое имя уже существует, появится сообщение об ошибке «*This name is used by another component*».

- **Remove a variable.** Удалить выделенную переменную.
- **Delete all variables.** Удалить все переменные.
- **Rename a variable.** Щелкните, чтобы переименовать переменную. Появится окно Rename:

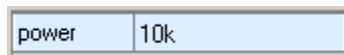


Введите новое имя переменной и нажмите **OK**. Если такое имя уже существует, появится сообщение об ошибке «*This name is used by another component*». Новое имя может состоять из любых букв и символов, однако рекомендуем вам использовать только буквы и цифры, и использовать имя, начинающееся с буквы. В этом случае, если имя используется в формуле или функции, нет необходимости заключать его в кавычки. Имена не чувствительны к регистру. Если переменная была переименована, ее имя будет автоматически модифицировано: во всех приложениях, в названиях кривых, в формулах и функциях.

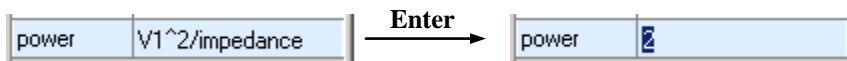
- **Add variable trace.** Добавить кривую переходного процесса для переменной.

## Редактирование переменных

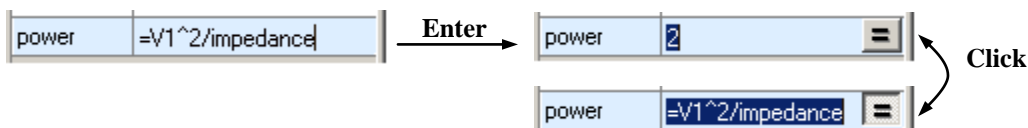
- Введите численное значение:



- Введите выражение и нажмите **Enter**. Выражение будет рассчитано и заменено числовым значением:

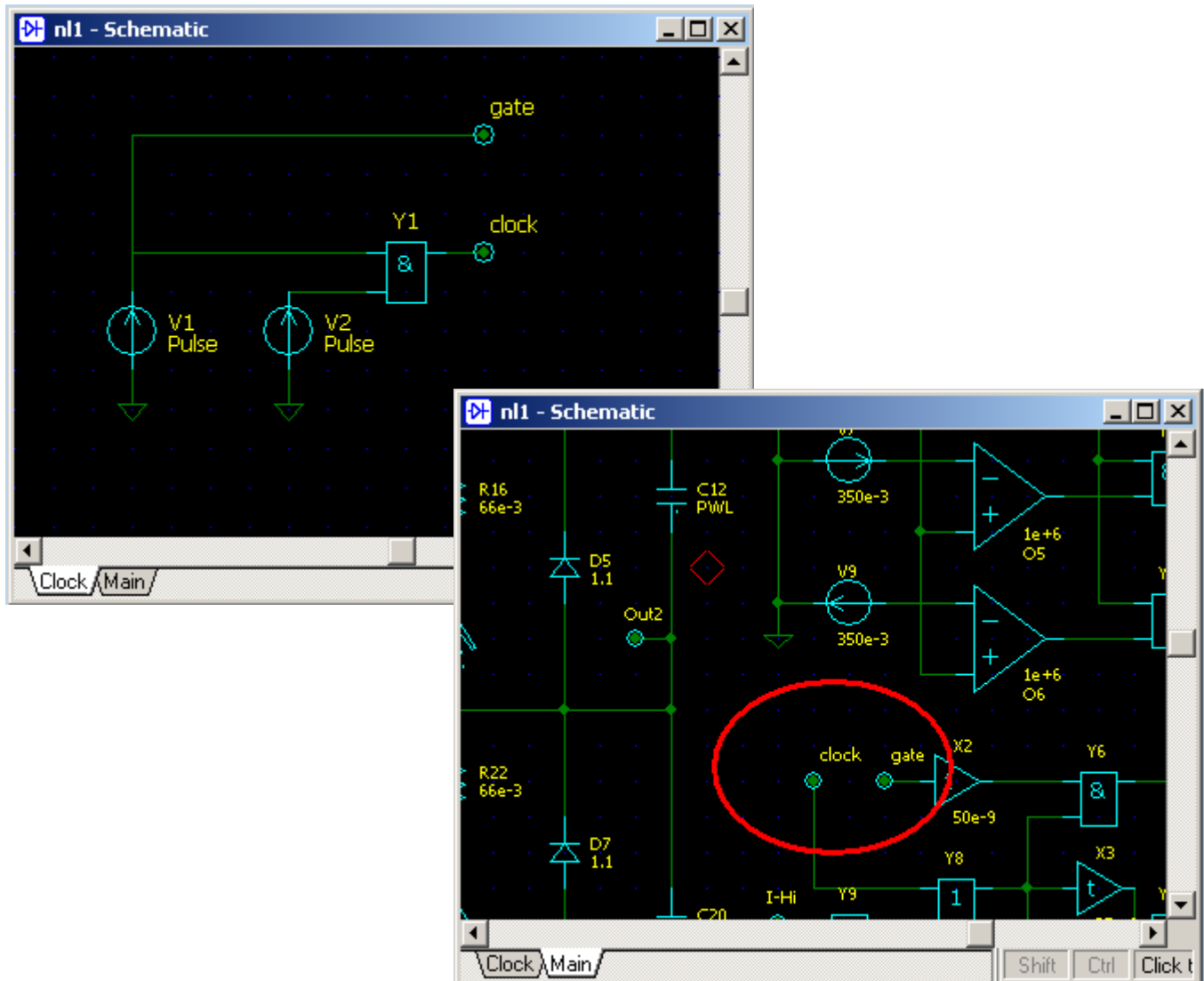


- Введите формулу и нажмите **Enter**. Формула будет вычислена и будет выведено ее численное значение. Щелкните кнопку **=**, чтобы посмотреть и редактировать формулу и вернуться назад к выводу численного значения:



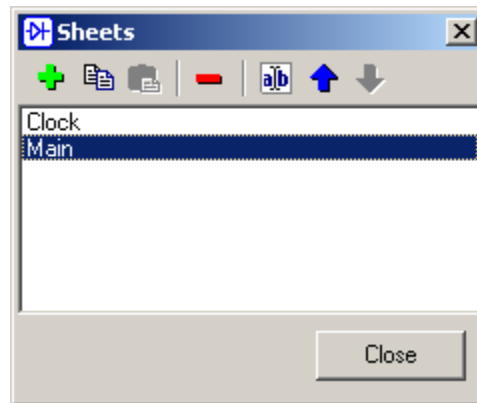
## Листы (Sheets)

Схема может содержать несколько листов. Электрические соединения между листами схемы могут быть сделаны с помощью этикеток (labels) и функций. В следующем примере на листе «Clock» находится генератор импульсов, а лист «Main» содержит основную схему. Этикетки «clock» и «gate» поддерживают электрическое соединение между листами:



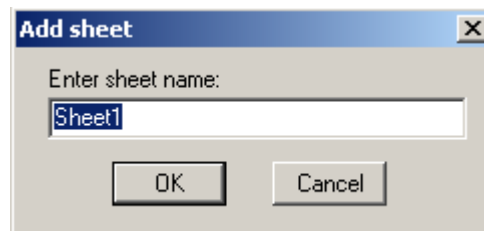
Существующие листы показаны в области **Sheet selection** (выбор листа) окна схемы. Щелчок правой клавиши мышки в области выбора дает доступ к соответствующим командам контекстного меню: **Add, Rename, Copy, Paste, Delete**. Можно работать с листами и в диалоговом окне **Sheets**.

Чтобы открыть диалоговое окно **Sheets**: выберите команду **Schematic | Sheets** или щелкните правой клавишей мышки по области **Sheet selection** и выберите команду **Sheets** из контекстного меню:



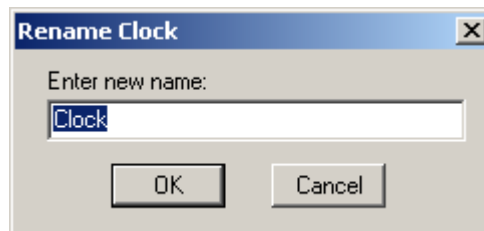
Выберите лист в списке и щелкните по кнопке панели для выполнения следующих операций:

- + • **Add** – добавить новый лист. Появится окно **Add sheet**:



Введите имя листа и щелкните по **ОК**.

- **Copy** – скопировать лист в буфер обмена.
- **Paste** – вставить лист из буфера обмена.
- **Remove** – удалить лист.
- **Rename** – переименовать лист. Появится окно **Rename**:

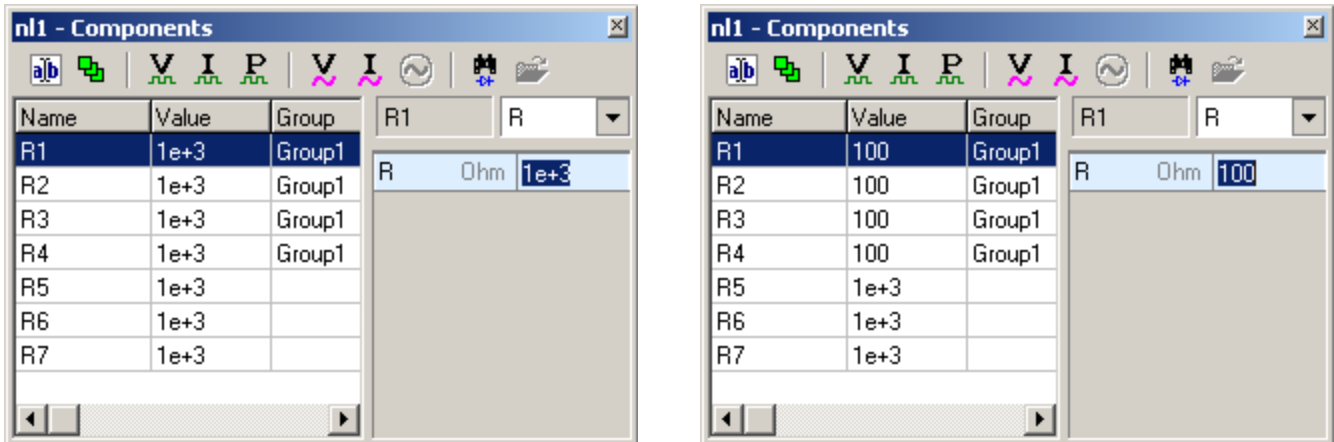



Введите новое имя листа и щелкните по **ОК**.

- **↑** – Передвинуть лист вверх (или влево в области выбора листа).
- **↓** – Передвинуть лист вниз (или вправо в области выбора листа).

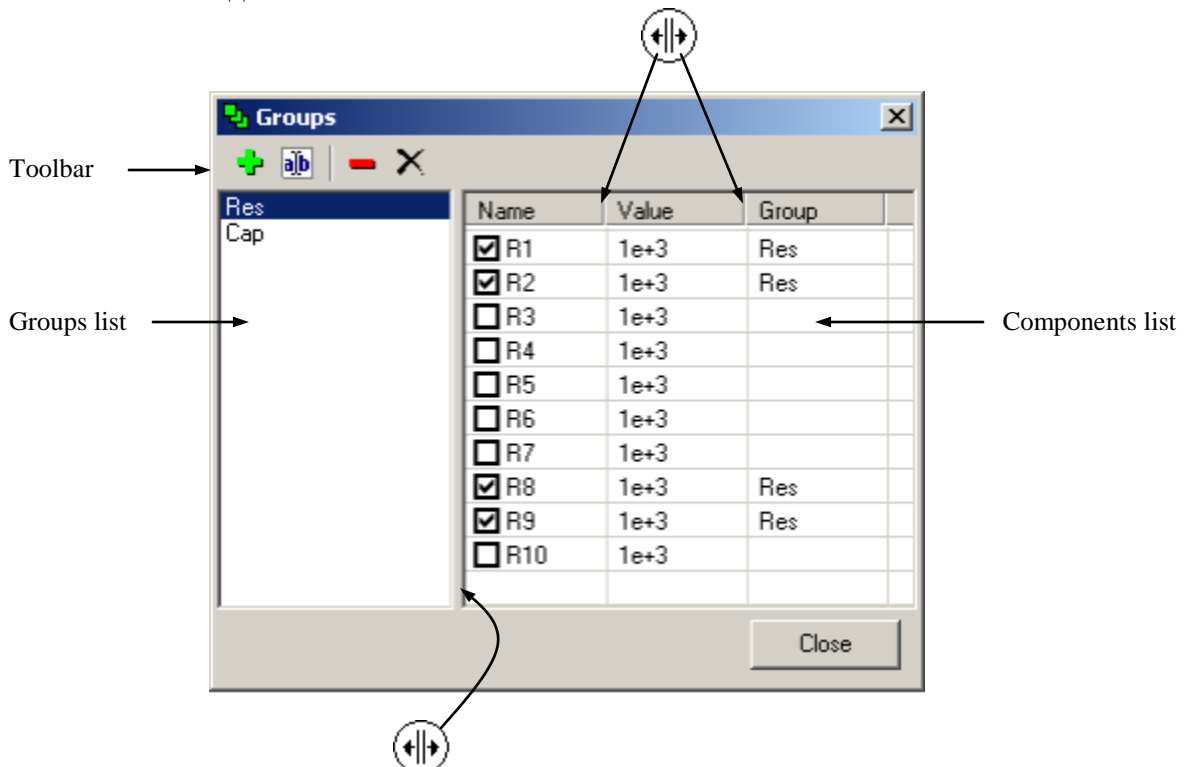
## Группы


Группа — это набор компонентов, которые всегда имеют одинаковую модель и параметры. Когда модель или параметр любого компонента в группе меняется, все другие компоненты также автоматически меняются. Например, резисторы R1...R4 принадлежат к группе **Group1**. Изменение значения R1 с  $1e+3$  на 100 автоматически изменит все резисторы в этой группе.



Чтобы прикрепить один компонент к новой или существующей группе, выберите компонент в окне **Components** и щелкните по кнопке . Чтобы работать со всеми группами или добавить несколько компонентов в группу, используйте диалоговое окно **Groups**.

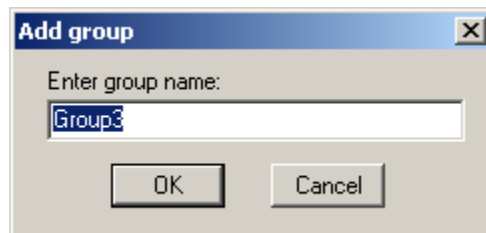
Чтобы открыть диалоговое окно **Groups**, выберите команду **Schematic | Groups**. Типовой вид и основные компоненты диалогового окна показаны ниже:



- Существующие группы показаны в **списке групп**. Щелкните по имени группы, чтобы отобразить компоненты.
- Компоненты показаны в **списке компонентов**. Компоненты показаны либо рядом с выделенной группой (имеют установленный флажок в списке и выделены на схеме), либо они имеют тот же тип и могут быть соотнесены с выделенной группой.
- Поместите указатель мышки поверх «разделителя» окон , затем нажмите левую кнопку мышки и перетащите его, чтобы изменить размер колонок.

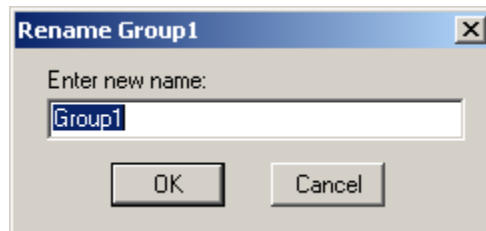
Щелкните по кнопкам панели, чтобы выполнить следующие операции:

-  • **Add** – добавить новую группу. Появится окно **Add group**:





Введите имя группы и щелкните по **ОК**. Новая группа будет добавлена в список групп, а **все** компоненты отобразятся в списке компонентов. Выделите компонент, чтобы назначить ему группу. Когда хотя бы один компонент назначен группе, **только** компоненты того же типа будут отображены в списке.

-  • **Rename** - переименовать выделенную группу. Появится окно **Rename group**:




Введите новое имя группы и щелкните по **ОК**.

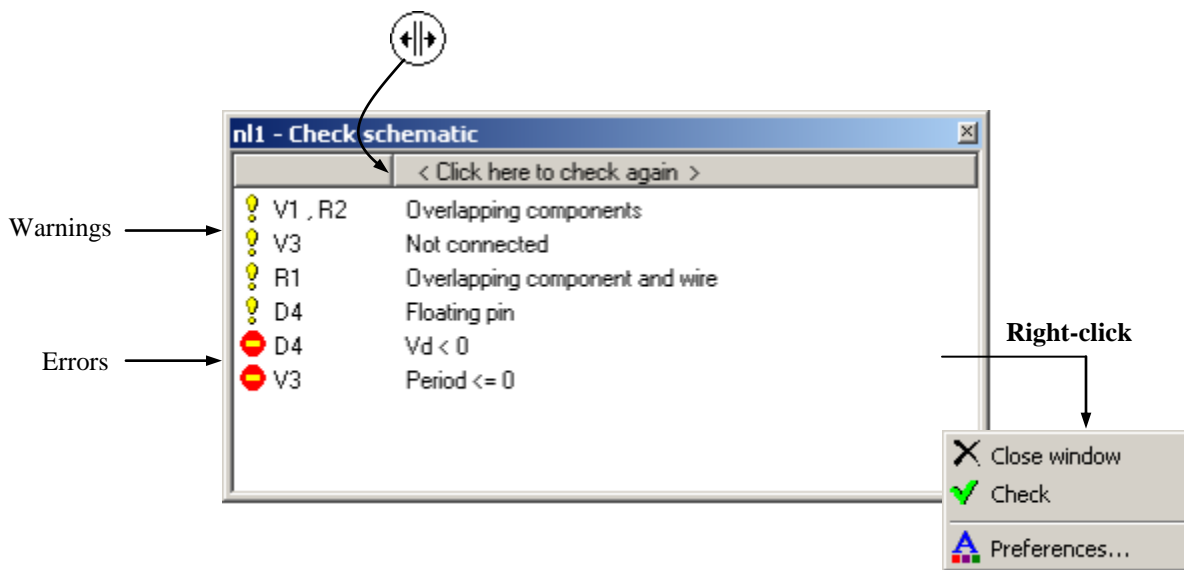
-  • **Remove** – удалить группу. Удаляется только имя группы: компоненты, входившие в группу, удалены **не будут!**
-  • **Delete** – удалить все группы. Удаляются только имена групп: компоненты, входившие в группы, удалены **не будут!**

## Проверка схемы

Команда **Check Schematic** выполняет проверку схемы на предмет потенциальных проблем, а параметры компонентов на наличие ошибок. Чтобы проверить схему:

- Выберите команду меню **Schematic | Check**, или
- Щелкните кнопку **Check Schematic** 
- Также проверка схемы происходит автоматически при выполнении анализа переходного процесса или АС анализа.



Обычный вид окна **Check Schematic** и его основных компонентов показан ниже:




- Сообщения с иконкой предупреждения ( ! ) отмечают потенциальные проблемы со схемой и не запрещают запуск анализа переходного процесса или АС анализа. Щелкните по линии сообщения, чтобы увидеть проблемные элементы схемы: они будут выделены и выведены в центре экрана. В настоящее время обнаруживаются следующие потенциальные проблемы:
  - **Floating pins.** Плавающие выводы. Один или больше выводов компонента не присоединены к другим компонентам, проводникам или земле.
  - **Non-connected components.** Не присоединенные компоненты. Все выводы компонента не присоединены к другому компоненту, проводнику или земле.
  - **Overlapping components and wires.** Перекрытие компонентов или проводников. Изображение элемента схемы (компонент, проводник, земля) перекрывается другим элементом схемы. Это может приводить к непредусмотренным соединениям или отсутствию соединения.
  - **Possibly floating schematic.** Возможное «плавание» схемы. Схема не имеет какой-либо земли или заданного потенциала (источника напряжения или этикетки) на ней. Это может привести к проблемам сходимости.



Перейдите на страницу **Warnings** окна **Preferences**, чтобы убрать все предупреждения или выделенные типы предупреждений из отображаемых. После этого предупреждения все еще останутся в рапортах журнала Transient/AC log, они могут быть прочитаны выбором команды **Transient | Log or AC | Log**.

- Сообщения с иконкой **ошибки** (  ) отмечают ошибки в параметрах компонента, такие как значения вне допустимого диапазона или ошибки в формуле. Если ошибки есть, анализ переходного процесса и AC анализ не будет выполнен. Щелкните по линии сообщения для выбора компонента с ошибкой. Ошибки также отображаются в журнале **Transient/AC log** и могут быть прочитаны с помощью команды **Transient | Log or AC | Log**.
- Щелкните правой клавишей мышки по окну, чтобы увидеть контекстное меню с соответствующими командами.
- Поместите курсор мышки поверх «разделителя»  , затем нажмите левую кнопку мышки и растащите колонки.

## Инструменты схемы

Чтобы открыть диалоговое окно Schematic Tools выберите команду меню **Schematic | Tools** или щелкните по кнопке **Schematic tools** . Выберите нужную страницу в списке выбора.

### Renumber (перенумерация)

Перенумеровать компоненты схемы. Выберите опции **Order** и **Names**, посмотрите пример перенумерации в окне **Example**, щелкните кнопку **Execute** для выполнения.

«Числовое» (**numerical**) имя — это имя, начинающееся с буквы, за которой следует число. Иначе имя рассматривается как «текст» (**text**). Предопределенное (автоматически) имя компонента — числовое. Например:

R123, C2 - числовое  
V12V, Rout - текст

### Initial Conditions (начальные условия)

Установить начальные условия (IC) выделенных типов компонентов в заданные значения. Выделите типы компонентов и IC значения, щелкните кнопку **Execute**, чтобы выполнить. Щелкните кнопки **Check all** и **Uncheck all** для установки/снятия выбора всех компонентов.

### Clean Up (чистка)

Почистить схему (привести в порядок). Выберите опции, щелкните кнопку **Execute** для выполнения. Щелкните **Check all** и **Uncheck** кнопки, чтобы установить/снять все опции.

### Formulas (формулы)

Заменить все формулы их значениями. Выберите опцию, щелкните кнопку **Execute** для выполнения.

### Parameters (параметры)

Установить выделенные параметры всех компонентов в заданное значение. Выберите параметры, введите новое значение параметра или выберите из выпадающего списка, щелкните кнопку **Execute** для выполнения.

- **Set diodes.** Устанавливает параметр **Vd** всех диодов и стабилитронов (zener) и/или **Vbe** параметр всех транзисторов.

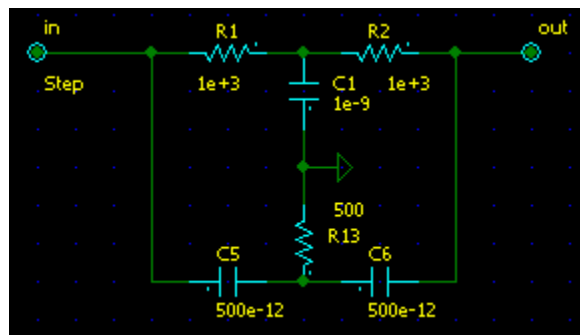
- **Set period.** Устанавливает период для всех источников синусоидального (**Sin**) и импульсного (**Pulse**) напряжения и тока и импульсных ключей. Значения **Width**, **Rise**, **Fall** и **Delay** будут также изменены пропорционально периоду..

## Transform (преобразование)

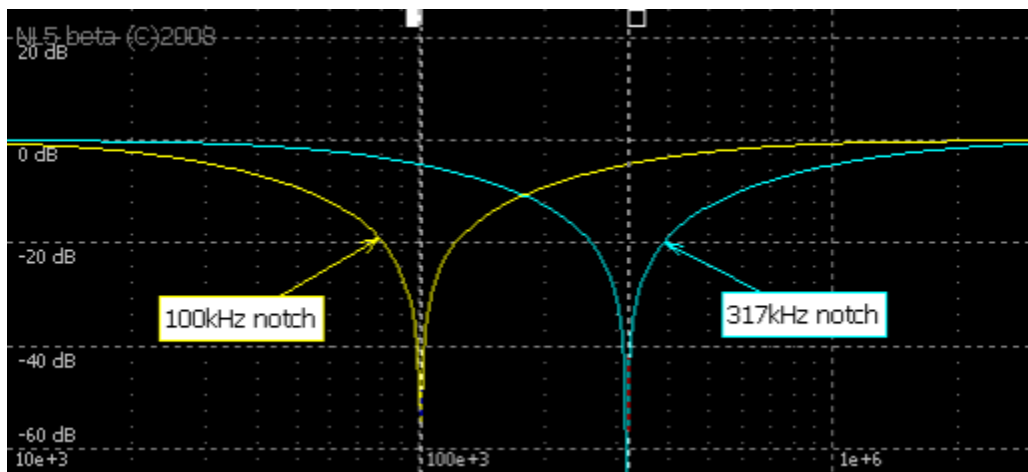
**Frequency (частота).** Преобразует частотный отклик схемы изменением значений R, C и L. Введите значения f1 и f2 или установите курсор на частоты f1 и f2 графика AC (активный курсор на f2). Выберите одну из следующих опций, затем щелкните по кнопке **Execute** для выполнения:

- **R = const.** C и L изменятся следующим образом:  $C=C*f1/f2$ ,  $L=L*f1/f2$
- **C = const.** R и L изменятся следующим образом:  $R=R*f1/f2$ ,  $L=L*(f1/f2)^2$
- **L = const.** R и C изменятся следующим образом:  $R=R*f2/f1$ ,  $C=C*(f1/f2)^2$

Пример: частота подавления фильтра смещается с 317 kHz к 100 kHz, сохраняя C = const.



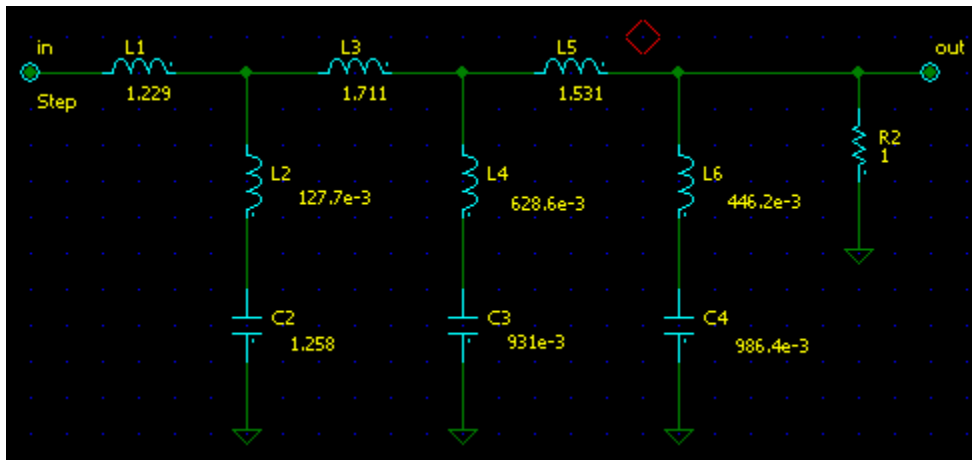
Вычислите частотную характеристику (AC). Установите курсор на частоту подавления и 100 kHz (активный курсор), откройте диалоговое окно **Schematic Tools**, страницу **Transform**, выберите C = const, щелкните **Execute**. Вычислите новую частотную характеристику. Частота подавления сместится к 100 kHz.



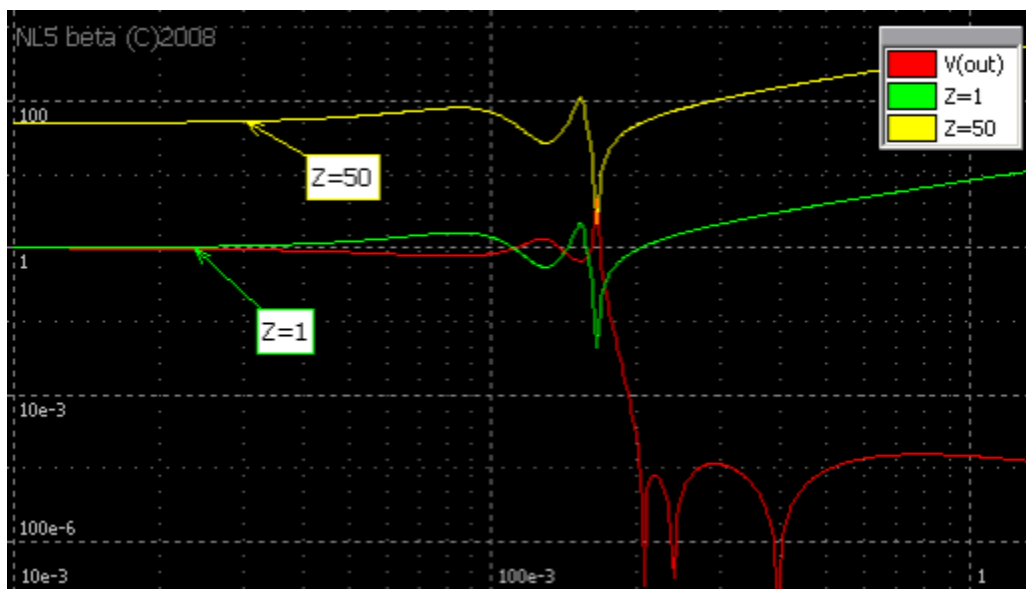
**Impedance (импеданс).** Преобразует импеданс схемы от  $r_1$  до  $r_2$  изменением значений  $R$ ,  $C$  и  $L$ . Введите значения  $r_1$  и  $r_2$ , затем щелкните по кнопке **Execute** для выполнения. Значения  $R$ ,  $C$  и  $L$  изменятся следующим образом:

$$\begin{aligned} R &= R \cdot r_2 / r_1 \\ C &= C \cdot r_1 / r_2 \\ L &= L \cdot r_2 / r_1 \end{aligned}$$

Пример: изменение характеристического импеданса фильтра с 1 до 50 Ohm.

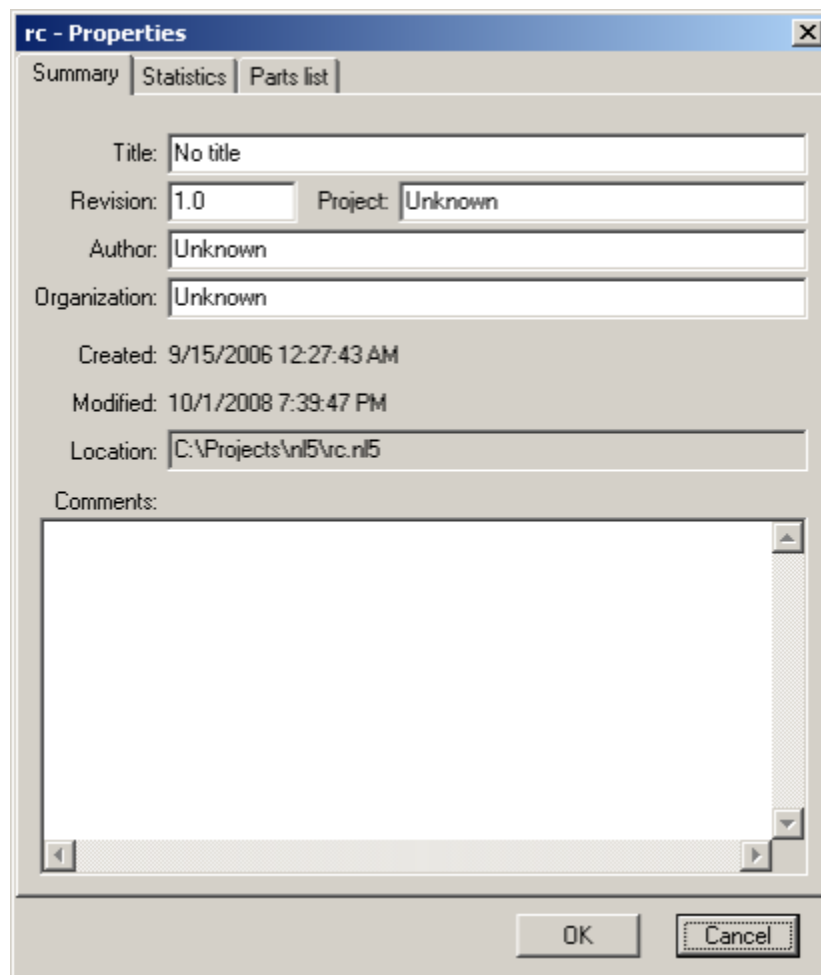


Вычислите частотную характеристику и входной импеданс. Откройте диалоговое окно **Schematic Tools** на странице **Transform**, введите  $r_1 = 1 \text{ Ohm}$  и  $r_2 = 50 \text{ Ohm}$ , щелкните **Execute**. Вычислите новую частотную характеристику и входной импеданс. Частотная характеристика останется такой же, входной импеданс изменится, как требовалось.



## Свойства (Properties)

Выберите команду **File | Properties**. Появится диалоговое окно **Properties**:

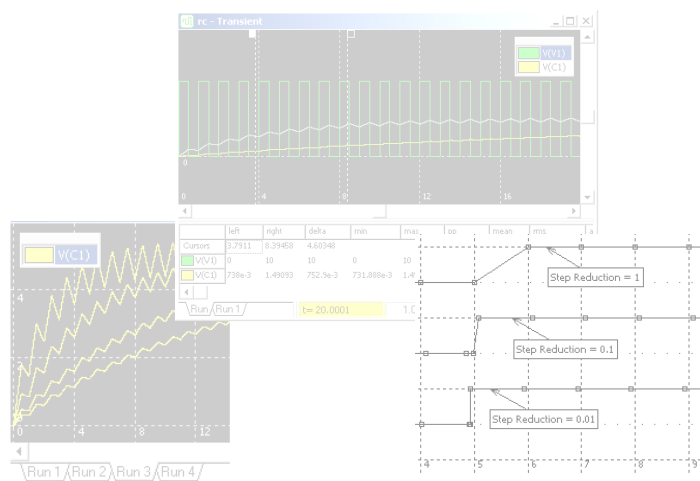


Страница **Summary** показывает основную информацию о документе и файле. Поля **Author** и **Organization** новой схемы заполняются значениями, заданными на странице **Document** диалогового окна **Preferences**. Большую часть полей можно редактировать.

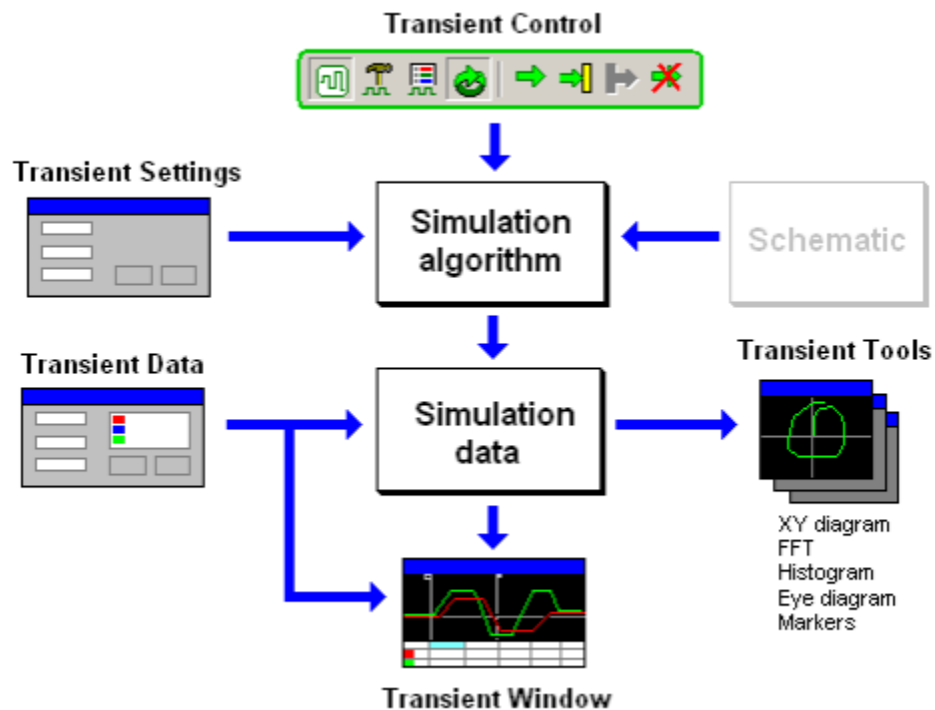
Страница **Statistics** показывает статистическую информацию схемы.

Страница **Parts list** показывает список компонентов в кратком или развернутом формате. Щелкните кнопку **Copy to clipboard**, чтобы скопировать список в буфер обмена.

# IV. Анализ переходного процесса (Transient)



Следующая упрощенная диаграмма поясняет процесс симуляции переходного процесса:



**Алгоритм симуляции** конфигурируется в диалоговом окне **Transient Settings** и управляется командами **Transient Control** (Основное Меню и инструментальная панель). Результаты симуляции схемы запоминаются в данных симуляции и одновременно отображаются в виде графика в **Transient Window** (окно переходного процесса). Окно **Transient Data** используется для конфигурирования того, какие данные симуляции следует сохранить, и как данные должны отображаться. Также данные могут использоваться в **Transient Tools** (инструменты переходного процесса), которые позволяют делать различные виды анализа и представление данных переходного процесса.

## Симуляция

### Алгоритм симуляции

NL5 — это кусочно-линейный (PWL) симулятор. Все компоненты в NL5 либо линейные, либо кусочно-линейные: состоят из некоторого числа линейных сегментов. Например, диод либо открыт, либо закрыт, следовательно, его PWL представление имеет только два сегмента. Пока все компоненты остаются в их текущих линейных сегментах, схема описывается той же системой линейных дифференциальных уравнений. Система модифицируется только в моменты, когда хотя бы один из компонентов меняет свой линейный сегмент. Когда это происходит, текущий линейный диапазон симуляции завершается и начинается новый. Обычно симуляция начинается с расчета рабочей точки по постоянному току (DC operating point при  $t=0$ ), а затем рассчитываются один или больше интервалов линейной симуляции. Производительность алгоритм может быть оптимизирована несколькими параметрами, расположенными в диалоговых окнах **Transient Settings** и **Advanced Settings**.

**DC operating point.** Симуляция всегда начинается в момент  $t=0$ . В этот момент вычисляется рабочая точка по постоянному току (DC). Расчет выполняется с учетом начальных условий (Initial Condition, IC) для компонентов. Например, конденсатор заменяется источником напряжения, если задано IC voltage (начальное напряжение), или игнорируется (открытая цепь), если IC не задано (пустое). Индуктивность заменяется источником тока, если IC current (начальный ток) задан, или закорачивается, если IC не задано. Диод рассматривается как открытая цепь, если состояние IC «Off», и как закороченная цепь, если состояние IC «On».

Если схема имеет более одного стабильного состояния, она может быть установлена в нужное состояние определением подходящих начальных условий (IC). Другой способ начальной установки схемы — использовать этикетку с моделью **model Label** и задать параметр **VIC** для нее. Если VIC не пусто, временный источник напряжения величиной VIC присоединяется к этикетке через резистор R только на время расчета рабочей точки на постоянном токе. Когда расчет рабочей точки закончен, источник напряжения удаляется.

Результат вычисления рабочей точки — известные напряжения, токи и состояния всех компонентов. Когда рабочая точка найдена, начинается первый линейный интервал.

**Linear range simulation.** В линейном интервале (linear range) схема описывается системой линейных дифференциальных уравнений, которые решаются методом трапецеидального интегрирования (Trapezoidal, метод трапеций). Метод поддерживает достаточную точность с умеренными устойчивостью и скоростью вычисления. В процессе интервала линейной симуляции алгоритм выполняет «обнаружение точки переключения» (switching point detection): проверяет условия для всех компонентов, которые могут изменить свое состояние (диоды, переключатели, логические компоненты), линейный сегмент (PWL models), или изменить амплитуду или наклон (Pulse и Step models). Если обнаруживаются какие-то изменения, текущий линейный интервал заканчивается и начинается новый.

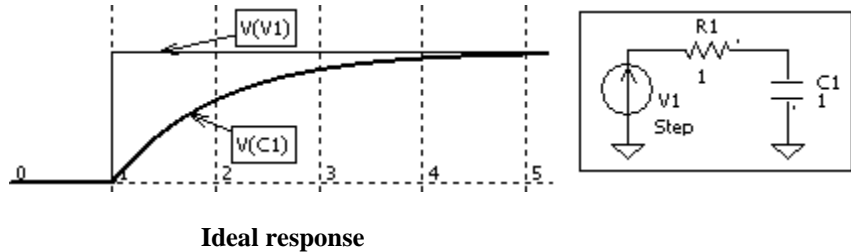
**Calculation step (шаг расчета).** В отличие от большинства аналоговых симуляторов NL5 не выполняет автоматического управления шагом. Выбор шага вычислений (calculation step) лежит на пользователе. Это дает пользователю возможность полного контроля над симуляцией, хотя требует некоторого опыта и понимания процесса. Эмпирическое правило — сохранять шаг



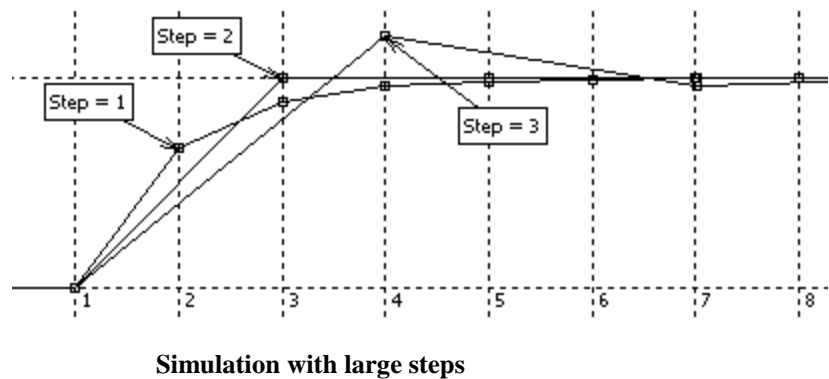
вычислений меньше постоянной времени в схеме, иначе метод интегрирования может стать нестабильным и производить «числовые осцилляции». NL5 обнаруживает такие осцилляции и выводит предупреждающие сообщения. В этом случае будет полезно исследовать проблему и либо уменьшить шаг вычисления, либо игнорировать осцилляции, как незначачие.

Однако иметь шаг расчета «меньше постоянной времени» не является необходимым условием. Иногда, даже достаточно большой шаг обеспечивает хорошую стабильность, при этом скорость симуляции может быть значительно выше. Чтобы найти оптимальный шаг расчета, запустите симуляцию несколько раз с разными шагами и сравните результаты симуляции. Как правило, уменьшение шага ниже некоторого уровня не дает каких-либо заметных изменений. Выбор шага близкого к этому уровню дает наилучшую скорость симуляции.

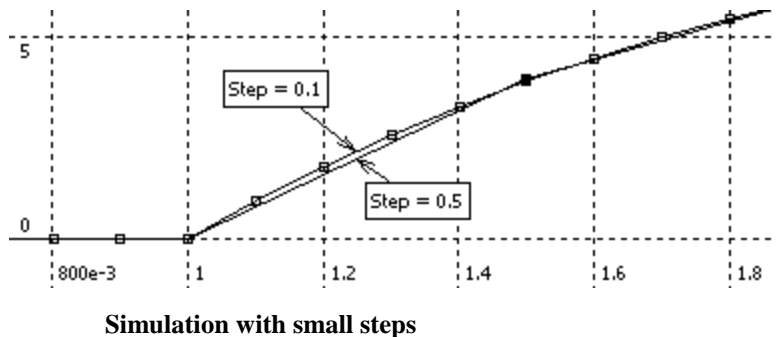
Следующий пример показывает, как шаг расчета сказывается на симуляции простой схемы. Постоянная времени RC цепи 1s. Таким образом, желательно иметь шаг < 1s.



Когда выбираются шаги 1s, 2s и 3s, форма сигнала неправильная. «Выброс» и дальнейшая осцилляция наблюдаются при шаге 3s. Однако если точная форма кривой вас не интересует, и если она не сказывается на функционировании остальной схемы, такой шаг вполне можно использовать. Сообщение «numerical oscillation, числовые осцилляции» можно выключить, установив опцию **Do not detect oscillations**.



Шаг расчета меньше 1s дает точную форму кривой. Например, различие между кривыми с шагом 0.5s и 0.1s может быть замечено только в самом начале переходного процесса, и оно крайне мало.

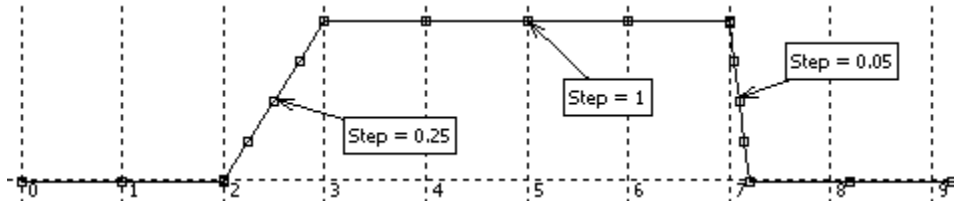


**Automatic step reduction.** Хотя шаг вычисления задается пользователем, NL5 все-таки может автоматически уменьшать шаг, чтобы удовлетворить следующим условиям:

- Период синусоидального источника содержит хотя бы 16 шагов.

- Состояние импульсов «On»/«Off» содержит хотя бы 4 шага.
- Ненулевой передний или задний фронт содержит хотя бы 4 шага.
- Интервал между двумя точками в File model источников V/I содержит хотя бы 4 шага.
- Время задержки линии передачи и «delay» компонента содержат хотя бы 2 шага.

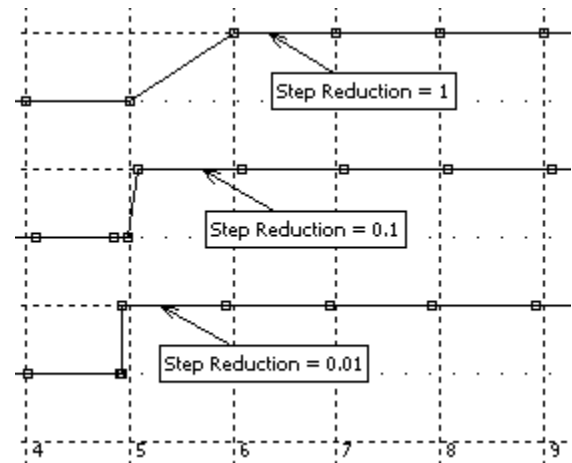
Следующий пример показывает, как расчетный шаг уменьшается во время фронтов импульса:



Автоматическое уменьшение шага может также использоваться для поддержания лучшего разрешения по времени обнаружения точек переключения. Если постоянная времени схемы большая, и большой расчетный шаг используется для линейного интервала времени, уменьшение шага только в точках переключения может значительно улучшить эффективность симуляции. Параметр **Step Reduction** задает, насколько шаг разрешения в процессе обнаружения точки переключения меньше, чем заданный шаг вычисления.

Например,  $\text{Step Reduction} = 0.1$  означает, что точки переключения будут обнаруживаться с разрешением по времени приблизительно в десять раз лучше, чем заданный шаг вычисления. Следующий график показывает форму сигнала, полученную с расчетным шагом = 1s, и step reduction равным 1, 0,1 и 0,01.

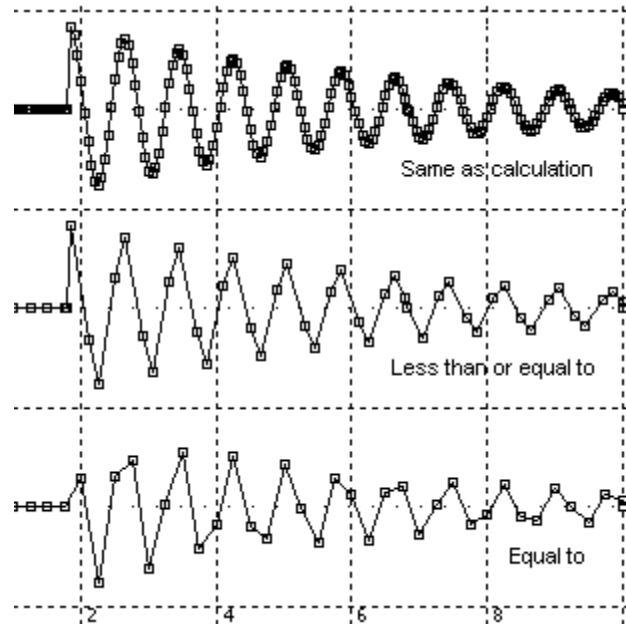
Использование автоматического уменьшения шага не сказывается сильно на скорости вычислений. Количество дополнительных расчетных шагов приблизительно равно  $-\log_2(\text{Step Reduction})$ . Для  $\text{step reduction} = 0.01$  будет сделано только 6 дополнительных шагов.



Однако, если предпочтительнее постоянный шаг расчета, любые его изменения могут быть отменены. Например, Function model некоторых компонентов имеет выходной сигнал, задержанный на один шаг. Если шаг постоянный, это проявляется как известная постоянная задержка, которая может быть учтена должным образом. Если шаг вычисления меняется, задержка будет также «переменной», и с симуляцией такой схемой могут возникнуть проблемы, особенно в системах с обратной связью. Установите **Do not reduce calculation step**, чтобы избавиться от автоматического уменьшения шага.

**Data sampling step.** Для точной симуляции может потребоваться очень маленький шаг вычислений. Однако в сохранении всех данных симуляции в памяти нет нужды, если интересующий сигнал гладкий и меняется относительно медленно. Опция **Data sampling step** позволяет выполнить симуляцию с таким мелким шагом, который нужен, но хранить только часть данных, используя достаточно мало памяти. Доступны следующие опции:

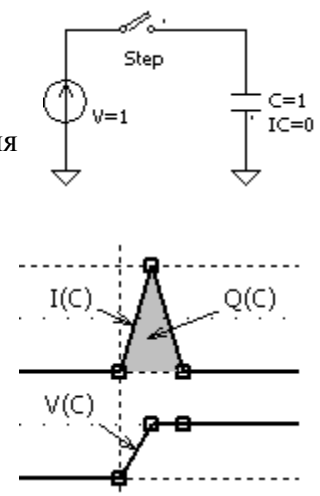
- **Same as calculation (сохранить все данные).** Все расчетные данные хранятся в памяти. Эта опция дает более точное отображение и анализ данных с наибольшим потреблением памяти.
- **Less than or equal to (меньше или равно).** Если выбрана эта опция, должен быть введен и максимальный шаг отображения. Данные хранятся с заданным шагом. Вдобавок хранятся все «критические» точки данных, такие как экстремумы (max и min), крутые фронты, точки переключения и т.д. Это дает достаточную экономию памяти, с все еще надежным отображением данных.
- **Equal to (равно).** Если выбрано, должен быть введен и шага отображения. Данные будут храниться только с заданным шагом, что может дать большую экономию памяти.



Однако некоторые важные детали переходного процесса могут быть потеряны, также как и риск появления биений для быстро меняющихся сигналов.

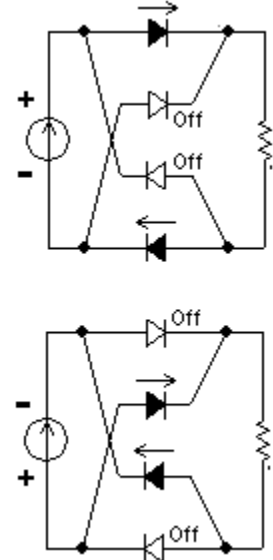
**Бесконечные импульсы напряжения и тока.** В отличие от многих симуляторов, основанных на Spice-алгоритме, NL5 способна симулировать схемы с действительно **идеальными** компонентами. Примером такого компонента является идеальный ключ, который имеет либо нулевое, либо бесконечное сопротивление, и переключается из одного состояния в другое мгновенно. Если такой ключ используется для заряда или разряда конденсаторов, могут возникать бесконечно короткие импульсы тока бесконечно большой амплитуды. Однако, несмотря на бесконечную амплитуду, площадь импульса (интеграл под кривой) будет ограниченным, и равным полному количеству заряда, перетекшего в конденсатор или из конденсатора в момент переключения. Аналогично, при мгновенном прерывании тока через индуктивность возникает бесконечный импульс напряжение, однако площадь этого импульса также ограничена и пропорциональна магнитному потоку.

Такие бесконечно короткие импульсы с бесконечной амплитудой обычно называются импульсами Дирака, или дельта-функцией. Так как изображать дельта-функцию на графике проблематично, в NL5 реализован следующий метод отображения. Дельта-функция напряжения или тока показана как треугольный импульс с шириной каждого склона равной минимальному шагу расчета в этот момент, и площадью, удовлетворяющей закону сохранения заряда и магнитного потока. Если шаг расчета мал, изображаемый импульс будет достаточно коротким, а амплитуда достаточно большой, так что импульс будет визуально восприниматься «почти» как дельта-функция. В то же время интеграл под импульсом даст абсолютно правильное значение заряда или магнитного потока. При изменении шага расчета длительность и, соответственно, амплитуда изображенного импульса изменятся, однако интеграл по-прежнему останется верным.



**Convergence (сходимость).** В симуляторах, основанных на Spice-алгоритме, проблема сходимости может обнаружиться в любое время: и во время анализа на постоянном токе (DC), и во время анализа переходного процесса. Поскольку симулятор NL5 кусочно-линейный, большую часть времени он работает с линейными системами, которые никогда не обнаруживают проблем со сходимостью. Единственно, когда NL5 симуляция может испытывать некоторые трудности, это в момент, когда один и более компонентов меняют свое состояние или линейный сегмент.

Для систем с идеальными кусочно-линейными компонентами типична ситуация, когда несколько компонентов должны одновременно изменить состояние, иначе система не будет сходиться. Например, в стандартном мостовом выпрямителе диоды всегда переключаются парами или даже все диоды одновременно. С идеальными диодами, имеющими нулевое сопротивление при включении и бесконечное, когда закрыты, простой алгоритм может столкнуться с некоторыми трудностями при разрешении процесса переключения. Возможным решением является добавление малого резистора последовательно и/или большого резистора параллельно диодам. Однако это может дать очень маленькие постоянные времени, которые потребуют очень маленького шага вычисления, так что все преимущества от использования идеальных диодов исчезнут.



Поскольку традиционные методы итерации не работают достаточно надежно для таких систем, NL5 использует собственный устойчивый алгоритм. До настоящего времени алгоритм работал превосходно со всеми протестированными схемами, однако никто не может помешать пользователям разработать что-то специфическое, что может иметь трудности со сходимостью.

Другая проблема, общая для любой программы, заключается в том, что использование арифметики с плавающей точкой приводит к потере точности из-за ошибок округления. Эти ошибки могут сказаться как на сходимости в точках переключения, так и на линейных интервалах симуляции.

Если симуляция существенно замедляется в точках переключения или останавливается с сообщением об ошибке «No solution», или если симуляция дает явно неверный результат, следующие опции и параметры могут помочь:

- **Change states one at a time (изменять состояние по одному).** Установка этой опции может улучшить сходимость в точках переключения.
- **Machine precision (машинная точность).** (известное также как “machine epsilon”). Этот параметр задает минимум относительной разницы между двумя числами с плавающей точкой, которые могут быть успешно распознаны. Это значение сказывается не только на сходимости в точках переключения, но на всех результатах симуляции, и может быть изменено в достаточно широких пределах ( $1e-6 \dots 1e-15$ ).

## Данные симуляции

**Traces (кривые).** Во время симуляции NL5 хранит данные в памяти. Данные для сохранения выбираются пользователем как кривые (traces) в диалоговом окне **Transient Data**. Доступно несколько типов кривых: V (напряжение), I (ток), P (мощность), Variable (переменная) и Function (функция).

Когда начинается симуляция, все кривые автоматически очищаются, а затем начинается сохранение новых данных симуляции. Новые данные отображаются на закладке **Run** окна **Transient**. Данные последней симуляции могут быть перемещены в **Storage** со специальной закладкой в окне **Transient**. Сохраненные таким образом данные не очищаются автоматически и могут быть использованы для сравнения результатов разных запусков симуляции.

Если выбрана специальная опция **Store last Run** («сохранить последние данные»), то в момент запуска новой симуляции текущие данные (**Run**) будут перемещены в специальную закладку **Last** («последний»), стерев при этом находящиеся там данные. Таким образом, закладка **Last** всегда содержит предыдущие данные, которые можно использовать для сравнения с данными последней симуляции **Run**.

Кривые могут копироваться в буфер обмена, сохраняться в файлах данных «nlt» или экспортироваться в текстовый файл формата «csv». В свою очередь данные могут вставляться из буфера обмена, загружаться из файла данных «nlt» или импортироваться из текстового файла, как новый график. Такой график всегда отображается в окне **Transient**, независимо от того, какая закладка выбрана. Он не очищается, когда запускается новая симуляция, и может использоваться в качестве опорной кривой для симуляции. Он также может быть переименован произвольным образом.

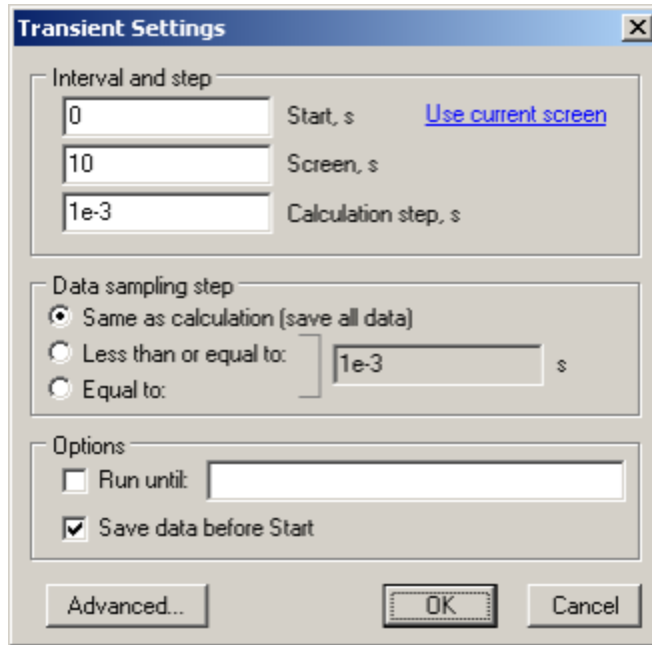
**Memory (память).** Данные симуляции хранятся в оперативной памяти. Память отводится только когда это необходимо, относительно маленькими блоками. Если доступной оперативной памяти не хватает для хранения постоянно нарастающего количества данных, операционная система начинает «сбрасывать» данные на диск, который может существенно замедлить симуляцию и отображение результатов. Чтобы избежать этого, используется следующий механизм: когда количество памяти, требуемой для отображения графика, превышает максимальное значение, заданное на странице **Transient** диалогового окна **Preferences**, блок памяти, в данный момент хранящий самое начало графика, освобождается и используется для новых данных. Таким образом, кривая будет усечена в начальной части, чтобы сохранить последние данные. Когда это происходит впервые для одной или нескольких кривых, появляется предупреждающее сообщение в строке состояния окна **Transient**.

Когда память графика усекается, график не может немедленно обновиться на экране: график будет показывать «не существующие» данные, пока его не перерисуют.

Общее текущее количество памяти, используемое для алгоритма симуляции и всех кривых, всегда отображается в поле **Memory used** окна состояния **Transient**, так что пользователь при необходимости может предпринять что-то разумное.

## Установки переходного процесса (Transient Settings)

Щелкните по кнопке **Transient settings** , или выберите команду **Transient | Settings** Основного Меню. Появится диалоговое окно **Transient Settings**:



**Interval and step (интервал и шаг).** Когда симуляция запущена, временной диапазон окна переходного процесса (**transient**) автоматически устанавливается в заданный интервал.

- **Start, s.** Левый край окна переходного процесса (начало отображения данных).
- **Screen, s.** Размер окна переходного процесса.
- **Calculation step, s. (шаг расчета).** Максимальный шаг расчета. Реальный шаг может быть уменьшен алгоритмом, если нужно.
- **Save data before Start (сохранить данные до момента Start).** Если установлено, все данные симуляции до момента начала отображения **Start** сохраняются в памяти и доступны для отображения. Иначе эти данные не сохраняются, позволяя экономить память.
- **Use current screen (применить текущий экран).** Щелкните, чтобы использовать текущие установки экрана переходного процесса в качестве нового интервала симуляции. Параметры **Start** и **Screen** будут установлены согласно тем, что отображаются на текущем графике переходного процесса.

**Data sampling step (шаг выборки данных).** Задает шаг выборки данных (сохранения) равный или отличный от шага расчета. Эта опция не сказывается на результатах расчетах, а только уменьшает количество сохраняемых данных.

- **Same as calculation (сохранить все данные).** Все расчетные данные хранятся в памяти. Эта опция дает отображение более точных данных и анализ с наибольшим потреблением памяти.
- **Less than or equal to (меньше или равно).** Если опция выбрана, должен быть введен и максимальный шаг отображения. Данные хранятся в заданном шаге. Вдобавок он хранит все

«критические» точки данных, такие как экстремумы (max и min), крутые фронты, точки переключения и т.д. Это дает достаточную экономию памяти, с все еще надежным отображением данных.

- **Equal to (равно).** Если выбрано, должен быть введен и шага отображения. Данные будут храниться только с заданным шагом, что может дать наибольшую экономию памяти. Однако некоторые важные детали переходного процесса могут быть потеряны, как и риск получения биений для быстро меняющихся сигналов.

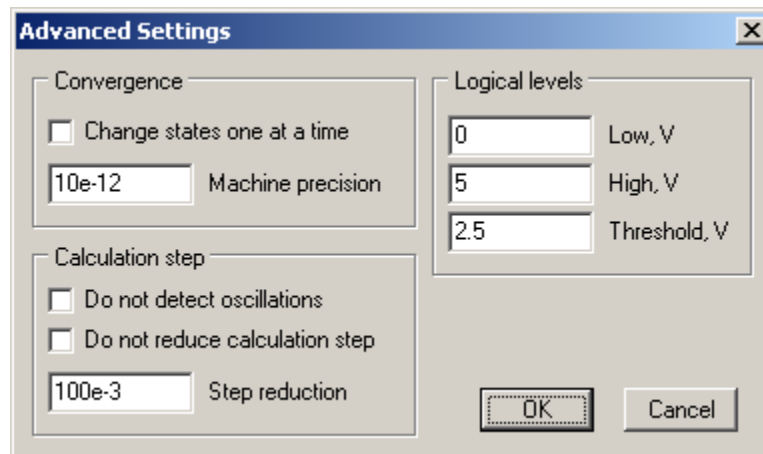
### Options (опции).

- **Run until (запустить до...).** Если установлено, и введено выражение, данное выражение будет вычисляться на каждом шаге расчета. Симуляция будет немедленно приостановлена, как только значение выражение будет положительным. Затем симуляцию можно продолжить. В выражении могут быть использованы переменные **t** – текущее время симуляции, напряжение, ток и мощность на компоненте в виде: **V(name)**, **I(name)**, and **P(name)**, где **name** – это имя компонента (V, I или P кривая должны быть разрешены для компонента). Выражение не будет вычисляться пока время **t < Start**. Примеры выражения **Run until**:

```
V(C1) > 5.0
(I(R2) > 1m) && (t > 10)
(P(Rload) > 3.3) || (t > 100)
```

- **Save data before Start (сохранить данные до момента Start).** Если установлено, все данные симуляции до момента начала отображения **Start** сохраняются в памяти и доступны для отображения. Иначе эти данные не сохраняются, позволяя экономить память.

**Advanced («продвинутые»).** Щелкните, чтобы открыть диалоговое окно **Advanced Settings**:



**Convergence (сходимость).** Параметры, которые могут сказаться на сходимости вычислений рабочей точки по постоянному току и расчете точек переключения.

- **Change states one at a time.** Задаёт режим переключения состояний компонентов при итерациях.
- **Machine precision («машинный ипсилон»).** Минимальная относительная разность между двумя числами с плавающей точкой, которая может быть надежно распознана.

**Calculation step (шаг расчета).** Несколько опций, относящихся к шагу вычислений.


- **Do not detect oscillations.** Не отображать предупреждающее сообщение, если обнаруживаются «числовые» осцилляции.
- **Do not reduce calculation step.** Всегда использовать только заданный шаг вычисления (не уменьшать шаг).
- **Step reduction.** Задаёт, насколько разрешение шага в процессе обнаружения точек переключения лучше, чем шаг расчета.

**Logical levels.** Эти установки применимы к логическим компонентам и некоторым моделям с логическим типом входа.

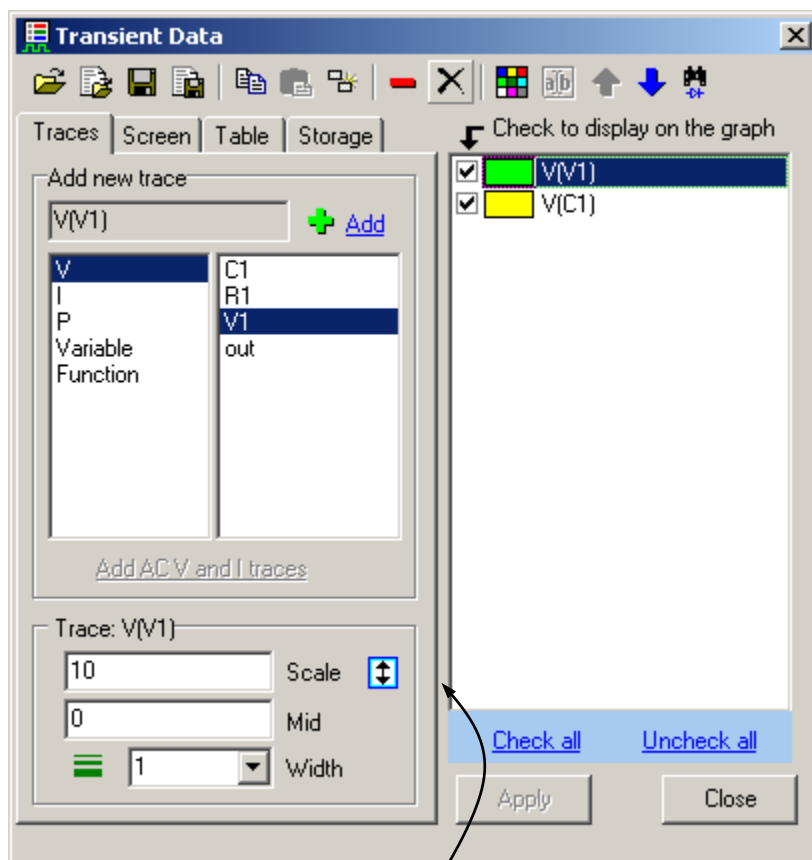
- **Low, V.** Низкий логический уровень. Должен быть  $< \mathbf{High}$ .
- **High, V.** Высокий логический уровень. Должен быть  $> \mathbf{Low}$ .
- **Threshold, V.** Логический порог: напряжение ниже порога считается **Low**, выше — **High**. Пороговое напряжение должно быть между **Low** и **High**.



## Данные переходного процесса (Transient Data)

Щелкните по кнопке **Transient data** , или выберите команду **Transient | Data**. Появится окно **Transient Data**. В окне всегда показаны данные, относящиеся к активному документу (схеме): переключение на другой документ автоматически обновит данные в окне. Окно содержит инструментальную панель, список кривых (traces) и 4 страницы, используемые для следующих операций:

- **Traces (кривые):** добавляет кривые, задает индивидуальные масштаб и ширину для кривых.
- **Screen (экран):** задает масштаб, линии сетки и другие опции экрана для графика.
- **Table (таблица):** конфигурирует таблицу данных.
- **Storage («хранилище»):** обслуживает хранение данных.



Поместите мышку поверх «разделителя» , затем нажмите левую кнопку мышки и перетащите разделитель, чтобы изменить размер панелей.

**Trace list (список кривых)** показывает все текущие доступные кривые. Флажок показывает следующие свойства кривых, в зависимости от того, какая страница выбрана:

- Страницы **Trace** и **Screen** – кривая будет показана на графике.
- Страница **Table** – кривая будет показана в таблице.
- Страница **Storage** – кривая будет сохраняться в **storage**.

Щелкните по **Check all**, чтобы поставить флажок на всех кривых, или **Uncheck all**, чтобы убрать флажок

Большая часть команд панели применима только к выделенным кривым. Одна или больше кривых могут быть выбраны из списка с помощью мышки и клавиш **Ctrl** и **Shift**. Пожалуйста, заметьте, что выделенные кривые подсвечены в списке, и «выделенное» состояние кривой не зависит от состояния её флажка. На рисунке выше обе кривые имеют выставленный флажок, но только V(V1) выбрана.

Дважды щелкните по кривой, чтобы изменить ее цвет.

В этой главе описаны команды инструментальной панели, и работа со страницей **Traces**. Другие страницы описаны в главе **Transient Window** (разделы **Graph**, **Data table** и **Storage**).

## Инструментальная панель

Команды кнопок инструментальной панели относятся ко всем или только выделенным кривым. Некоторые из этих команд также доступны через контекстные меню в окне **Transient**.

- **Open file**. Загрузить кривые из файла данных типа “nlt”.
- **Import traces**. Импортировать кривые из текстового файла формата “txt” или “csv”. Формат данных файла должен быть аналогичен формату экспорта кривых. Первая колонка содержит время (в секундах), другие колонки содержат данные кривой. Первая строка — это строка заголовка: она может содержать любой текст в первой колонке, и имена кривых в других колонках. Если имя кривой состоит из символа, отличного от цифры и буквы, оно должно заключаться в кавычки. Данные и имена могут разделяться запятыми, пробелами или табуляцией. Например:

```
t(s),input,"trace 2"  
0,0,10.0  
0.1,1.1,9.0  
0.2,5.1,8.0  
0.3,7.5,7.0
```

Когда файл загружен, его содержимое отображается в диалоговом окне **Import Traces** для проверки:

| t(s) | input | trace 2 |
|------|-------|---------|
| 0    | 0     | 10.0    |
| 0.1  | 1.1   | 9.0     |
| 0.2  | 5.1   | 8.0     |
| 0.3  | 7     | 5       |

Щелкните по **OK**, чтобы принять импортированный файл. Новые кривые будут созданы и показаны на графике.








- **Save selected traces** – сохранить выделенные кривые в файл данных типа “nlt”. Только выделенные (подсвеченные) кривые из списка будут сохранены.
- **View/Export selected traces** – посмотреть и экспортировать выделенные кривые в текстовом виде (“txt” или “csv” формат). Только выделенные (подсвеченные) кривые из списка будут показаны. Появится диалоговое окно **View/Export**:

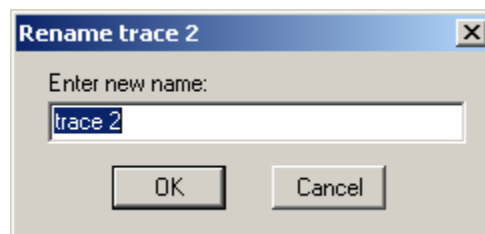
| time[s] | input  | trace 2 |
|---------|--------|---------|
| 0       | 0      | 10      |
| 1e-3    | 11e-3  | 9.99    |
| 2e-3    | 22e-3  | 9.98    |
| 3e-3    | 33e-3  | 9.97    |
| 4e-3    | 44e-3  | 9.96    |
| 5e-3    | 55e-3  | 9.95    |
| 6e-3    | 66e-3  | 9.94    |
| 7e-3    | 77e-3  | 9.93    |
| 8e-3    | 88e-3  | 9.92    |
| 9e-3    | 99e-3  | 9.91    |
| 10e-3   | 110e-3 | 9.9     |
| 11e-3   | 121e-3 | 9.89    |
| 12e-3   | 132e-3 | 9.88    |
| 13e-3   | 143e-3 | 9.87    |
| 14e-3   | 154e-3 | 9.86    |
| 15e-3   | 165e-3 | 9.85    |
| 16e-3   | 176e-3 | 9.84    |
| 17e-3   | 187e-3 | 9.83    |

Выделенные кривые показаны как текст в таблице. Первоначально кривые показываются во временном интервале между курсорами экрана или, если курсоры выключены, на видимом экране. Измените значения **From** и **To** и нажмите **Enter** или щелкните по кнопке **Update table**, чтобы изменить интервал. Кривые отображаются с фиксированным




временным шагом, заданным значением **Step**. Начальный шаг автоматически устанавливается таким, чтобы число точек было близко к значению, заданному параметром **Approximate number of points** страницы **Transient** диалогового окна **Preferences** (но не превышающему значения **Max number of points**, заданного там же). Число значащих цифр для колонок времени и данных может быть задано.

Щелкните по **Export**, чтобы сохранить таблицу в текстовом файле, как значения, разделенные запятыми.

-  • **Copy selected traces** – скопировать выделенные кривые в буфер обмена. Только выделенные (подсвеченные) кривые из списка будут скопированы.
-  • **Paste traces** – вставить кривые из буфера обмена.
-  • **Duplicate selected traces** – сдублировать выделенные кривые. Эта операция эквивалентна операциям **Copy/Paste**. . Только выделенные (подсвеченные) кривые из списка будут сдублированы.
-  • **Remove selected traces** – удалить выделенные кривые. Только выделенные (подсвеченные) кривые из списка будут удалены.
-  • **Delete all traces** – удалить все кривые из списка.
-  • **Select color** – задать цвет выделенной кривой. Должна быть выделена только одна кривая. Двойной щелчок по кривой выполняет ту же операцию.
-  • **Rename trace** – переименовать выделенную кривую. Должна быть выделена только одна кривая. Переименованы могут быть только кривые, загруженные или импортированные из файла, сдублированные или вставленные из буфера обмена. Переименование кривой типа **Function** изменяет саму функцию. Появится окно **Rename**:

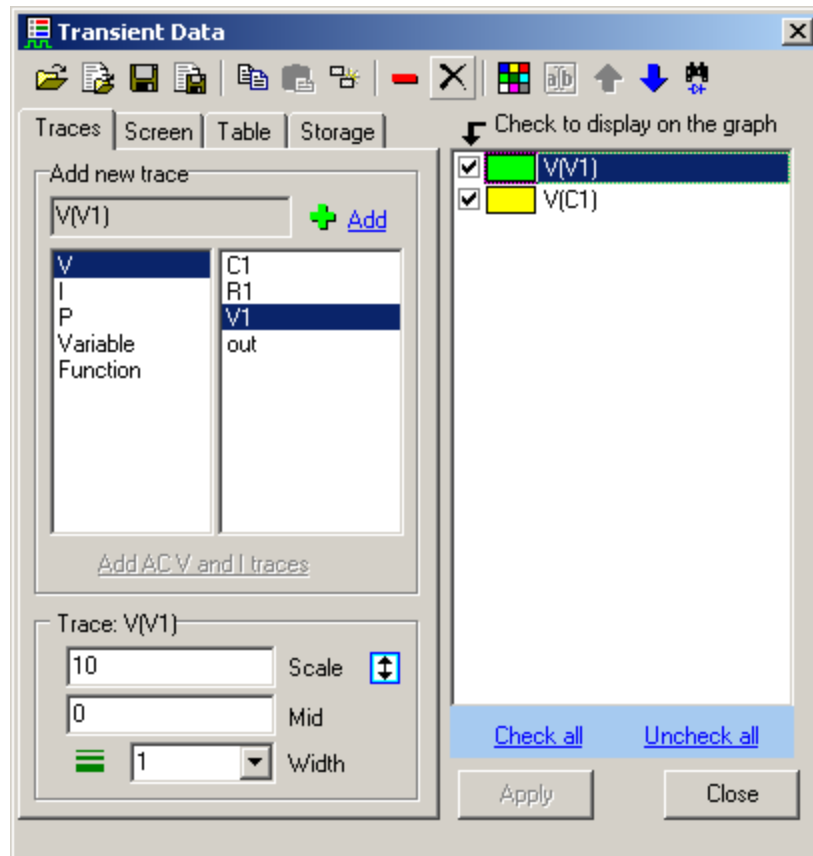


Введите новое имя кривой и щелкните по **OK**.

-  • **Move selected traces up** – передвинуть выделенные кривые вверх в списке. Эта операция меняет порядок кривых в списке, на графике и в таблице.
-  • **Move selected traces down** - передвинуть выделенные кривые вниз в списке. Эта операция меняет порядок кривых в списке, на графике и в таблице.
-  • **Find component**. Если выделенная кривая – это V, I или P на компоненте, щелкните, чтобы показать компонент на схеме. Компонент будет выделен (подсвечен) и отцентрован на экране.


## Traces (кривые)

Страница **Traces** окна **Transient Data** используется для добавления и удаления кривых, и для установок для отдельных кривых масштаба и ширины.




**Add new trace** (добавить новую кривую). Выберите тип графика в списке слева:


- **V** – напряжение.
- **I** – ток.
- **P** – мощность.
- **Variable** – схемная переменная, определенная в окне **Variables**.
- **Function** – произвольная функция.

Если выбрана кривая **V**, **I** или **P**, список справа покажет компоненты, доступные для графика этого типа: модель компонента должна поддерживать выбранный тип. Выделите компонент и щелкните по кнопке **Add** , или дважды щелкните по имени переменной, чтобы добавить новый график в список. Имя кривой состоит из буквы с последующим именем компонента в скобках:

V(R1), I(C2), P(L3)

Если выбрана кривая **Variable**, список справа покажет все переменные, доступные в схеме. Выберите переменную и щелкните по кнопке **Add** , или дважды щелкните по имени

переменной, чтобы добавить новый график в список. Имя кривой будет то же, что и у переменной.


Если выбрана кривая **Function**, введите функцию в окне редактирования и щелкните по кнопке **Add** , чтобы добавить новую кривую в список. Функция может состоять из арифметических операторов и функций, параметров компонентов, текущего времени переходного процесса  $t$  и **V**, **I** и **P** на компоненте:

```
sin(t*1000)*(1+cos(t*10))
V(in)/I(A1)
V(X1.V1)
sq(V(r1))/r1
```

Имя кривой — это сама функция, так что переименование кривой изменит функцию.

Графики **V**, **I** и **P** могут также добавляться из контекстного меню схемы и кнопками панели окна **Components**.

Для каждой кривой могут быть установлены следующие индивидуальные параметры:

- **Scale.** Масштаб: значение кривой на половину экрана.
- **Mid.** Значение кривой в середине экрана.
- **Width.** Ширина линии кривой в пикселях.
-  • **Fit the Screen.** Щелкните, чтобы автоматически установить **Scale** и **Mid** так, чтобы кривая была отображена на весь экран.


Выберите одну или несколько кривых в списке **Trace**, измените параметры и нажмите **Enter** или щелкните по кнопке **Apply**. Если выделенные кривые имеют разные значения для одного параметра, соответствующее поле будет оставаться чистым. Оставьте поле чистым, чтобы сохранить индивидуальные значения неизменными, или введите новое значение, чтобы применить его ко всем выбранным кривым.

## Симуляция

Используйте команды Основного Меню, кнопки инструментальной панели или горячие клавиши для выполнения симуляции.

➔ • **Start transient (Transient | Start, or F6)**. Запустить анализ. Когда запускается анализ переходного процесса, открывается окно графиков **Transient** и временной диапазон экрана устанавливается в значения, заданные в диалоговом окне **Transient Settings: Start** — это левый край экрана, **Screen** — размер экрана. Хотя симуляция всегда начинается в момент  $t=0$ , результаты будут отображаться на графике только с момента **Start**. В зависимости от установки флажка **Save data before Start** данные симуляции до момента начала отображения могут игнорироваться или сохраняться в памяти, чтобы оставаться доступными для отображения позже.

Результаты переходного процесса немедленно отображаются в окне графиков **Transient**. Текущее время симуляции показано в поле **Simulation progress** с зеленым фоном, если переходной процесс в настоящий момент выполняется, или с желтым фоном, если он приостановлен. Количество памяти, используемой для симуляции и графиков, показано в поле **Memory used**.

➔ | • **Pause transient (Transient | Pause, or Space)**. Приостановить анализ (Пауза). Анализ переходного процесса может быть приостановлен, а затем продолжен в любое время. Когда график переходного процесса достигает конца экрана, процесс приостанавливается автоматически (если только не включен режим непрерывного анализа кнопкой **Continuous transient mode**  ).

➔ | • **Continue transient (Transient | Continue, or F7, or Space)**. Продолжить анализ.

Почти все операции в NL5 могут выполняться, когда запущен анализ переходного процесса. Вы можете менять масштаб графика, цвета, добавлять или удалять кривые, менять шаг симуляции, параметры компонентов, и даже редактировать схему. Изменения будут незамедлительно применяться и сказываться на симуляции переходного процесса. Если требуется для операции, анализ переходного процесса будет автоматически приостанавливаться, а затем сразу по окончании операции возобновляться. Для некоторых критических операций, однако, анализ может не возобновляться или даже останавливаться полностью.

✖ • **Stop transient (Transient | Stop)**. Остановить анализ. Когда анализ переходного процесса остановлен, он не может быть продолжен и должен запускаться заново с самого начала.

📄 • **Transient Log (Transient | Log)**. Информация журнала регистрации, показываемая в диалоговом окне, может использоваться для поиска проблем. Последняя запись сохраняется в файле схемы. При отправке файла схемы в службу поддержки (Customer Service) для помощи, пожалуйста, сохраняйте схему после симуляции, чтобы последняя запись журнала регистрации была включена в файл. Щелкните по кнопке **Copy to clipboard**, чтобы скопировать эту текстовую запись в буфер обмена.



- **Save IC (Transient | Save IC)** Сохраняет текущее состояние всех компонентов в их **IC** (начальные условия), если параметр **IC** существует для компонента. Эта функция может использоваться для сохранения состояния компонентов, когда точка периодического установившегося состояния найдена, так что следующая симуляция может сразу стартовать из установившегося состояния без повторного выполнения долгого процесса симуляции в поисках этого состояния. Заметьте, что команда **Save IC** не сохраняет фазы периодических источников, так что для точности результатов момент, когда команда выполняется (сохраняются начальные условия), должен быть выбран правильно.

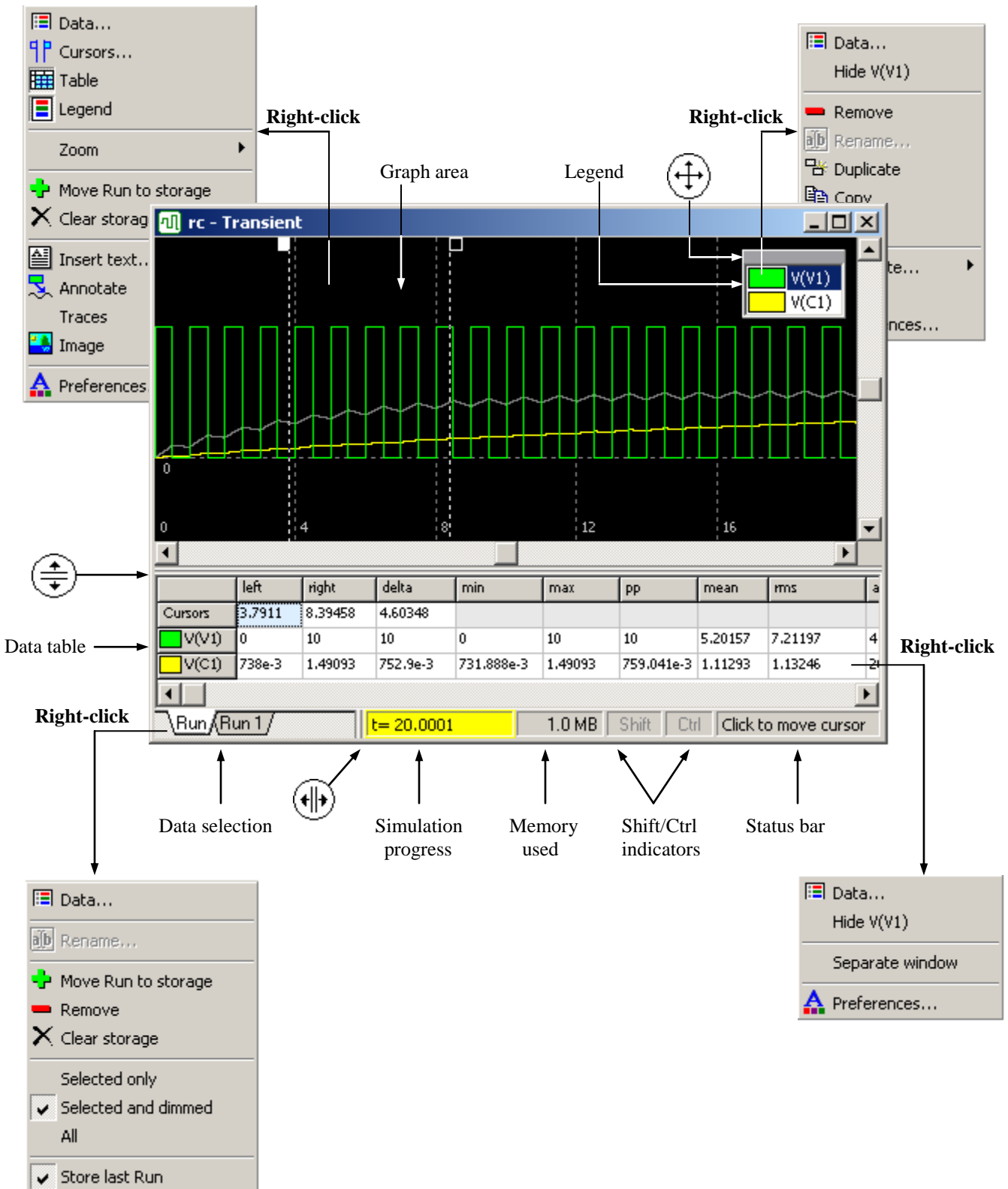





- **Sweep (Transient | Sweep)**. Позволяет запускать серии анализов переходного процесса, когда параметр компонента или переменная меняется в заданном диапазоне, и сохранять данные анализа. Режим **Sweep** конфигурируется на странице **Sweep** окна **Tools**.



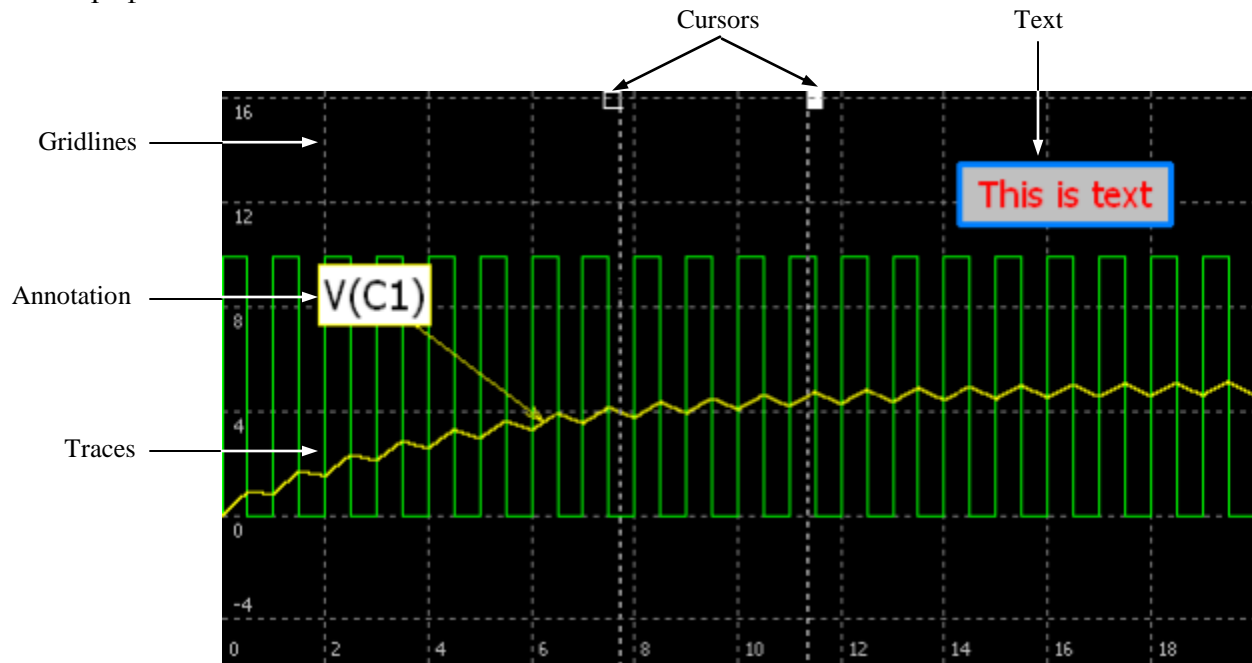
## Окно переходного процесса (Transient window)

Типичный вид окна переходного процесса и его основных компонент показан ниже:



- **Graph** area (область графика), содержит кривые с аннотациями, курсорами и текстом.
- **Legend** window (условные обозначения), содержит список кривых, показанных на графике. Щелкните по серому заголовку панели окна условных обозначений  , чтобы перетащить окно.
- **Data table**, таблица данных, содержит информацию о курсоре/экране и расчетные данные кривой.
- **Data selection**, выбор данных, содержит закладки последних симуляций и сохраненных данных (**storage**). Щелкните по закладке, чтобы выбрать **Run** (результаты симуляции) или сохраненные данные.
- **Simulation progress**, прогресс симуляции, показывает время текущей симуляции и состояние (идет/приостановлена).
- **Memory used**, использование памяти, показывает количество памяти, использованной для симуляции и данных кривых.
- Индикатор **Shift/Ctrl** подсвечен, когда клавиши Shift и/или Ctrl нажаты.
- **Status bar** показывает подсказку, относящуюся к текущей позиции указателя мышки и состоянию **Shift/Ctrl**.
- Поместите указатель мышки поверх «разделителя»  , затем нажмите левую клавишу мышки и перетащите разделитель, чтобы изменить размер области Data selection (выбор данных).
- Поместите указатель мышки поверх «разделителя»  , затем нажмите левую клавишу мышки и перетащите, чтобы изменить размер таблицы данных.
- Щелкните правой клавишей мышки по графику, условным обозначениям, таблице данных или области выбора данных, чтобы увидеть контекстное меню с соответствующими командами.
- Общие свойства окна переходного процесса, такие как цвета, шрифты и некоторые другие опции, могут быть настроены на страницах **Graphs**, **Table**, **Annotation** и **Text** диалогового окна **Preferences**. Свойства, присущие документу (схеме), могут также устанавливаться в окне **Transient Data**.




Область графика и ее компоненты показаны ниже:



## График

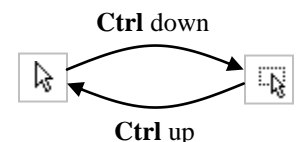
Навигация по графику может выполняться командами, доступными в контекстных меню окна переходного процесса, кнопками инструментальной панели переходного процесса, горячими клавишами, клавишами клавиатуры и мышкой. Очень часто некоторые операции могут выполняться разными способами. Так, например, масштабирование графика, *zoom in/out*, может быть сделано либо только с использованием клавиш клавиатуры, либо только мышкой, либо клавиатурой и мышкой вместе. Пользователя может сам выбрать наиболее эффективный и удобный для него способ.

Есть три режима операций с графиком:

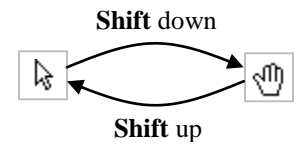
-  • **Cursors**. Перемещение курсоров.
-  • **Zoom**. Масштабирование.
-  • **Scrolling**. Прокрутка.

Режим может выбираться щелчком кнопки на панели переходного процесса. Также есть быстрые способы переключения из режима *Cursors* в *Zoom* и *Scrolling*:

- Нажмите и удержите клавишу **Ctrl**, щелкните и перетащите мышкой, чтобы масштабировать график. Отпустите клавишу **Ctrl**, чтобы вернуться в режим *Cursors*:



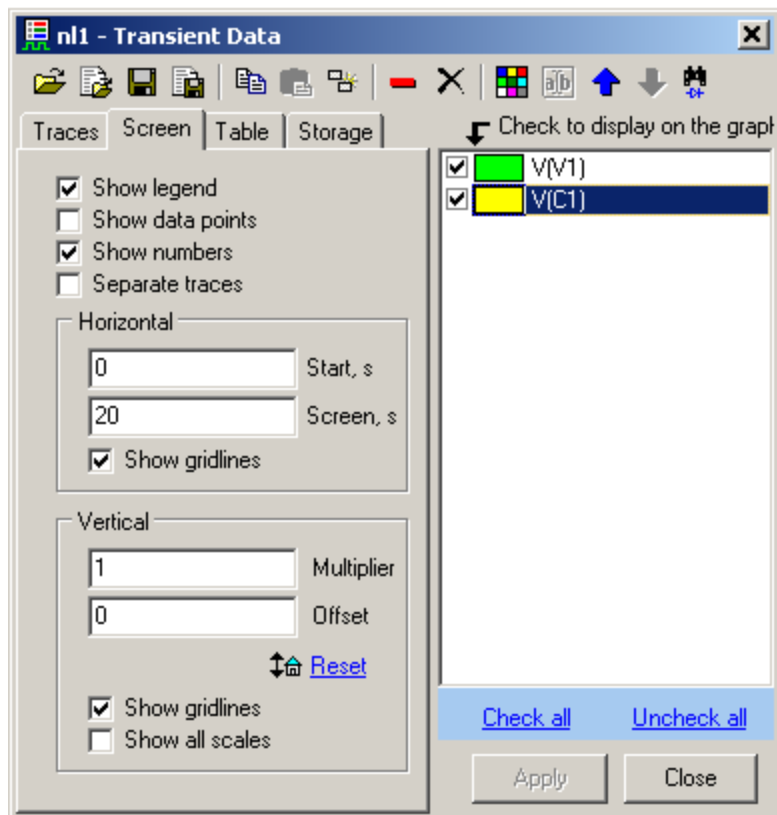
- Нажмите и удержите клавишу **Shift**, щелкните и перетащите мышкой чтобы прокрутить график. Отпустите клавишу **Shift**, чтобы вернуться в режим *Cursors*:





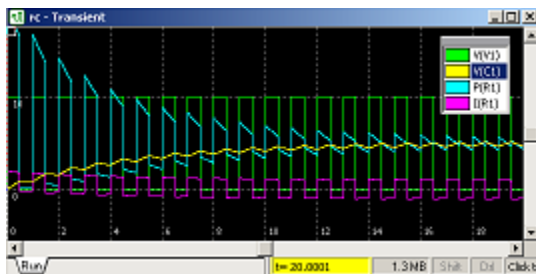
Кривые появляются на графике со своим индивидуальным масштабом, шириной и цветом, определенными на странице **Traces** окна **Transient Data**. Когда выполняется масштабирование графика, изменения масштаба отдельных кривых не происходит. Вместо этого меняются параметры экрана **Multiplier** и **Offset**, которые применяются ко всем кривым. Масштаб выделенной кривой показан на графике. Если выделенная кривая меняется, разметка осей и линии сетки могут тоже измениться.

Плотность линий сетки выбирается автоматически так, чтобы последняя значащая цифра шага была 1, 2 или 5 и дистанция между линиями сетки приблизительно равна значению, заданному на странице **Graphs** диалогового окна **Preferences**, как **Gridlines interval** в пикселях.

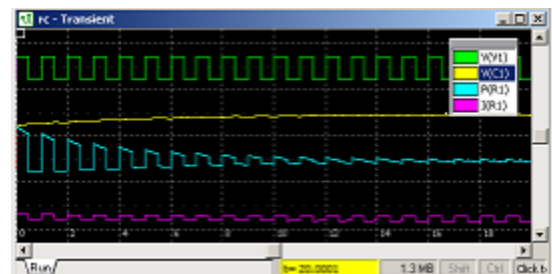
Масштабы, линии сетки и некоторые другие опции графика можно изменить на странице **Screen** окна **Transient Data**:



- **Show legend.** Выберите, чтобы показать окно условных обозначений. Также используйте кнопку **Legend**  на инструментальной панели или в контекстном меню.
- **Show data points.** Выберите, чтобы маркировать точки рассчитанных данных по всей кривой в виде маленьких квадратиков. Эта опция значительно замедляет вывод графиков, но может быть полезна при отладке и выборе шага вычислений.
- **Show numbers.** Выберите, чтобы выводить числа на шкалах.
- **Separate traces.** Также используйте кнопку разделения кривых **Separate traces**  или нажмите **Tab** в окне переходного процесса. Кривые будут разделены вертикально, что помогает различить похожие кривые. Горизонтальные линии сетки используются для деления кривых, а вертикальные шкалы не отображаются. Если график масштабируется мышкой, будет работать только горизонтальное.



Normal mode




Separated traces

**Horizontal.** Устанавливает горизонтальную шкалу и линии сетки.


- **Start.** Время у левого края экрана.
- **Screen.** Размер экрана.
- **Show gridlines.** Выберите, чтобы показать линии сетки.

**Vertical.** Устанавливает вертикальную шкалу и линии сетки.

- **Multiplier.** Множитель шкалы экрана, применяется ко всем кривым.
- **Offset.** Смещение экрана, применяется ко всем кривым.
-  **Reset.** Сбрасывает вертикальный множитель в 1, а смещение в 0.
- **Show gridlines.** Выберите, чтобы показать линии сетки.
- **Show all scales.** Показать шкалы для всех кривых в цвете соответствующей кривой.

## Legend (условные обозначения)


Окно **Legend** содержит список кривых, показанных на графике.

- Чтобы показать/скрыть условные обозначения, щелкните на инструментальном меню по кнопке **Legend** , или используйте команду контекстного меню, или используйте флажок **Show legend** на странице **Screen** окна **Transient Data**.



- Щелкните по строчке кривой, чтобы выбрать ее. Выделенная кривая будет показана поверх всех кривых.
- Дважды щелкните по строчке кривой, чтобы выделить ее и открыть окно **Transient Data**.
- Щелкните правой клавишей мышки по строчке кривой, чтобы выделить ее и открыть контекстное меню. Меню будет содержать некоторые общие команды и команды, относящиеся к выбранной кривой.
- Щелкните на сером заголовке окна и перетащите окно.
- Размер шрифта условных обозначений и ширина окна могут быть выбраны на странице **Legend** диалогового окна **Preferences**.


## Курсоры

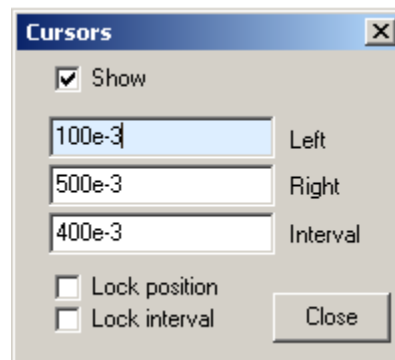
Курсоры используются в основном для выделения временного интервала на графике для расчета таблицы данных (**Data table**). Выделенный (активный) курсор показан с закрашенным квадратиком сверху. Чтобы показать/скрыть (show/hide) курсоры, щелкните по кнопке инструментальной панели **Show/hide cursors** .



Выберите режим **Cursors** (  ), чтобы перемещать курсор на графике.

- Дважды щелкните по графику, чтобы установить оба курсора в одну точку. Этим также активизируются курсоры, если они были выключены.
- Щелкните по графику, чтобы переместить ближайший курсор в эту точку.
- Щелкните и переместитесь, чтобы выделить и передвинуть курсор.

Чтобы поместить курсор в заданное место, и для других опций, щелкните правой клавишей мышки по графику и выберите команду **Cursors**  из контекстного меню. Появится диалоговое окно **Cursors**:




- **Show**. Установите флажок, чтобы показывать курсоры.
- **Left, Right, Interval**. Введите новое положение курсора или расстояние между курсорами, нажмите **Enter**, чтобы применить, или **Esc**, чтобы отменить. Если изменен интервал, изменится положение активного курсора.

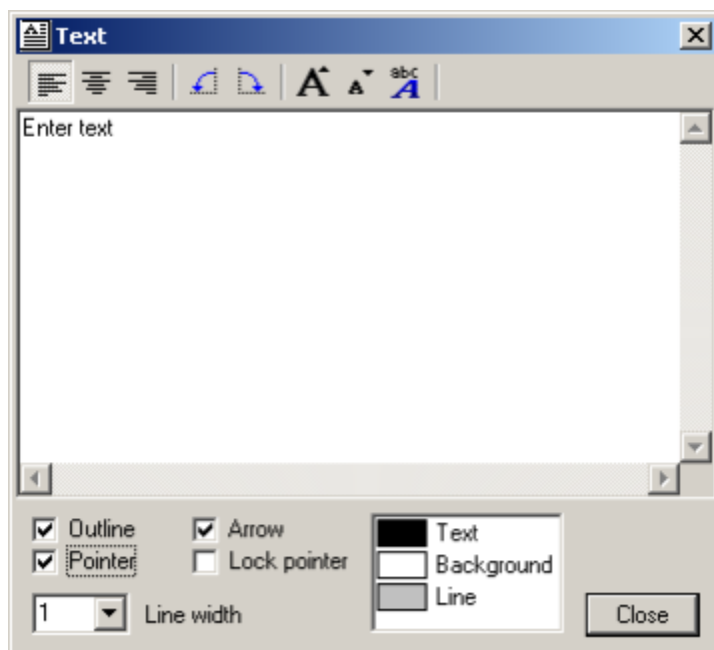
- **Lock position.** Закрепляет курсоры в текущей позиции, так что курсоры не могут перемещаться.
- **Lock interval.** Сохраняет текущий интервал между курсорами. Если один курсор будет передвинут, второй автоматически последует за ним, сохраняя заданный интервал.

Следующие кнопки инструментальной панели можно использовать для перемещений курсоров:

- ✂ • **Right maximum.** Переместить активный курсор к ближайшему справа максимуму выделенной кривой.
- ✂ • **Left maximum.** Переместить активный курсор к ближайшему слева максимуму выделенной кривой.
- ↴ • **Right minimum.** Переместить активный курсор к ближайшему справа минимуму выделенной кривой.
- ↵ • **Left minimum.** Переместить активный курсор к ближайшему слева минимуму выделенной кривой.
- ✂ • **Maximums.** Переместить один курсор к ближайшему справа максимуму, а другой – к ближайшему слева максимуму выделенной кривой.
- ✂ • **Minimums.** Переместить один курсор к ближайшему справа минимуму, а другой – к ближайшему слева минимуму выделенной кривой.

## Текст

Чтобы добавить текст на окно графиков, щелкните правой клавишей мышки по графику и выберите команду **Insert Text**  из контекстного меню. Появится диалоговое окно **Text**:






Введите текст в окно. Одновременно текст появится на окне графиков:





Текст можно форматировать, используя кнопки инструментальной панели:

**Alignment.** Устанавливает выравнивание для многострочного текста.

-  • **Align left.** Выравнивание по левому краю.
-  • **Center.** Выравнивание по центру.
-  • **Align right.** Выравнивание по правому краю.

**Orientation.** Изменить ориентацию текста.

-  • **Rotate left.** Повернуть влево.
-  • **Rotate right.** Повернуть вправо.

**Font.** Изменить размер шрифта или выбрать шрифт.

-  • **Larger font.** Увеличить размер шрифта.

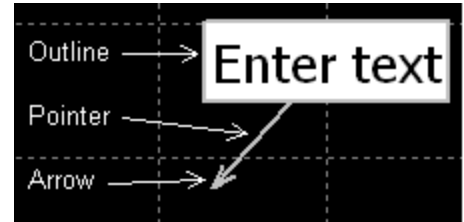





- **Smaller font.** Уменьшить размер шрифта.
- **Select font.** Выбрать шрифт.


**Outline and pointer options.** Опции указателя и рамки.

- **Outline.** Рисовать прямоугольную рамку.
- **Pointer.** Рисовать линию указателя из текста в заданную точку.
- **Arrow.** Рисовать линию указателя со стрелкой.
- **Lock pointer.** Закрепить конец указателя: конец указателя не будет перемещаться, даже если текст перемещается.
- **Line width.** Задаёт ширину линии для рамки и указателя.
- **Color.** Двойной щелчок по пункту в списке для изменения цвета.





Если график масштабировался или прокручивался, текст остается на том же месте, «заякоренный» в левом верхнем углу окна графика. Чтобы переместить текст, щелкните по тексту и перетащите его. Если указатель закреплен, переместится только текст. Чтобы переместить только указатель, щелкните по концу указателя и перетащите его.

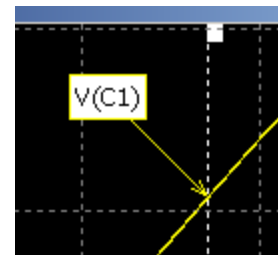
Чтобы отредактировать текст, дважды щелкните по тексту или щелкните правой клавишей мышки по тексту и выберите команду **Edit text**  из контекстного меню. Появится такое же диалоговое окно **Text**.


Чтобы удалить текст, щелкните правой клавишей мышки по тексту и выберите команду **Delete text**  из контекстного меню.

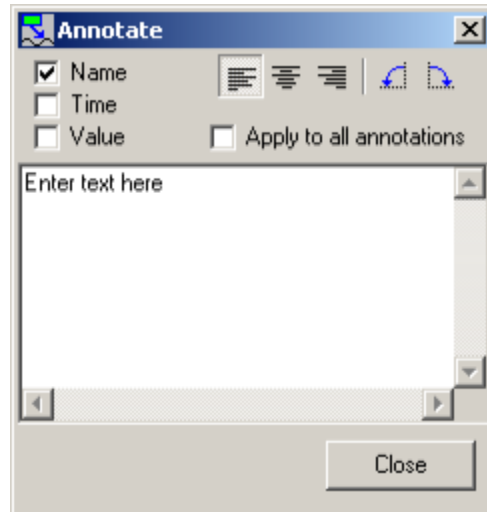
## Аннотация

Аннотации — это текст с указателем, который всегда указывает на ту же точку данных кривой, даже когда график масштабируется или прокручивается. Аннотации принадлежат кривой, так что если кривая удаляется, все аннотации к ней тоже удаляются. Аннотации также удаляются, если данные кривой очищаются. Например, если аннотация добавлена к графику симуляции (**Run**), и запускается новая симуляция, аннотация исчезнет, поскольку данные графика очищаются при новом запуске симуляции.

Чтобы добавить аннотацию, установите активный курсор на точку времени, где нужна аннотация, щелкните правой клавишей мышки по графику, выберите **Annotate**, затем выберите команду **Selected trace**  или **All traces** . Эти же кнопки доступны на инструментальной панели переходного процесса. Аннотация будет добавлена только в том случае, когда данные кривой существуют в момент, указываемый курсором. Если курсоры выключены, аннотация будет добавлена приблизительно в месте 1/3 экрана.






Шрифт, цвета, количество значащих цифр и некоторые другие свойства аннотации могут быть заданы на странице **Annotation** диалога **Properties**. Чтобы изменить текст аннотации и какие-то свойства аннотации, дважды щелкните или щелкните правой клавишей мышки по ней, выберите команду **Edit annotation**  из контекстного меню, внесите изменения в появившемся диалоговом окне **Annotate**.





Введите текст в окне. Текст сразу отображается в аннотации. Доступны следующие опции и команды форматирования:

- **Name.** Отображает имя графика в тексте аннотации.
- **Time.** Отображает время в тексте аннотации.
- **Value.** Отображает значение кривой (амплитуды) в тексте аннотации.

**Alignment.** Задаёт выравнивание многострочного текста.



-  • **Align left.** Выравнивание по левому краю.
-  • **Center.** Выравнивание по центру.
-  • **Align right.** Выравнивание по правому краю.

**Orientation.** Изменить ориентацию текста.

-  • **Rotate left.** Повернуть влево.
-  • **Rotate right.** Повернуть вправо.

- **Apply to all annotations.** Выберите, чтобы применить данные установки для всех аннотаций графика.

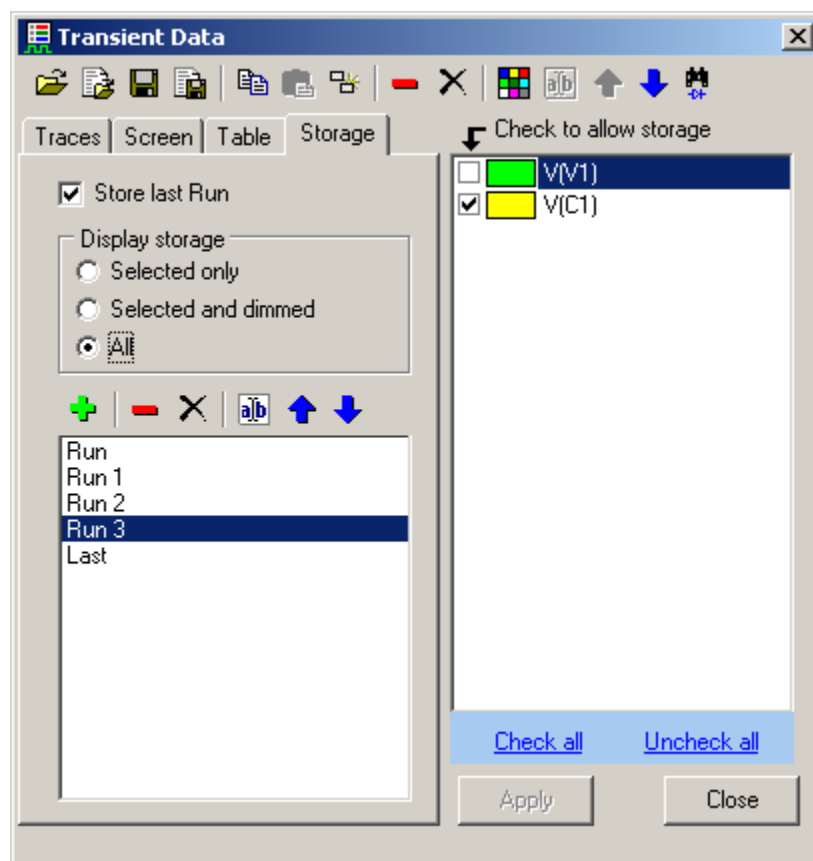
Чтобы переместить текст аннотации с сохранением привязки указателя к той же точке графика, щелкните по тексту аннотации и перетащите его. Чтобы переместить указатель, щелкните по концу указателя и перетащите его. Указатель изменит время, все еще следуя амплитуде графика. Текст аннотации будет перемещен с указателем.


Чтобы удалить аннотацию, щелкните правой клавишей мышки по аннотации и выберите команду **Delete annotation**  из контекстного меню. Чтобы удалить все аннотации, щелкните правой клавишей мышки по графику, выберите **Annotate**, затем выберите команду **Delete all** .

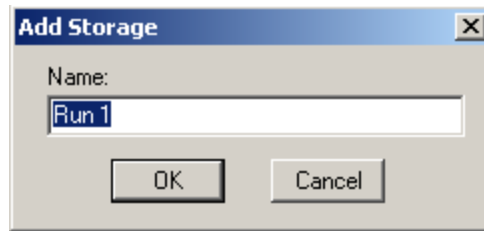
## Storage (накопитель данных, «хранилище»)

Результаты последнего запуска симуляции всегда показаны на закладке **Run** окна **Transient**. Данные последнего запуска могут быть перемещены в **storage** (накопитель данных), так что их можно будет сравнивать с другими запусками симуляции. Каждое сохранение данных имеет закладку в области **Data selection** (выбор данных). Сохраненные данные могут выбираться щелчком по закладке. Данные принадлежат кривой, так что, если кривая удалена, сохраненные данные будут также удалены.

Для доступа к командам, относящимся к сохранению данных, щелкните правой клавишей мышки по графику или области **Data selection**, затем выберите команду из контекстного меню. Список доступных сохраненных данных и команды, относящиеся к хранению и отображению данных, можно найти на странице **Storage** окна **Transient Data**:

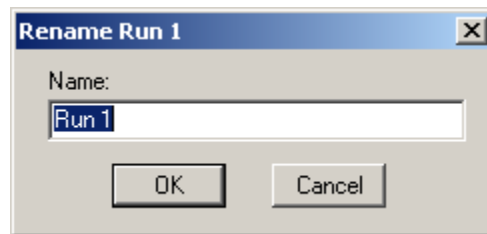


-  • **Move Run to storage.** Перенести последние данные симуляции в **storage**. Откроется окно **Add Storage**:



Введите новое имя или оставьте предложенное по умолчанию и щелкните по **OK**. Будет создана новая закладка с заданным именем в области **Data selection** окна **Transient**. Заметьте, что имена **Run** и **Last** являются зарезервированными и не должны использоваться.

- **Remove** – удалить выделенные данные из **storage**. Данные последней симуляции **Run** также могут быть удалены.
- ✕ • **Clear storage** – удалить все данные из **storage**.
- alb • **Rename** – переименовать выделенные данные. Откроется окно **Rename**:



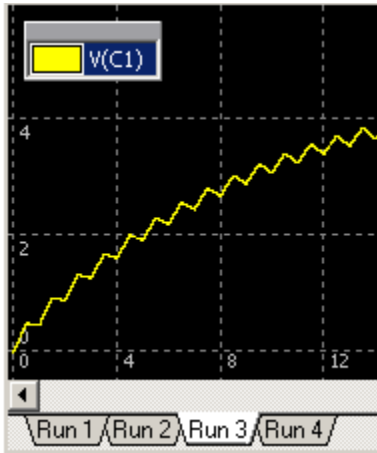
Введите новое имя и щелкните по **OK**. Заметьте, что имена **Run** и **Last** являются зарезервированными и не должны использоваться.

- ↑ • **Move selected up**. Передвинуть выделенные данные вверх (влево в области **Data Selection**).
- ↓ • **Move selected down**. Передвинуть выделенные данные вниз (вправо в области **Data Selection**).
- **Store last Run (сохранить последние данные)**. Если выбрана эта опция, то в момент запуска новой симуляции предыдущие (последние) данные будут перемещены в специальную закладку **Last** («последний»), стерев при этом находящиеся там данные. Таким образом, закладка **Last** всегда содержит предыдущие данные, которые можно использовать для сравнения с данными последней симуляции **Run**.

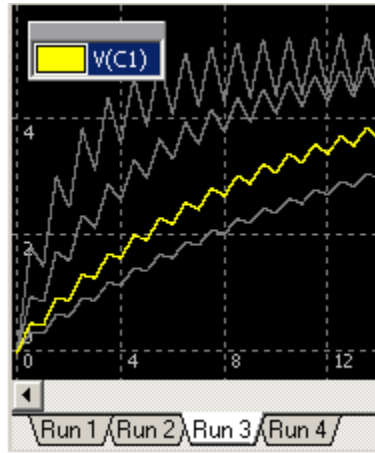
### Display storage (отображать storage)

- **Selected only**. Только выделенные данные отображаются на графике.
- **Selected and dimmed**. Выделенные данные отображаются с нормальным цветом кривой, а другие данные отображаются с пониженной цветностью, заданной на странице **Graphs** окна **Preferences**.
- **All**. Все данные отображаются с нормальными цветом кривых.

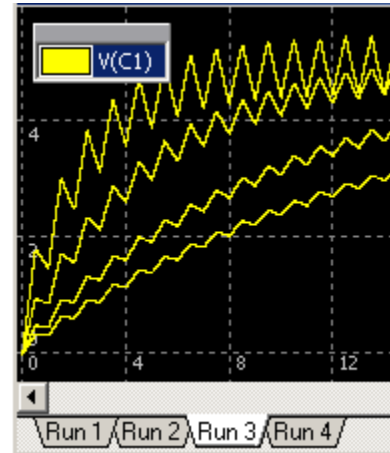
Например:



Selected only



Selected and dimmed



All

Когда страница **Storage** выбрана, флажки в списке кривых задают те кривые, для которых разрешено сохранение данных.

## Data table (Таблица данных)


Таблица данных показывает позиции курсоров, значения кривых и некоторые характеристики кривых, рассчитанных между курсорами, такие как: значение, максимум, минимум, RMS и т.д. Если курсоры выключены, таблица показывает данные левого и правого краев экрана и значения, рассчитанные между левым и правым краями экрана:

|         | left    | right   | delta |
|---------|---------|---------|-------|
| Cursors | 3.05609 | 6.92456 | 3.86  |
| V(V1)   | 10      | 0       | -10   |
| V(C1)   | 2.23089 | 3.63322 | 1.40  |

Cursors enabled, active cursor is highlighted

|        | left | right   | delta |
|--------|------|---------|-------|
| Screen | 0    | 20      | 20    |
| V(V1)  | 0    | 0       | 0     |
| V(C1)  | 0    | 4.66321 | 4.66  |

Cursors disabled, screen size = 20

- Чтобы показать/спрятать таблицу, щелкните по кнопке инструментальной панели **Table**  или щелкните правой клавишей мышки по графику и выберите команду **Table** из контекстного меню.
- Щелкните по строке кривой в таблице для выбора кривой. Выделенная кривая будет показана поверх всех других.
- Дважды щелкните по таблице, чтобы открыть окно **Transient Data**.

- Щелкните правой клавишей мышки, чтобы открыть контекстное меню. Меню будет содержать некоторые общие команды и команды, относящиеся к выделенной кривой.
- Цвета, шрифты и количество значащих цифр, используемые в таблице, могут задаваться на странице **Table** диалогового окна **Preferences**.

Таблица может отображаться внизу окна **Transient** или как отдельное окно: щелкните правой клавишей мышки по таблице и выберите команду **Separate window**:

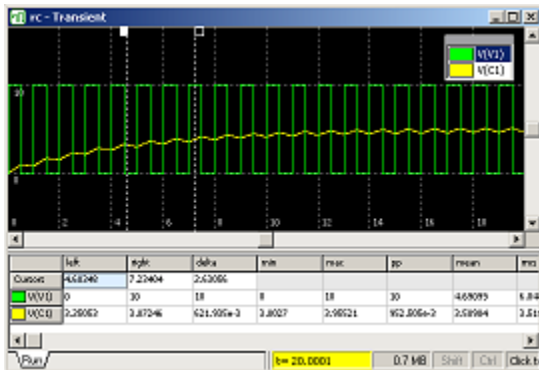


Table in the Transient Window

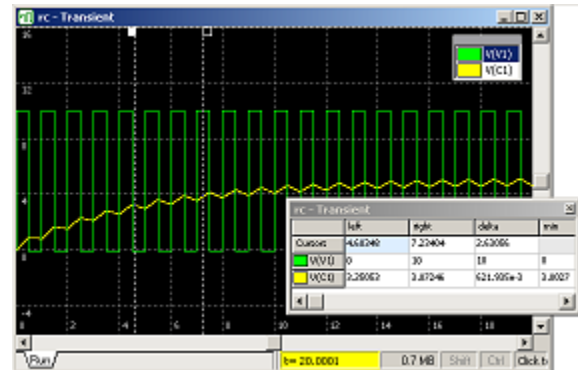
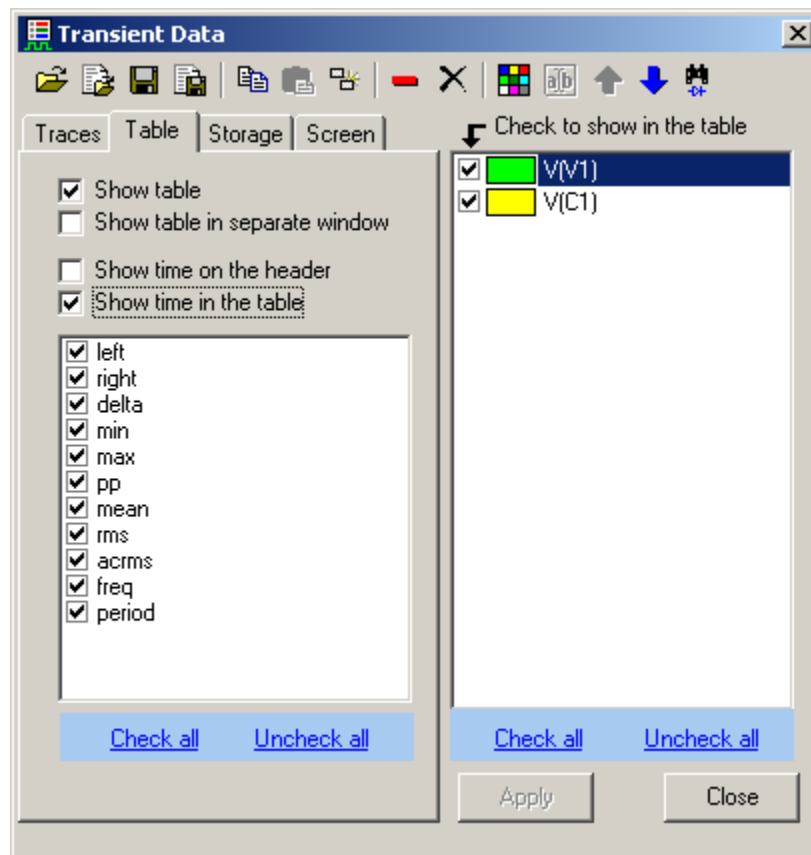


Table in separate window

Значения, показанные в таблице, также как и другие опции таблицы, могут выбираться на странице **Table** окна **Transient Data**:



- **Show table.** Выберите, чтобы показывать таблицу.
- **Show table in separate window.** Если выбрано, таблица будет показана в отдельном окне.

- **Show time on the header.** Если выбрано, позиция курсоров будет показана в заголовке строки, в колонках **left**, **right** и **delta**.

| Cursors | 3.67505 | 6.46035 | 2.7853  |
|---------|---------|---------|---------|
| V(V1)   | 0       | 10      | 10      |
| V(C1)   | 2.79145 | 3.90708 | 1.11563 |

- **Show time in the table.** Если выбрано, позиции курсоров будут показаны в отдельной строке.


|         | left    | right   | delta   |
|---------|---------|---------|---------|
| Cursors | 3.67505 | 6.46035 | 2.7853  |
| V(V1)   | 0       | 10      | 10      |
| V(C1)   | 2.79145 | 3.90708 | 1.11563 |



- **Table values.** Выберите значения для отображения в таблице:
  - **left** – значение кривой под левым курсором.
  - **right** – значение кривой под правым курсором.
  - **delta** – правое значение минус левое.
  - **min** – минимум кривой между курсорами.
  - **max** – максимум кривой между курсорами.
  - **pp** – значение кривой от пика до пика между курсорами.
  - **mean** – среднее значение кривой между курсорами.
  - **rms** – RMS (среднеквадратичное) значение кривой между курсорами.
  - **acrms** – AC RMS значение кривой между курсорами: RMS вычисляется по данным кривой после вычета из них среднего значения кривой.
  - **freq** – вычисленная частота сигнала между курсорами. Частота вычисляется на основании количества и величин интервалов между точками, где кривая пересекает средний уровень.
  - **period** –  $1 / \text{freq}$

Когда страница **Table** выбрана, флажки в списке кривых задают те кривые, которые будут показаны в таблице.

## Прокрутка и масштабирование

Чтобы прокрутить график, используйте любой из следующих методов:



- Переместите указатель мышки к левому или правому краю графика. Указатель мышки примет вид «большой стрелки». Щелкните или удержите левую клавишу мышки, чтобы прокрутить график.
- В режиме **Cursors**  : нажмите и держите клавишу **Shift**, затем щелкните и перетащите график.

- В режиме *Scrolling*  : щелкните и перетащите график.
- Нажмите клавишу **Ctrl** и поверните колесико мышки, чтобы прокрутить горизонтально.
- Нажмите клавишу **Shift** и поверните колесико мышки, чтобы прокрутить вертикально.
- Нажмите клавиши **Right** и **Left** (курсорная панель клавиатуры).
- Нажмите клавишу **End**, чтобы отцентровать кривые (установить в середине экрана).
- Нажмите **Ctrl-End**, чтобы отцентровать конец кривых.
- Нажмите **Shift-End**, чтобы отцентровать середину кривых.
- В режиме *Zoom*  : дважды щелкните по графику, чтобы отцентровать эту точку.

Чтобы масштабировать график, используйте любой из следующих методов:

- Поверните колесико мышки, чтобы масштабировать горизонтально.
- Нажмите клавиши **Ctrl** и **Shift** и поверните колесико мышки, чтобы масштабировать вертикально.
- Щелкните кнопку инструментальной панели, или используйте горячие клавиши, или щелкните правой клавишей мышки по графику, выберите **Zoom**, затем выберите команду:
  - ↔ **Horizontal Zoom-in (Ctrl-PgUp)** – горизонтальное увеличение.
  - ⇐ **Horizontal Zoom-out (Ctrl-PgDn)** – горизонтальное уменьшение.
  - ↔ **Fit the screen horizontal (Ctrl-Home)** – автомасштабировать горизонтально.
  - ⏏ **Fit cursors to screen** – привести курсоры к экрану
  - ⬆ **Vertical Zoom-in (PgUp)** - вертикальное увеличение.
  - ⬇ **Vertical Zoom-out (PgDn)** - вертикальное уменьшение.
  - ⬆ **Fit the screen vertical (Home)** – автомасштабировать вертикально.
  - ⏏ **Fit the screen (Shift-Home)** – автомасштабировать горизонтально и вертикально.
  - ⬆ **Reset vertical scale (set Multiplier=1, Offset=0)** – сбросить вертикальный множитель и смещение.
  - **Zoom-in (Shift-PgUp)** – увеличить (по вертикали и горизонтали).
  - **Zoom-out (Shift-PgDn)** – уменьшить (по вертикали и горизонтали).

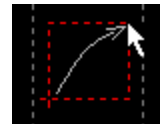
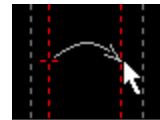
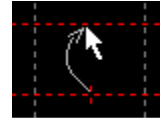
Чтобы расширить выделенную область:

- В режиме *Zoom*  : щелкните и растащите выделенную область.
- В режиме *Cursors*  : нажмите и удержите клавишу **Ctrl**, затем щелкните и растащите выделенную область.

Выделение области экрана зависит от того, как указатель мышки перемещается относительно начальной точки.



- Если указатель мышки перемещается только вверх или вниз, будут показаны две горизонтальные линии, а когда левая клавиша мышки будет отпущена, выделенная область увеличится по вертикали.
- Если указатель мышки движется только влево и вправо, будут показаны две вертикальные линии, а когда левая клавиша мышки будет отпущена, выделенная область увеличится по горизонтали.
- Если указатель мышки движется диагонально, будет показан прямоугольник, а когда левая клавиша будет отпущена, выделенная область увеличится до заполнения экрана.















Чтобы отменить или выполнить повторно ранее отмененное прокручивание и масштабирование, щелкните кнопки инструментальной панели:


















- ↶ • **Undo** - отменить.
- ↷ • **Redo** – выполнить повторно.








## Команды переходного процесса

Следующие команды, кнопки и горячие клавиши доступны в Основном Меню, на инструментальной панели переходного процесса и в контекстных меню.












-  • **Open/Show transient window (F5)** – открыть/показать окно переходного процесса.
-  • **Transient Settings** – открыть диалоговое окно **Transient Settings**.
-  • **Transient Data** – открыть окно **Transient Data**.
-  • **Continuous transient mode** - режим непрерывной симуляции.
-  • **Start (F6)** - запустить анализ.
-  • **Pause (Space)** - приостановить анализ (Пауза).
-  • **Continue (F7, Space)** - продолжить анализ.
-  • **Stop** - остановить анализ.
-  • **Log** – показать журнал регистрации анализа.
-  • **Sweep** – запустить серию анализов с изменяемым параметром.
-  • **Save IC** – сохранить текущее состояние в начальные условия.
-  • **Preferences** – открыть диалоговое окно **Preferences**.

### Toolbar and some context menus (инструментальная панель и контекстные меню):





-  • **Cursors.** Перемещение курсоров.
-  • **Zoom.** Масштабирование.
-  • **Scrolling.** Прокрутка.
-  • **Horizontal Zoom-in (Ctrl-PgUp)** – горизонтальное увеличение
-  • **Horizontal Zoom-out (Ctrl-PgDn)** – горизонтальное уменьшение.
-  • **Fit the screen horizontal (Ctrl-Home)** – автомасштабировать горизонтально.
-  • **Fit cursors to screen** – привести курсоры к экрану.
-  • **Vertical Zoom-in (PgUp)** - вертикальное увеличение.
-  • **Vertical Zoom-out (PgDn)** - вертикальное уменьшение.
-  • **Fit the screen vertical (Home)** – автомасштабировать вертикально.
-  • **Fit the screen (Shift-Home)** – автомасштабировать горизонтально и вертикально.
-  • **Reset vertical scale (set Multiplier=1, Offset=0)** – сбросить вертикальный множитель и смещение
-  • **Undo scale (Backspace).** Отменить прокрутку или масштабирование.
-  • **Redo scale.** Выполнить повторно отмененную прокрутку или масштабирование.
-  • Show/hide **Cursors** – скрыть/показать курсоры.
-  • Show/hide **Data Table** – скрыть/показать таблицу данных.
-  • Show/hide **Legend** – скрыть/показать условные обозначения (список кривых).

-  • **Separate traces** – разделить кривые.
-  • **Right maximum.** Переместить активный курсор к ближайшему справа максимуму выделенной кривой.
-  • **Left maximum.** Переместить активный курсор к ближайшему слева максимуму выделенной кривой.
-  • **Right minimum.** Переместить активный курсор к ближайшему справа минимуму выделенной кривой.
-  • **Left minimum.** Переместить активный курсор к ближайшему слева минимуму выделенной кривой.
-  • **Maximums.** Переместить один курсор к ближайшему справа максимуму, а другой – к ближайшему слева максимуму.
-  • **Minimums.** Переместить один курсор к ближайшему справа минимуму, а другой – к ближайшему слева минимуму.

### Graph commands (команды контекстного меню окна графиков):






-  • **Cursors** – открыть диалоговое окно **Cursors**.
- **Traces** ► (Следующие команды применяются только к кривым, показанным на графике)
  -  ○ **Open** - загрузить кривые из файла данных типа “nlt”.
  -  ○ **Import** -импортировать кривые из текстового файла формата “txt” или “csv”.
  -  ○ **Save** - сохранить кривые в файл данных типа “nlt”.
  -  ○ **View/Export** - посмотреть и экспортировать кривые в текстовом виде (“txt” или “csv” формат).
  -  ○ **Copy** - скопировать кривые в буфер обмена.
  -  ○ **Paste** - вставить кривые из буфера обмена.
-  • **Image** ►
  -  ○ **Copy to clipboard.** Скопировать изображение окна переходного процесса в буфер обмена.
  -  ○ **Save as BMP.** Сохранить изображение окна переходного процесса в файле формата «bmp».
  -  ○ **Save as JPG.** Сохранить изображение окна переходного процесса в файле формата «jpg».

### Storage command (команды хранения данных):




-  • **Move Run to storage** - перенести последние данные симуляции в **storage**.
-  • **Remove** - удалить выделенные данные из **storage**
-  • **Clear storage** - удалить все данные из **storage**
-  • **Rename** - переименовать выделенные данные.
- **Store last Run** – в начале анализа сохранять последние данные под именем **Last**.

- **Selected only**
  - **Selected and dimmed**
  - **All**
- } Виды отображения **storage**.

#### Annotation commands (команды аннотаций):

-  • **Annotate selected trace** – аннотировать выделенную кривую.
-  • **Annotate all traces** – аннотировать все кривые.
-  • **Edit annotation** – редактировать аннотацию.
-  • **Delete annotation** – удалить аннотацию.
-  • **Delete all** – удалить все аннотации.







#### Text commands (команды текста):

-  • **Insert text** – вставить текст на графике.
-  • **Edit text** – редактировать текст.
-  • **Delete text** – удалить текст.

#### Data table commands (команды таблицы данных):

- **Hide trace name** - не показывать данную кривую в таблице.
- **Separate window** – показать таблицу в отдельном окне.

#### Legend commands (команды legend – условных обозначений):

- **Hide trace name** - не показывать данную кривую на графике.
-  • **Remove** – удалить данную кривую.
-  • **Rename** – переименовать данную кривую.
-  • **Duplicate** – сдублировать данную кривую.
-  • **Copy** – скопировать данную кривую в буфер обмена.
-  • **Paste** – вставить кривые из буфера обмена.
-  • **Find component** - показать компонент на схеме.

## Клавиатура и горячие клавиши

Следующие клавиши клавиатуры и горячие клавиши можно использовать в окне переходного процесса:

- **Space** – приостановить/продолжить анализ переходного процесса.

- **Tab** – разделить кривые.
- **Left, Right** – прокрутить горизонтально.
- **Up, Down** – выбрать кривую.
- **End** - отцентровать кривые (установить в середине экрана).
- **Ctrl-End** - отцентровать конец кривых.
- **Shift-End** - отцентровать середину кривых.
- **Shift-PgUp** - увеличить (по вертикали и горизонтали).
- **Shift-PgDn** – уменьшить (по вертикали и горизонтали).

## Операции с мышкой

Следующие операции с мышкой можно использовать в окне переходного процесса.

- **Щелчок** (левая клавиша) - открыть контекстное меню.
- **Колесико мышки** – горизонтальное увеличение/уменьшение.
- **Ctrl- Колесико мышки** – горизонтальная прокрутка.
- **Shift- Колесико мышки** – вертикальная прокрутка.
- **Ctrl-Shift- Колесико мышки** – вертикальное увеличение/уменьшение.

В режиме *Cursors*  :

- **Щелчок** (левая клавиша) – если курсоры показаны, поставить ближайший курсор в эту точку.
- **Щелчок и перетаскивание (Click and drag).**
  - На аннотации – передвинуть текст или указатель аннотации.
  - На тексте – передвинуть текст или указатель текста.
  - Иначе – передвинуть курсор.
- **Двойной щелчок (Double-click).**
  - На аннотации – редактировать аннотацию.
  - На тексте – редактировать текст.
  - Иначе – показать курсоры, поставить оба курсора в эту точку.

В режиме *Zoom*  :

- **Щелчок и перетаскивание (Click and drag)** – выбрать и отмасштабировать.
- **Двойной щелчок (Double-click)** - центрировать экран.








В режиме *Scrolling*  :

- **Щелчок и перетаскивание (Click and drag)** – прокрутить график.
- **Двойной щелчок (Double-click)** - центрировать экран.

## Инструменты переходного процесса (Transient Tools)

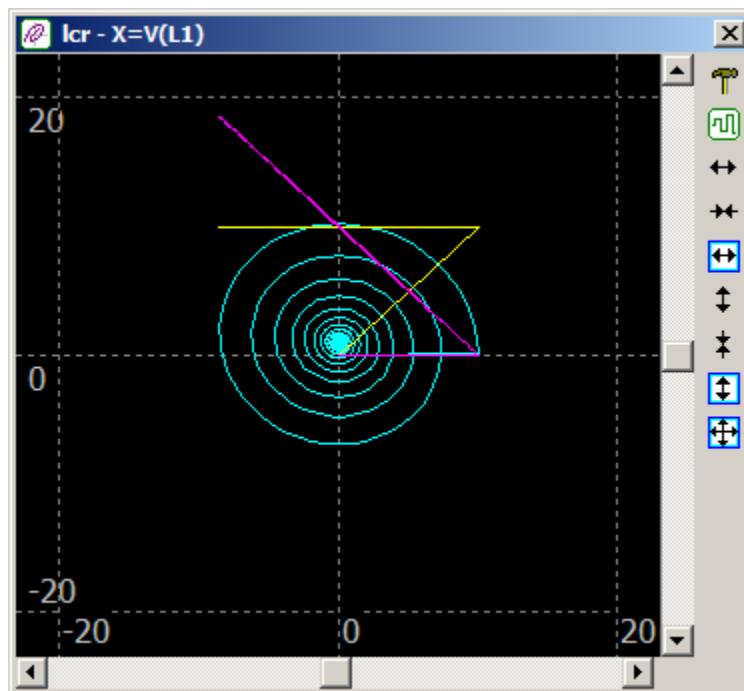
**Transient Tools** позволяют сделать различные виды анализа данных переходного процесса и представить данные в разных форматах. Выберите **Transient | Tools** и затем строку с требуемым инструментом.

В настоящее время доступны следующие инструменты:

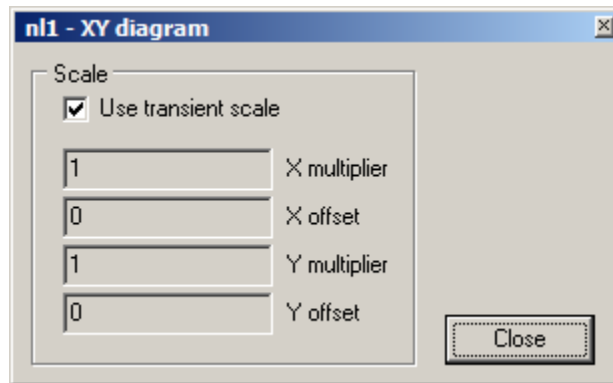
-  • XY diagram (XY-диаграмма)
-  • Histogram (гистограмма)
-  • FFT (Быстрое Преобразование Фурье)
-  • Eye diagram
-  • Markers (маркеры)
-  • Power (мощность)
-  • Line snapshot

### XY diagram (XY-диаграмма)








XY-диаграмма показывает все кривые, как функцию выделенной кривой. Имя выделенной кривой (ось X) показано в строке заголовка. Диаграмма показывает кривые только между курсорами (или на всем экране, если курсоры выключены).



- **Settings.** Открывает диалог **Settings**:



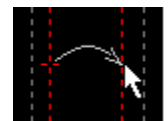
По умолчанию оси X и Y диаграммы имеют те же масштабы, что и вертикальная шкала экрана переходного процесса. Снимите флажок **Use transient scale** и введите нужные множители и смещение для осей.

- **Use transient scale.**
-  • **Horizontal Zoom-in.**
-  • **Horizontal Zoom-out.**
-  • **Fit the screen horizontal.**
-  • **Vertical Zoom-in.**
-  • **Vertical Zoom-out.**
-  • **Fit the screen vertical.**
-  • **Fit the screen.**
- Щелкните правой клавишей мышки по окну, чтобы увидеть доступные команды.
- Дважды щелкните по окну, чтобы отцентровать кривые.
- Щелкните левой клавишей мышки по окну и растащите выделенную область. Выделение области экрана зависит от того, как указатель мышки перемещается относительно начальной точки.

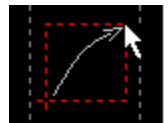
- Если указатель мышки перемещается только вверх или вниз, будут показаны две горизонтальные линии, а когда левая клавиша мышки будет отпущена, выделенная область увеличится по вертикали.



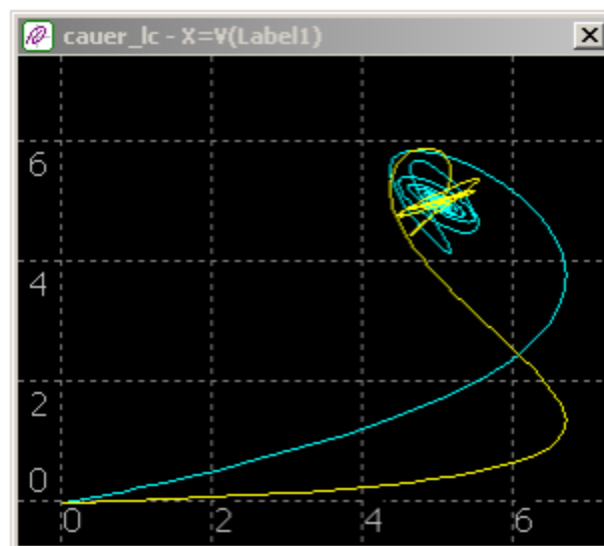
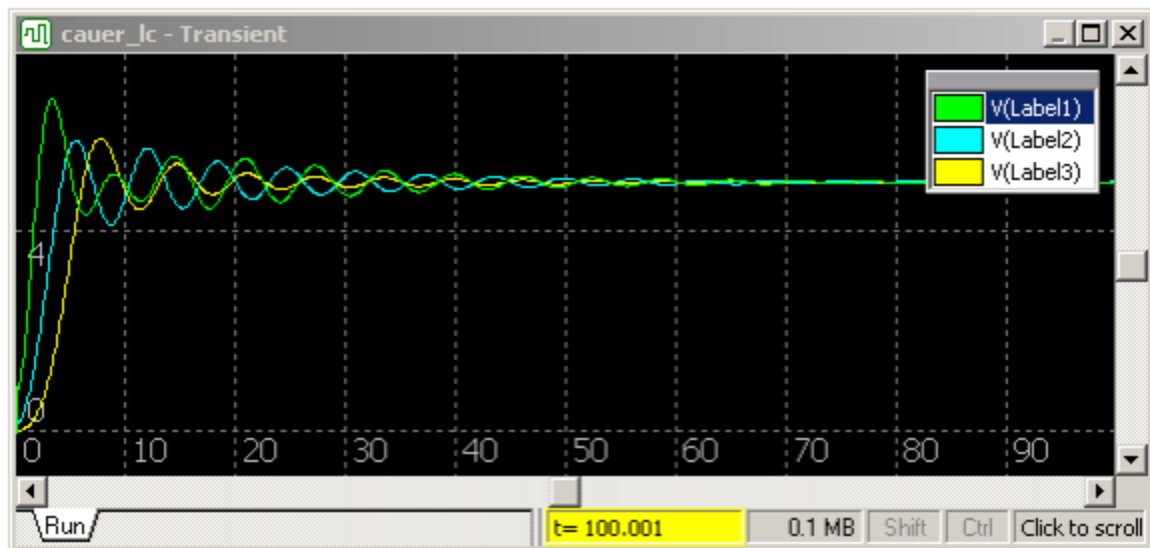
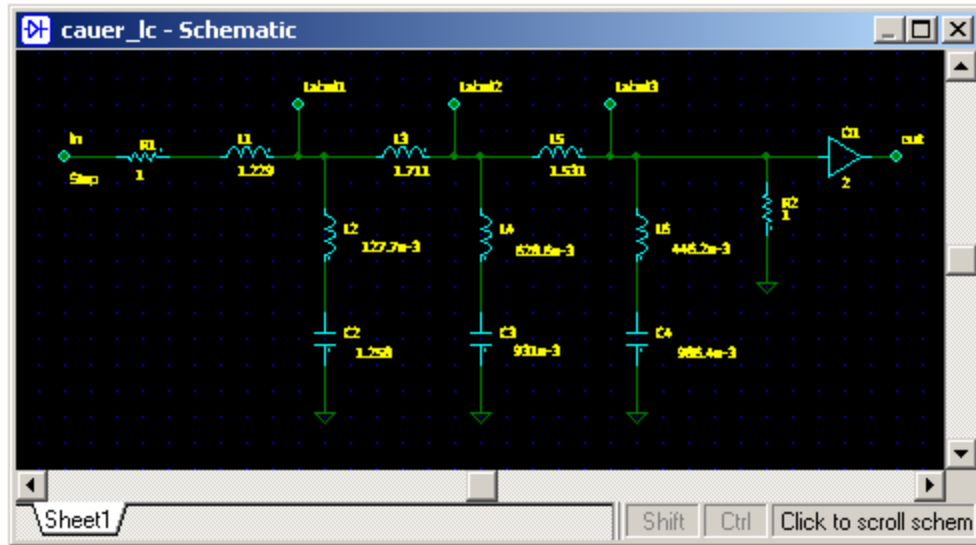
- Если указатель мышки движется только влево и вправо, будут показаны две вертикальные линии, а когда левая клавиша мышки будет отпущена, выделенная область увеличится по горизонтали.



- Если указатель мышки движется диагонально, будет показан прямоугольник, а когда левая клавиша будет отпущена, выделенная область увеличится до заполнения экрана.



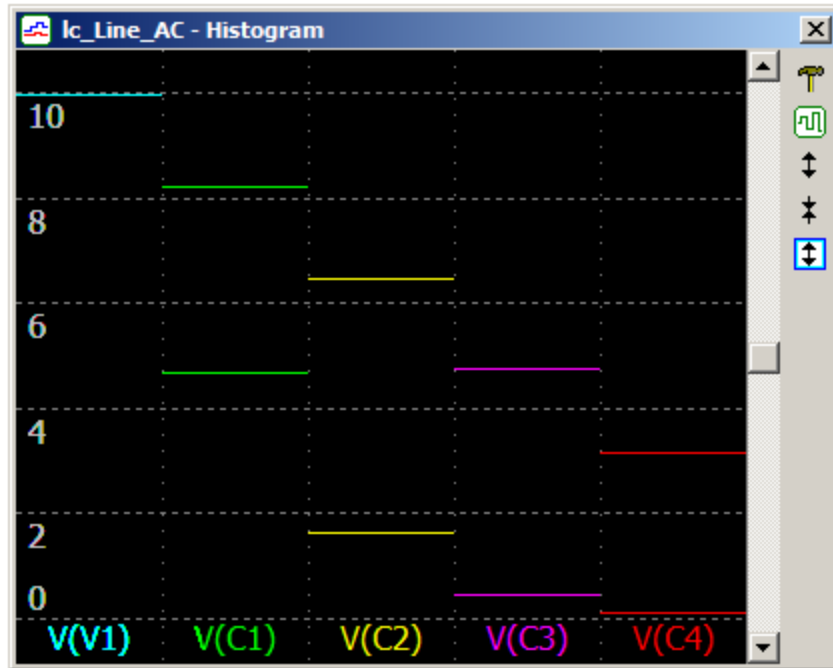
Пример: схема, анализ переходного процесса и XY-диаграмма.




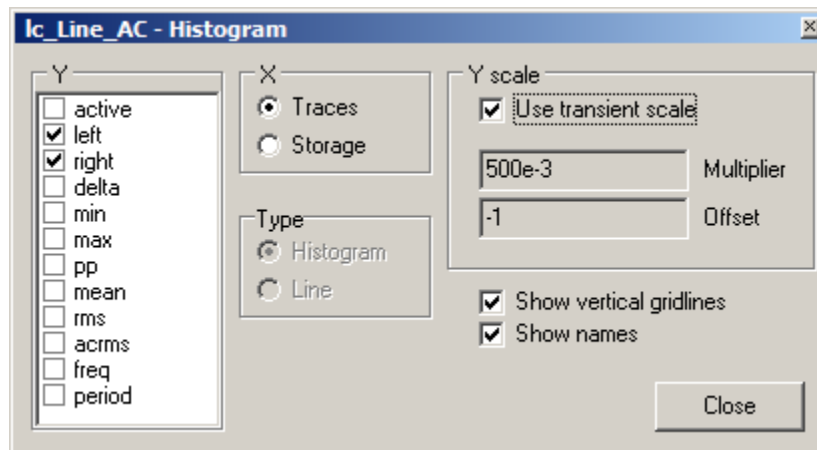


## Histogram (гистограмма)

Гистограмма представляет значения кривой и некоторые ее расчетные характеристики, полученные между курсорами (или на всем экране, если курсоры выключены), в графическом формате. Гистограмма может также показать «поперечное сечение» кривых или сохраненных данных.



-  • **Settings.** Открывает диалог **Settings**:




- **Y values.** Выберите переменные, отображаемые по оси Y. **active** — это выделенный в настоящий момент курсор (левый или правый). Другие значения – это то, что отображено в таблице данных переходного процесса.
- **X.** Выбирает режим гистограммы: что будет показано по оси X.

- **Traces.** Показать «поперечное сечение» всех кривых, отображаемых сейчас на графике.
- **Storage.** Показывает «поперечное сечение» **Storage** для всех кривых, отображаемых сейчас на графике.
- **Type.** Выбирает тип гистограммы для режима **Storage**:
  - **Histogram** - гистограмма.
  - **Line** - линии.
- **Y scale.** По умолчанию шкала Y диаграммы имеют тот же масштаб, что и вертикальная шкала экрана переходного процесса. Снимите флажок **Use transient scale** и введите нужные множители и смещение.
  - **Show vertical gridlines** – показать вертикальные линии, разделяющие данные гистограммы.
  - **Show names** – показать имена кривых или **storage** на оси X.

 • **Use transient scale.**

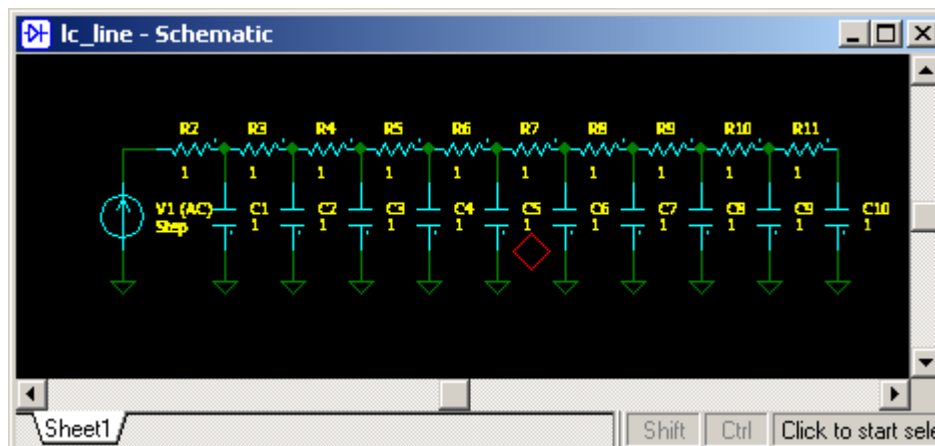
 • **Vertical Zoom-in.**

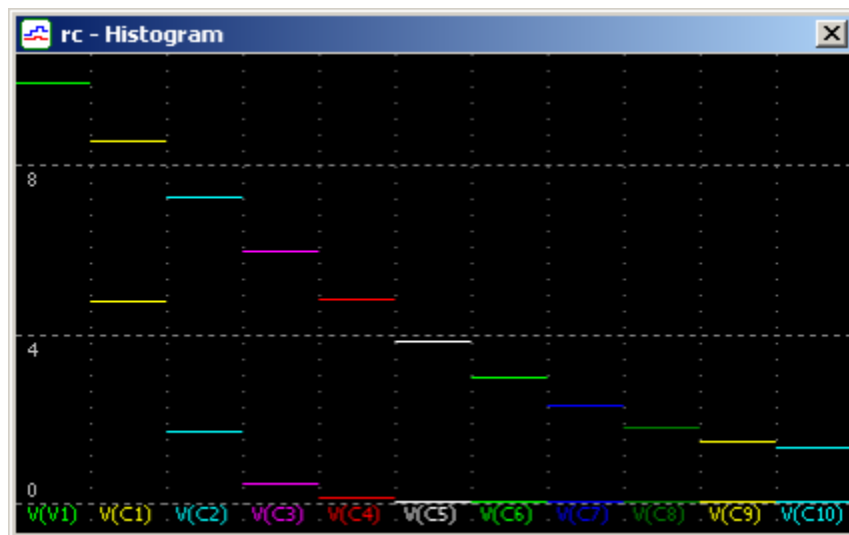
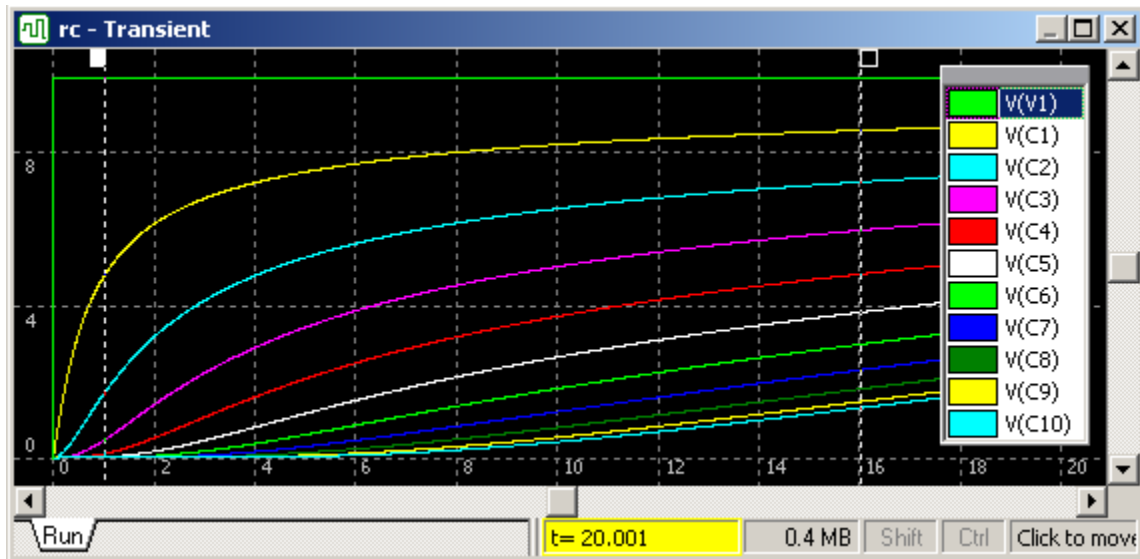
 • **Vertical Zoom-out.**

 • **Fit the screen vertical.**

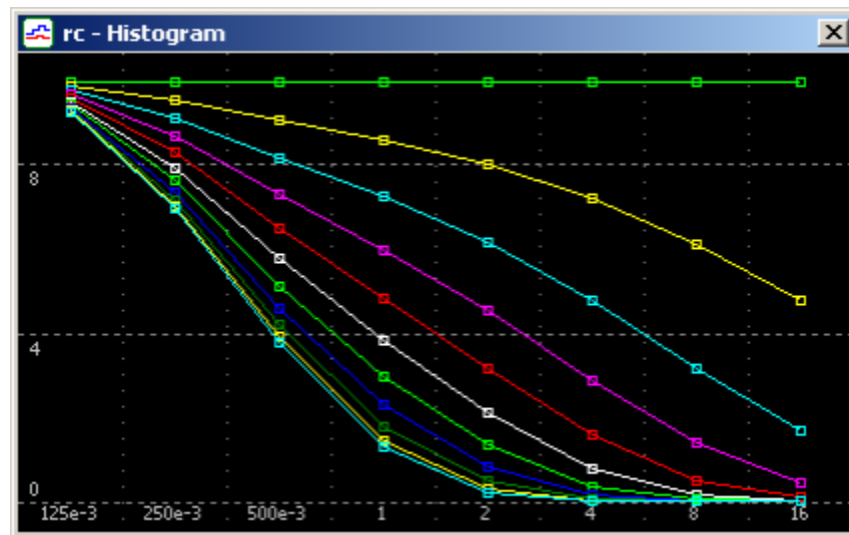
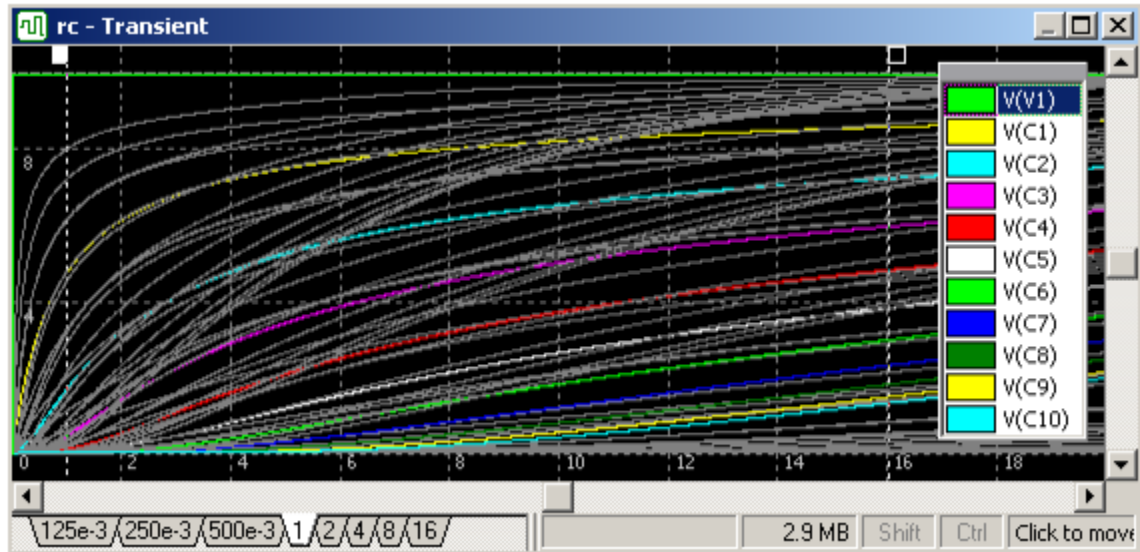
- Щелкните правой клавишей мышки по окну, чтобы увидеть доступные команды.
- Дважды щелкните по окну, чтобы открыть диалоговое окно **Settings**.

Режим **Traces** или «поперечное сечение кривых» могут использоваться для отображения «пространственного» распространения сигнала по схеме. Следующий пример показывает моделирование теплопроводности стержня, использующее электрические аналогии. RC цепь моделирует одномерный стержень с источником температуры (V1), приложенным к одному концу. Кривые показывают температуру на некотором расстоянии от этого конца. Если температура изменилась, как ступенька, температурный фронт распространяется по стержню. Гистограмма показывает распределение температуры вдоль стержня при  $t=1$  (левый курсор) и  $t=16$  (правый курсор).





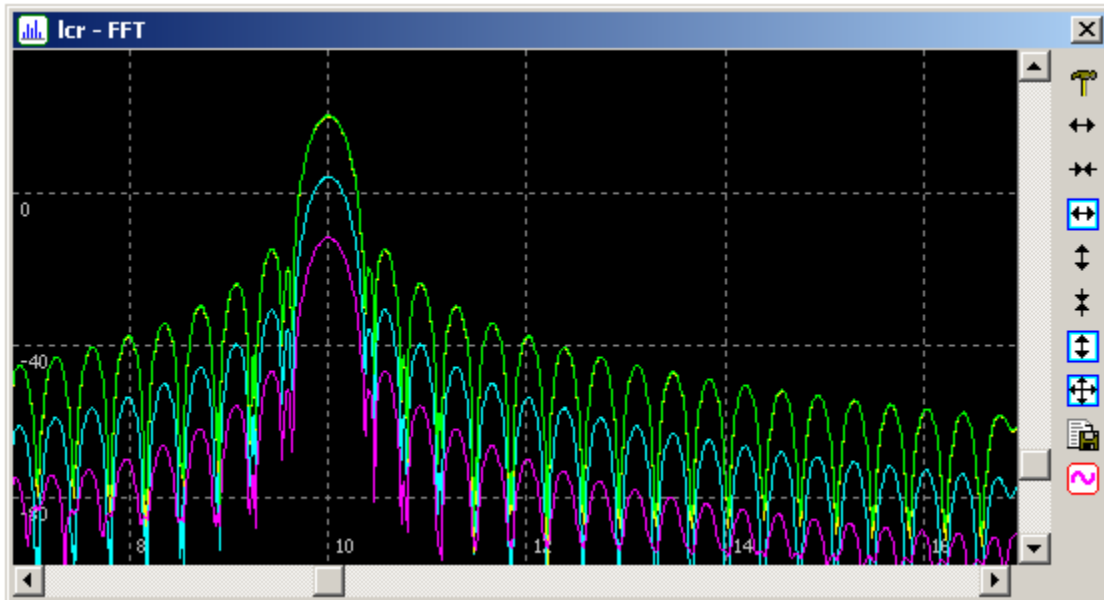
Режим **Storage** или «поперечное сечение **Storage**» может быть использован, чтобы показать, как значения кривых в какое-то время зависят от параметров схемы. Следующий пример показывает моделирование предыдущей схемы с сопротивлением (эквивалентным обратной величине теплопроводности), меняющимся от 0.125 до 16 с шагом X2, с помощью **sweep** переходного процесса. Каждый запуск запоминается в **Storage**. Ось X гистограммы — данные **Storage** (то есть сопротивление). Линии разного цвета показывают температуру на некотором расстоянии от конца при  $t=16$  (активный курсор), как функцию сопротивления (теплопроводности).



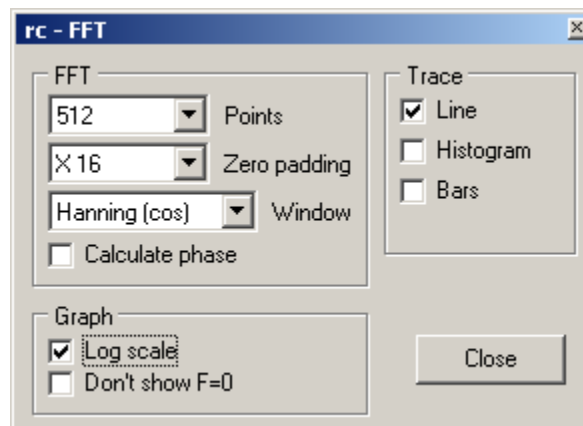
Можно заметить, что «поперечное сечение» гистограммы Storage при  $R=1$  то же, что и верхняя линия гистограммы Trace, показанной в предыдущем примере.

## FFT (Быстрое Преобразование Фурье)

FFT (Быстрое Преобразование Фурье) вычисляется только между курсорами (или на всем экране, если курсоры выключены) для всех кривых, отображенных на графике. График FFT не показывает фазу.



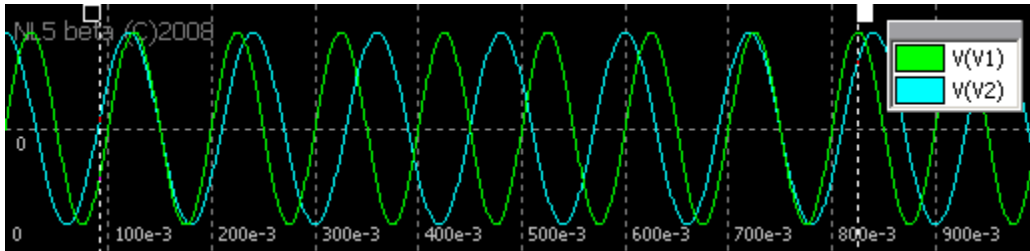
- **Settings.** Открывает диалог **Settings**:



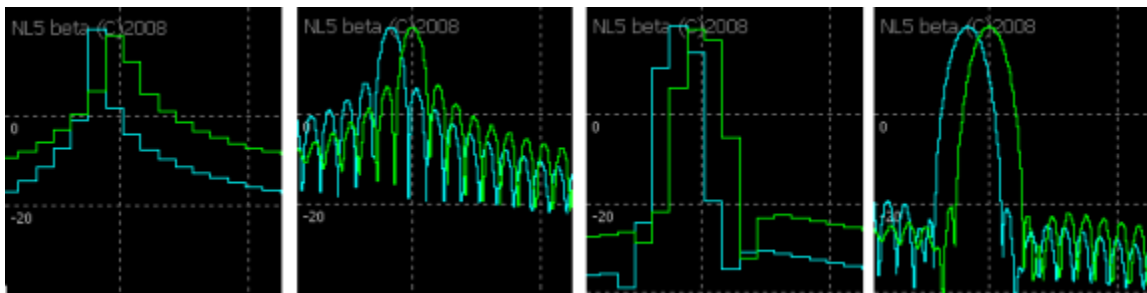
- **FFT.**
  - **Points.** Количество точек FFT, **256...1048576** ( $2^8...2^{20}$ ).
  - **Zero padding.** Добавляет нули к данным переходного процесса. Может быть от **None** до **X16**. Дополнение нулями (zero padding) — это стандартная техника для улучшения спектрального разрешения и предупреждения наложения пиков.

- **Window.** Кадрирование (windowing) — это стандартная техника для уменьшения эффектов просачивания и улучшения спектрального разрешения. В настоящий момент доступно 10 окон (включая прямоугольник).
- **Calculate phase.** Хотя фаза не имеет особого значения для FFT и не отображается на графике, она может быть рассчитана и отображена в окне AC (см. «Показать в окне AC»).

Следующий пример показывает эффекты **Zero padding** и **Window** на результатах FFT:



Transient



No zero padding  
No window

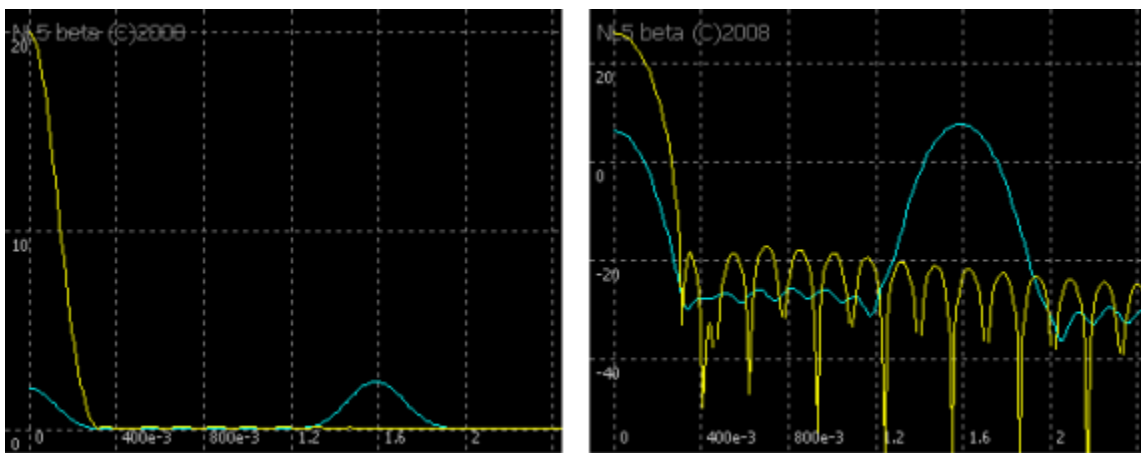
X16 zero padding  
No window

No zero padding  
Hamming window

X16 zero padding  
Hamming window

### ○ Graph.

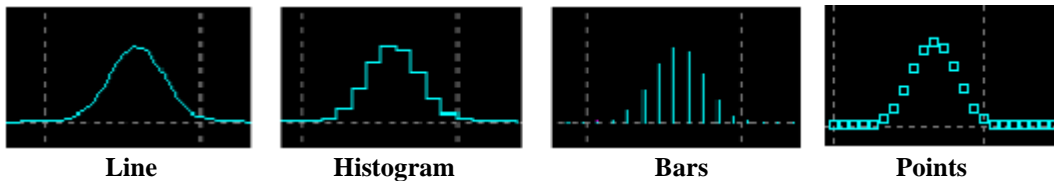
- **Log scale.** Если выбрано, амплитуда показана в ДБ.



Linear scale

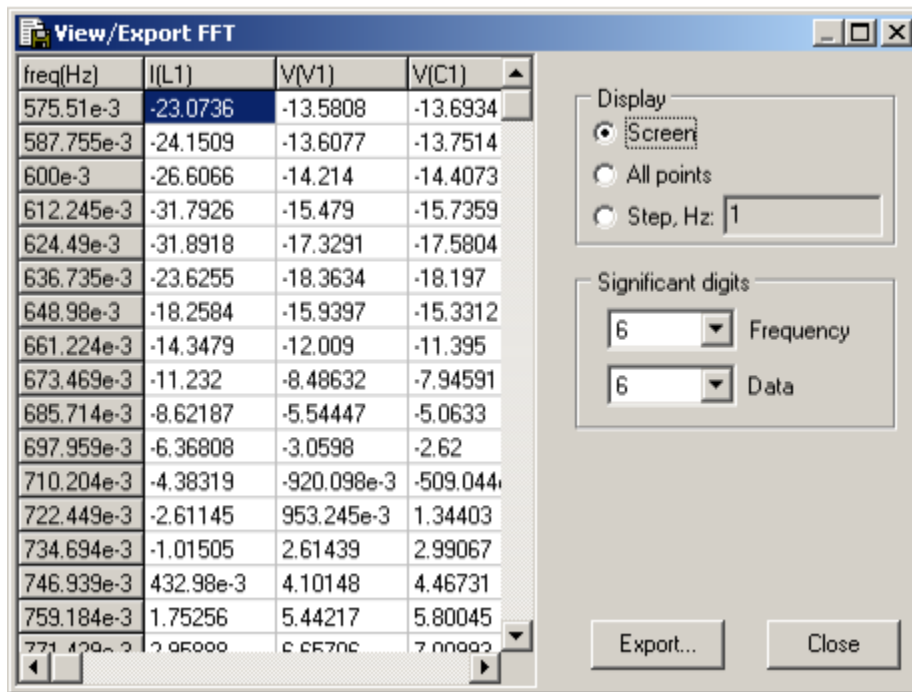
Log scale

- **Don't show F=0.** Если выбрано, точка нуля частоты не отображается на графике. Выбирайте эту опцию, если вас не интересует DC компонента сигнала (постоянная составляющая).
- **Trace.** Вид отображения кривых.
  - **Line** - прямая линия между точками.
  - **Histogram** - гистограмма.
  - **Bars** - вертикальная линия от нуля к точке кривой. Не доступно для логарифмической шкалы.
  - **Points** – выводить точки данных квадратными маркерами.
  -



- ↔ • **Horizontal Zoom-in** – горизонтальное увеличение
- ↔ • **Horizontal Zoom-out** – горизонтальное уменьшение.
- ↔ • **Fit the screen horizontal** – автомасштабировать горизонтально.
- ↑ • **Vertical Zoom-in** - вертикальное увеличение
- ↓ • **Vertical Zoom-out** - вертикальное уменьшение.
- ↑ • **Fit the screen vertical**– автомасштабировать вертикально.
- ↔ • **Fit the screen** – автомасштабировать горизонтально и вертикально.

- **View/Export.** Посмотреть данные FFT в текстовой таблице, экспортировать в текстовом (“csv”) формате. Появится диалоговое окно **View/Export**:



Текстовая таблица показывает амплитуду всех FFT кривых. Выберите режим **Display**, чтобы показать:

- **Screen.** Показывает точки в диапазоне частот, видимые только на экране.
- **All points.** Показывает все рассчитанные FFT точки.
- **Step, Hz.** Показывает точки с заданным шагом по частоте. Этот режим может быть использован, чтобы увидеть только гармоники заданной частоты.

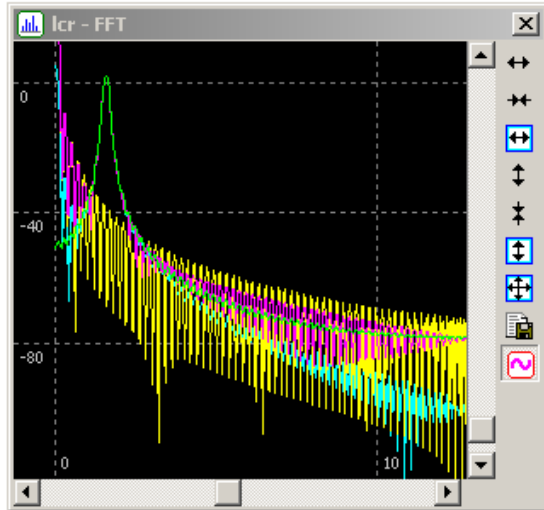
Задайте **Significant digits** для данных в колонках **Frequency** и **Data**.

Щелкните по **Export** чтобы экспортировать таблицу в текстовом (“csv”) формате.

- **Show in AC window (показать в окне AC).** Если выбрано, FFT кривые будут также показаны в AC окне. Имена кривых будут «FFT of *имя кривой*». При выключении этой опции, кривые будут удалены из AC окна. Если окно FFT закрывается с включенной опцией, кривые не будут удалены. Работа с AC кривыми в окне AC позволяет удобно масштабировать и прокручивать графики, использовать курсоры и таблицу данных, а также посмотреть фазу кривых. FFT кривые могут дублироваться в окне AC, чтобы использовать их как эталонные кривые для сравнения с новыми FFT кривыми. Заметьте, что FFT данных из **Storage** не показываются в окне AC.



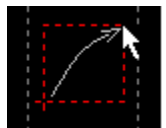
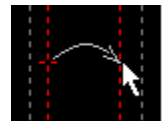
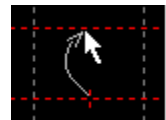
FFT window



AC window

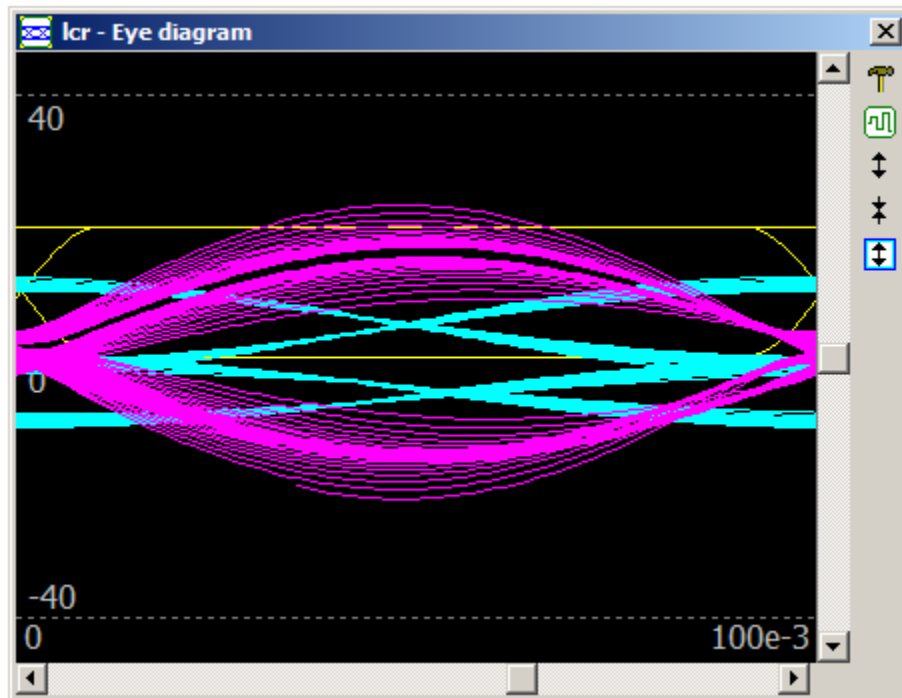


- Щелкните правой клавишей мышки по окну, чтобы увидеть доступные команды.
- Дважды щелкните по окну, чтобы отцентровать кривые.
- Щелкните левой клавишей мышки по окну и растяните выделенную область. Выделение области экрана зависит от того, как указатель мышки перемещается относительно начальной точки.
  - Если указатель мышки перемещается только вверх или вниз, будут показаны две горизонтальные линии, а когда левая клавиша мышки будет отпущена, выделенная область увеличится по вертикали.
  - Если указатель мышки движется только влево и вправо, будут показаны две вертикальные линии, а когда левая клавиша мышки будет отпущена, выделенная область увеличится по горизонтали.
  - Если указатель мышки движется диагонально, будет показан прямоугольник, а когда левая клавиша будет отпущена, выделенная область увеличится до заполнения экрана.
- Колесико мышки с **Ctrl** и **Shift** может использоваться для прокрутки и масштабирования изображения, в точности, как в окне переходного процесса.

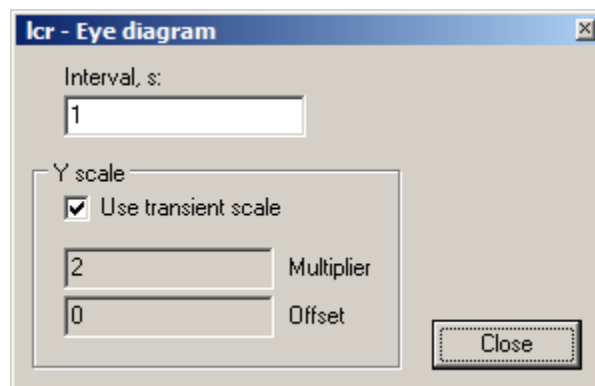


## Eye diagram (Глазковая диаграмма)





Глазковая диаграмма используется для анализа искажений и неустойчивости периодических сигналов. Хотя эта диаграмма применима в основном к телекоммуникационным и цифровым процессам, она может быть полезна и для аналоговой электроники. Диаграмма показывает кривые только между курсорами (или на всем экране, если курсоры выключены).



- 🔧 • **Settings.** Открывает диалог **Settings**:

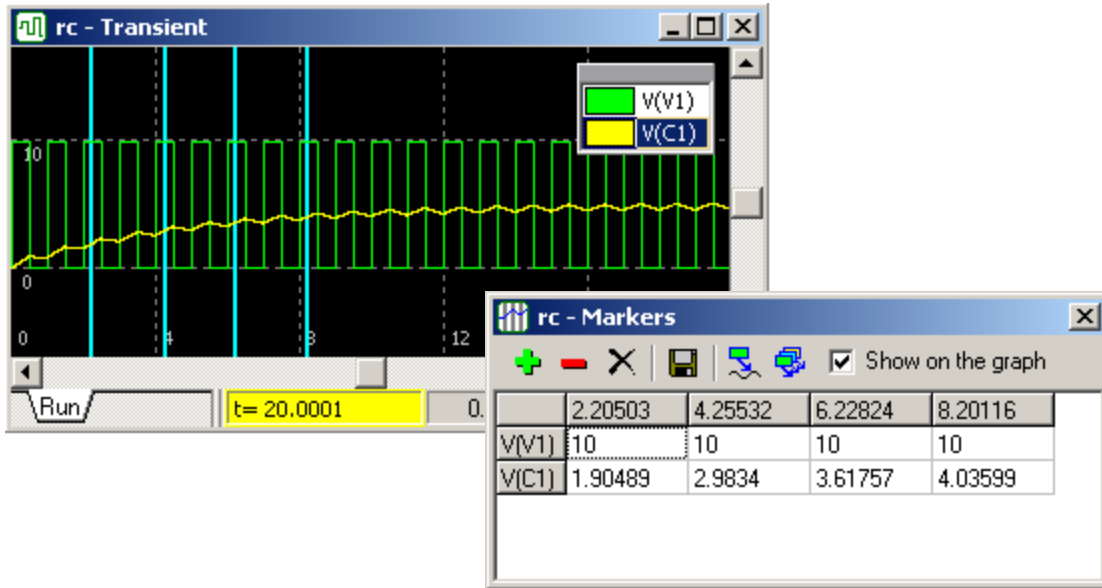


- **Interval, s.** Ширина диаграммы.
- **Y scale.** По умолчанию шкала Y диаграммы имеют тот же масштаб, что и вертикальная шкала экрана переходного процесса. Снимите флажок **Use transient scale** и введите нужные множители и смещение.

-  • **Use transient scale.**
-  • **Vertical Zoom-in.**
-  • **Vertical Zoom-out.**
-  • **Fit the screen vertical.**
  - Используйте горизонтальную полосу прокрутки, чтобы изменить фазу диаграммы.
  - Щелкните правой клавишей мышки по окну, чтобы увидеть доступные команды.
  - Дважды щелкните по окну, чтобы открыть диалоговое окно **Settings**.

## Markers (маркеры)

Маркеры предлагают удобный способ отслеживания амплитуд кривых в заданных точках. В отличие от курсоров, маркеры всегда остаются в заданном положении. Количество маркеров не ограничено. Ниже вы можете видеть 4 маркера, показанных на графике переходного процесса, с амплитудами кривых, отображаемыми в таблице **Markers table**

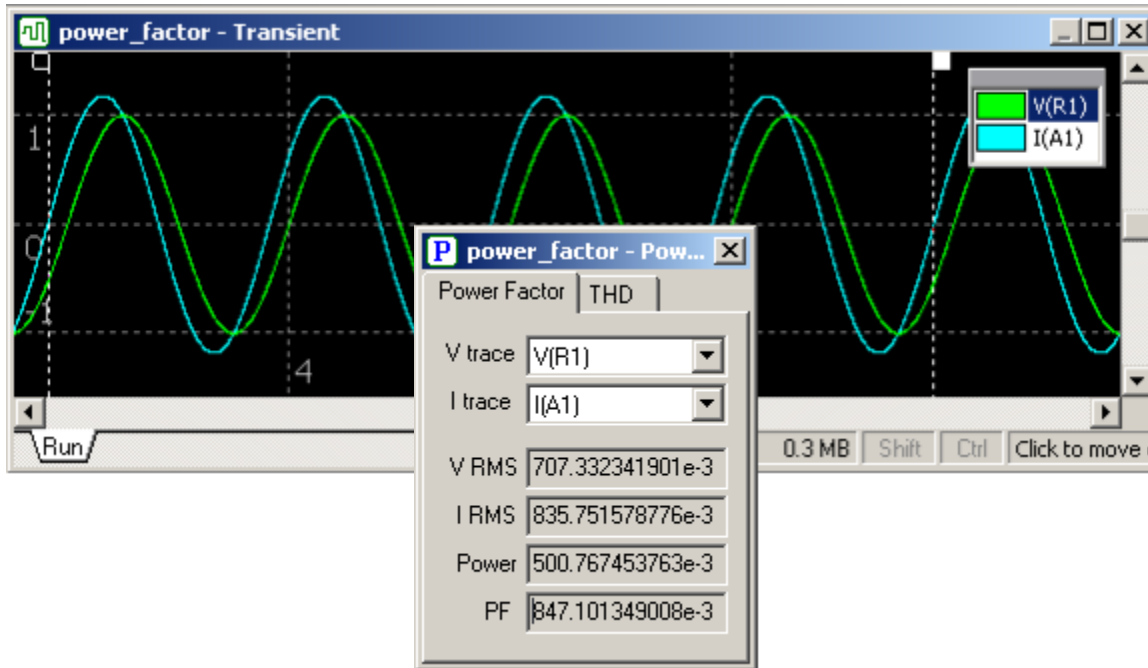


- **Таблица маркеров** содержит кривые, изображенные на графике.
- +** • **Add** – добавить новый маркер. Время маркера – это время активного курсора.
- • **Remove** – удалить маркер из таблицы. Щелкните любую ячейку в таблице, принадлежащую колонке маркера, затем щелкните по кнопке. На примере вверху выделен первый маркер.
- X** • **Delete** – удалить все маркеры.
- 📁** • **Export** – экспортировать таблицу маркеров в файл в текстовом (“csv”) формате.
- 📄** • **Annotate** – аннотировать выделенную кривую на позициях маркеров на графике переходного процесса.
- 📄** • **Annotate all traces** - аннотировать все кривые на позициях маркеров на графике переходного процесса.
- **Show on the graph** – выберите, чтобы показывать маркеры на графике переходного процесса. Ширина и цвет маркеров задаются на странице **Graphs** диалогового окна **Preferences**.

## Power (мощность)

Инструмент «Мощность» вычисляет Коэффициент мощности (Power Factor) и Коэффициент Нелинейных Искажений (THD – Total Harmonic Distortion).

Выберите страницу **Power Factor** для вычисления среднеквадратичного значения напряжения и тока, мощности и коэффициента мощности.

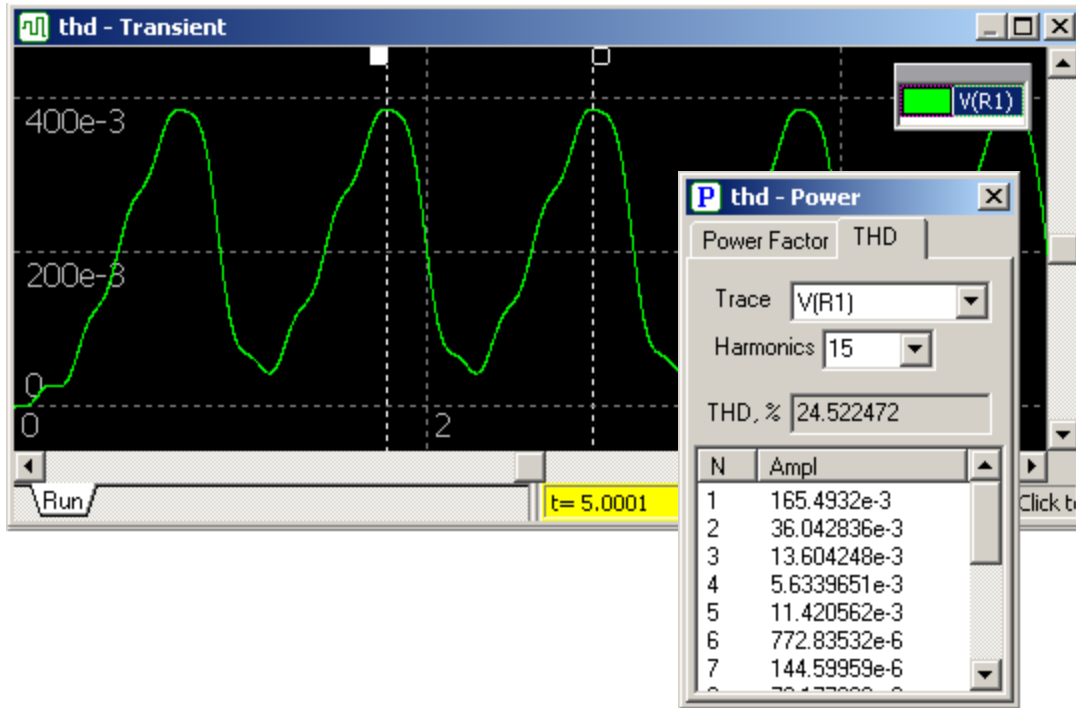


- **V trace.** Выберите кривую напряжения из выпадающего списка.
- **I trace.** Выберите кривую тока из выпадающего списка.

Коэффициент мощности PF вычисляется как: 
$$PF = \frac{Power}{VRMS \times IRMS}$$

Все параметры вычисляются только между курсорами (или на всем экране, если курсоры выключены), так что убедитесь в том, что выбранный интервал содержит целое число периодов сигнала.


Выберите страницу **THD** для вычисления Коэффициента Нелинейных Искажений (THD).

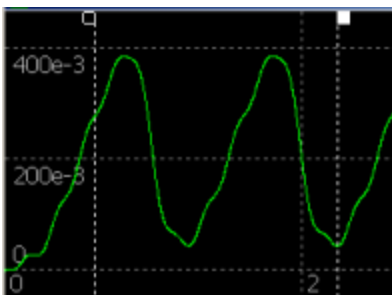


- **Trace.** Выберите кривую из выпадающего списка.
- **Harmonics.** Введите или выберите из выпадающего списка количество вычисляемых гармоник (максимум = 40).

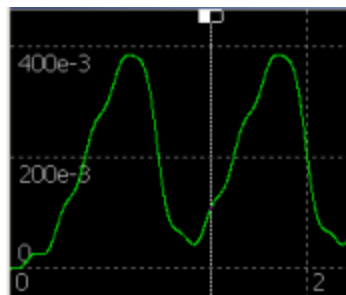
THD вычисляется как:  $THD = \frac{\sqrt{V_2^2 + V_3^2 + V_4^2 + \dots}}{V_1}$ , где  $V_n$  – амплитуда n-ой гармоники.

THD вычисляется только между курсорами (или на всем экране, если курсоры выключены), так что убедитесь в том, что выбранный интервал содержит в точности один период основной (первой) гармоники сигнала. Это можно сделать несколькими способами. Например:

- Дважды щелкните по окну переходного процесса, чтобы поставить оба курсора в одну точку между максимумами сигнала. Щелкните кнопку **Maximums** : один курсор установится на ближайший максимум справа, второй курсор – на ближайший максимум слева. Этот метод может работать неправильно, если сигнал имеет локальные максимумы.

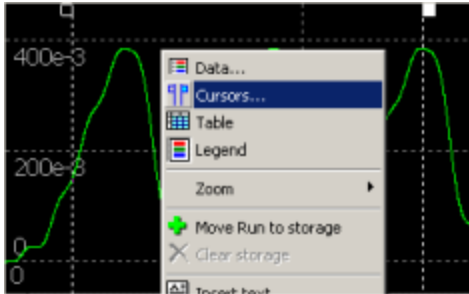
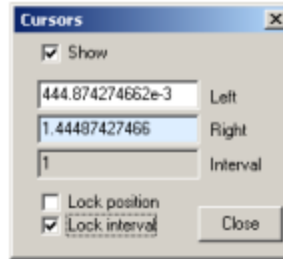
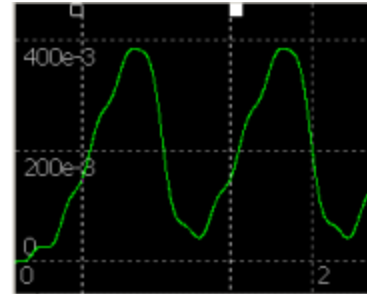


Double-click

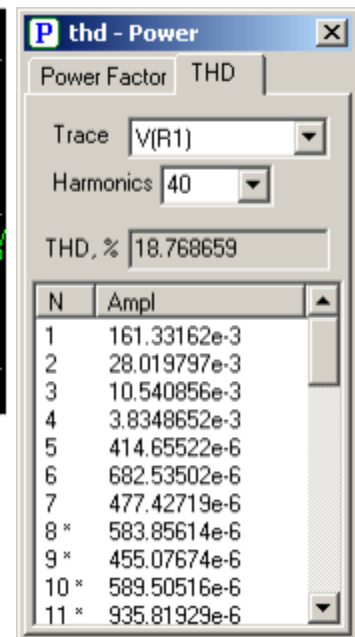
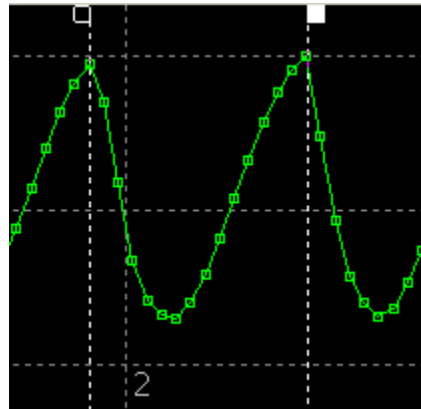


Click **Maximums** button

- Если период первой гармоники сигнала известен, щелкните правой кнопкой мыши на окне переходного процесса, выберите команду **Cursors** в контекстном меню. В диалоговом окне **Cursors** введите период сигнала (**Interval**) и установите флажок **Lock interval**. После этого интервал между курсорами будет всегда оставаться равным периоду сигнала.

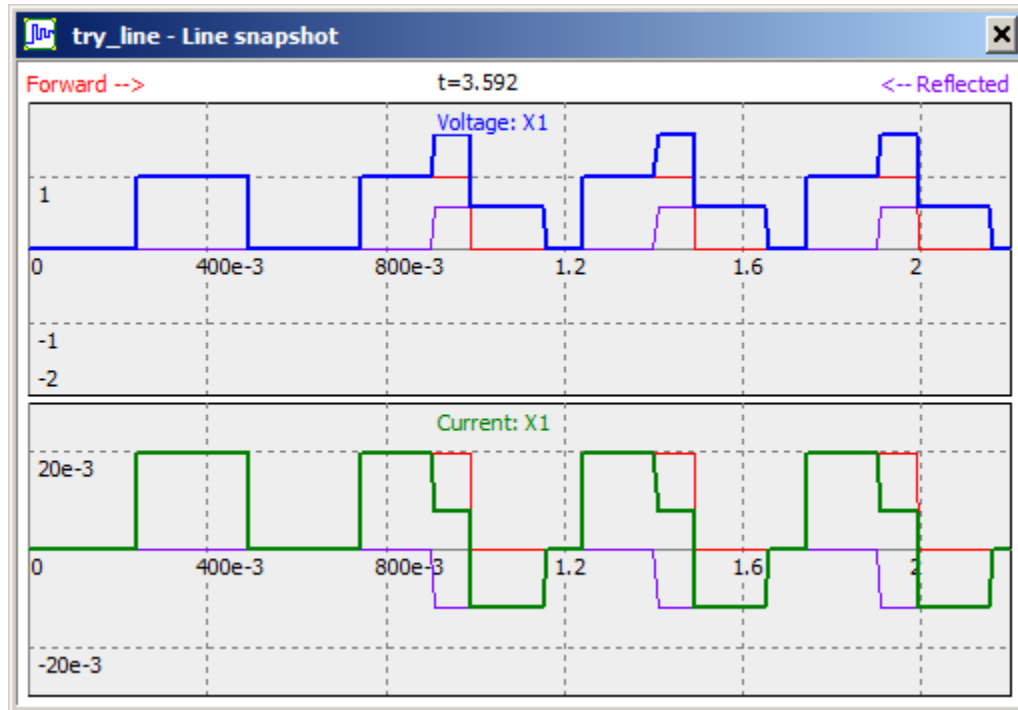
Right-click, select **Cursors**Enter **Interval**,  
select **Lock interval**

Если шаг расчета достаточно большой, высшие гармоники, начиная с некоторой, могут не удовлетворять критерию Найквиста: частота гармоники превышает половину частоты «дискретизации». Такие гармоники будут помечены звездочкой (\*) в списке амплитуд гармоник.

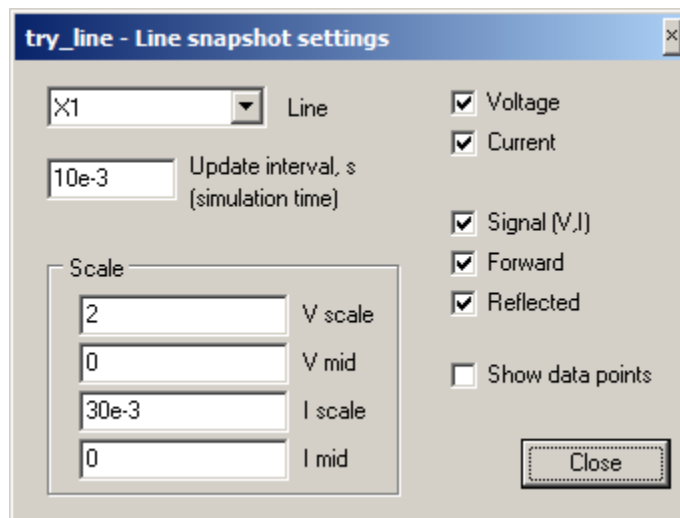


## Line snapshot

Line snapshot показывает распределение напряжения и тока внутри линии передачи (X). Он также показывает прямую и отраженную волну напряжения и тока. Все сигналы могут показываться непосредственно во время симуляции переходного процесса, обновляясь либо на каждом шаге симуляции, либо с заданным периодом обновления. Когда симуляция остановлена или приостановлена, показывается текущее (последнее) состояние линии.



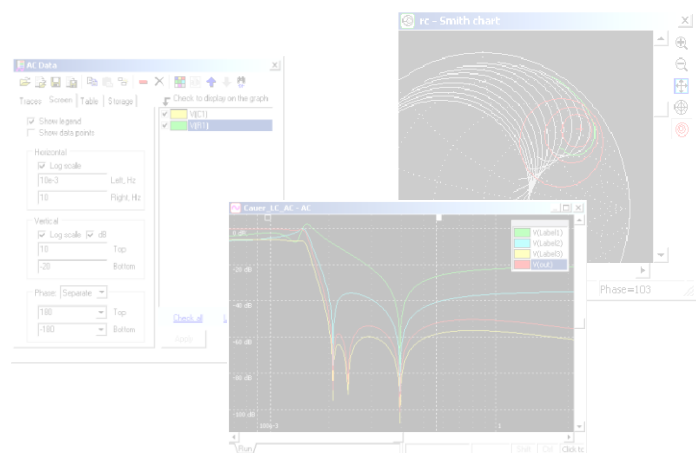
- Щелчок правой кнопкой мышки на графике вызывает контекстное меню соответствующими командами.
- Щелкните правой кнопкой мышки дважды чтобы открыть диалоговое окно **Settings**:



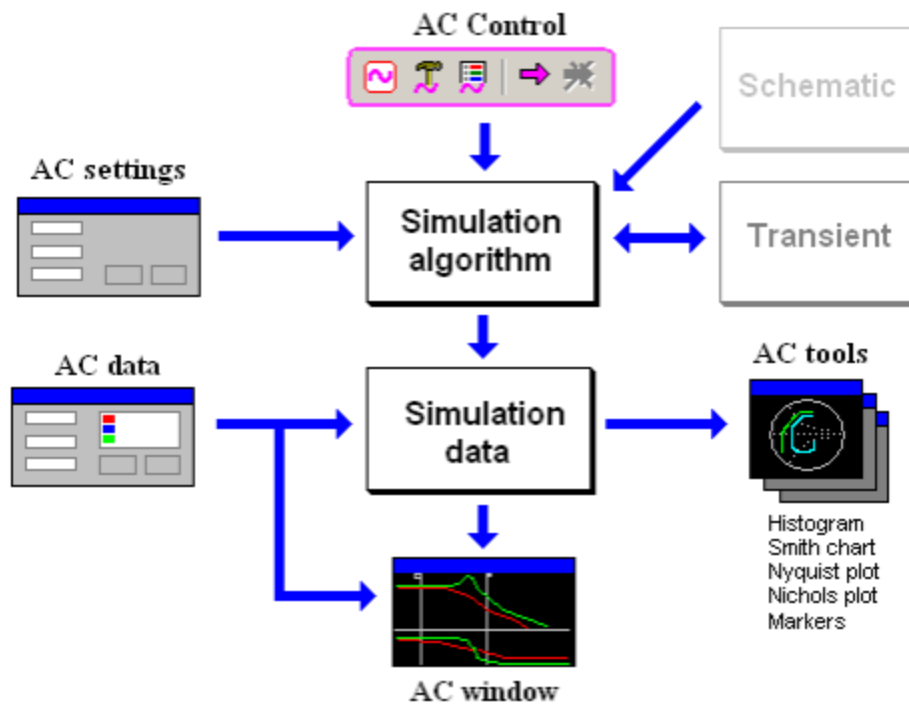


- **Line.** Выберите существующий компонент «transmission line» из выпадающего списка.
  - **Update interval, s.** Интервал обновления графиков. Если задан ноль, графики обновляются на каждом шаге симуляции.
  - **Scale.** Масштабы графиков напряжения и тока. Нажмите **Enter** чтобы применить изменения.
  - **Voltage.** Показывать график напряжения.
  - **Current.** Показывать график тока.
  - **Signal (V, I).** Показывать напряжение и ток (суперпозиция прямой и отраженной волн).
  - **Forward.** Показывать прямую волну напряжения и тока.
  - **Reflected.** Показывать отраженную волну напряжения и тока.
  - **Show data points.** Показывать точки данных маленькими квадратиками.
- Щелкните **Close** чтобы применить изменения и закрыть окно **Settings**.

# V. Частотная Характеристика (АС)



Следующая упрощенная диаграмма поясняет процесс АС симуляции:



Алгоритм симуляции конфигурируется в диалоговом окне **AC Settings** и управляется командами Основного Меню и кнопками инструментальной панели. Результаты симуляции сохраняются в **Simulation data** и одновременно отображаются на графике в **AC Window**. Диалоговое окно **AC Data** используется для конфигурирования того, какие данные симуляции будут сохранены, и как данные будут отображаться. Также данные могут использоваться в **AC Tools** (инструменты АС), которые позволяют делать различные виды анализа и представления данных АС. Анализ переходного процесса используется для метода симуляции **Sweep AC source**.

## Симуляция

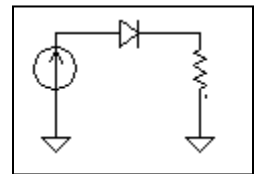
В NL5 есть два метода АС анализа: линейризация схемы (**Linearize schematic**) и «включение» источника переменного напряжения/тока (**Sweep AC source**).

### Метод линейризации схемы

Линейризация схемы — это стандартный малосигнальный анализ на переменном токе (AC analysis). В первый момент все нелинейные компоненты замещаются линейными эквивалентами в их рабочих точках. Затем сигнал заданной частоты с единичной амплитудой и нулевой фазой подается на входной узел, а сигналы других узлов находятся решением системы линейных уравнений. Процесс повторяется для заданного числа частот.

Для того чтобы линейризовать схему, должно быть известно состояние всех компонентов схемы. Это может быть сделано вручную, заданием начальных условий (IC) для всех нелинейных компонентов, диодов и управляемых ключей, или автоматическим расчетом рабочих точек на постоянном токе (установите **Calculate DC operating point** в диалоговом окне **AC Settings**). Рабочая точка на постоянном токе рассчитывается так же, как в анализе переходного процесса

Этот метод работает всегда для линейных цепей. Он также может использоваться для цепей с нелинейными компонентами, но только если эти компоненты могут быть правильно линейризованы в рабочей точке: бесконечно малая амплитуда входного сигнала (AC) не должна менять состояние компонента. Например, следующая цепь не может быть корректно проанализирована этим методом, поскольку диод будет менять свое состояние каждый раз, когда меняется полярность входного сигнала.



Метод не может использоваться для цепей переключающего типа, поскольку все переключающие устройства будут устанавливаться либо в открытое, либо в закрытое состояние, а не будут периодически переключаться, как требуется.

### Метод «включения» источника (Sweep AC source)

Данный метод позволяет рассчитать характеристику любого типа схем. Реальный синусоидальный сигнал заданной амплитуды и частоты подается на входной узел; автоматически выполняется анализ переходного процесса в заданном интервале времени, пока все сигналы не стабилизируются. И, наконец, из сигнала в других узлах извлекается амплитуда и фаза заданной частоты с помощью дискретного преобразования Фурье. Частотная характеристика вычисляется путем сравнения этой амплитуды и фазы с амплитудой и фазой входного сигнала. Процесс повторяется для заданного числа точек в заданном частотном диапазоне. Параметры, необходимые метода устанавливаются в диалоговом окне **AC Settings**.

Анализ переходного процесса, необходимый для этого метода, автоматически запускается и контролируется программой. Вы не должны задавать кривые переходного процесса: все необходимые кривые будут автоматически добавлены и удалены после окончания анализа. Единственный параметр, относящиеся к переходному процессу, который необходимо задать — это шаг вычисления (**Transient calculation step**). Он должен быть выбран из тех же соображений, как и шаг для обычного анализа переходного процесса данной схемы без «включенного» АС

источника. Во время АС анализа шаг вычисления может быть автоматически уменьшен, чтобы не превышать чем  $1/16$  периода «включенного» АС источника.

**АС source amplitude** – амплитуда АС источника (напряжение или ток) может быть постоянной, или функцией частоты источника ( $f$ ). Частотно-зависимая амплитуда может быть полезна для поддержки устойчивой работы схемы на разных частотах источника сигнала. Например, если коэффициент передачи переключающей схемы уменьшается с частотой, будет полезным иметь амплитуду источника растущей с частотой, чтобы увеличить отношение сигнал/шум.

Когда АС источник устанавливается на определенную частоту, анализ переходного процесса производится в течение временного интервала, содержащего некоторое количество периодов этой частоты. Чем больше периодов используется для вычислений, тем лучше точность АС анализа может быть достигнута. Однако это может потребовать значительного времени, особенно на низких частотах. Чтобы установить разумный баланс между точностью и временем симуляции, используется параметр **Error (ошибка)**. Фактически данный параметр лишь качественно описывает ожидаемый уровень погрешности симуляции, а не задает реальную величину ошибки. Параметр **Error** может принимать следующие значения:

**Error = 0** – на каждой частоте анализ переходного процесса будет проведен на максимальном количестве периодов этой частоты - 128. Это даст наилучшую точность анализа, но может потребовать очень много времени.

**0 < Error < 100** – количество периодов для анализа переходного процесса будет выбрано автоматически на каждой частоте между 2 и 128, основываясь на отклике схемы. Ожидается очень высокая точность симуляции при параметре равном 1 и ниже, хорошая - при значениях от 10 до 50, и приемлемая при значениях больше 50. Соответственно, большая точность потребует большего времени симуляции.

**Error = 100** - на каждой частоте анализ переходного процесса будет проведен только на 2 периодах этой частоты. Это самая быстрая опция, с возможно большей погрешностью.

Рекомендуется следующая стратегия анализа. Сначала установите **Error=100**, чтобы понять, работает ли метод в принципе, и соответствует ли примерно поведение схемы ожидаемому. Затем используйте величину **10** для анализа с хорошей точностью и приемлемой скоростью. Если требуется значительно лучшая точность, или результаты симуляции не выглядят правильными (что может быть для некоторых специфичных типов схем), используйте величину **1**.

Чтобы проводить АС анализ ключевых схем, сначала должна быть найдена «периодическая» рабочая точка (“periodic operating point”). Если выбрана опция **Find periodic operating point**, NL5 автоматически запустит вычисление переходного процесса с нулевой амплитудой АС источника до тех пор, пока периодический процесс в схеме не установится. Только после этого начнется АС анализ. Если опция **Find periodic operating point** не выбрана, пользователю рекомендуется привести схему в рабочую точку вручную, запустив анализ переходного процесса на необходимое время. Когда процесс стабилизируется, сохраните текущее состояние схемы в начальные условия (Initial Conditions) командой **Transient | Save IC** в Основном Меню. Если схема не находится в своей рабочей точке, правильные результаты АС анализа не гарантируются.

Если выбрана опция **Sweep frequency from high to low**, АС анализ будет выполняться, начиная с высокой частоты, где обычно требуется меньше времени симуляции на одну частоту. Результаты

симуляции будут видны быстрее, и пользователь может принять решение, например, о прекращении симуляции, не дожидаясь результатов для низких частот.

Опция **Show transient data** позволяет видеть данные анализа переходного процесса, а также сохранить их и после завершения АС анализа (иначе данные переходного процесса будут автоматически удалены). Данные переходного процесса могут быть полезны для нахождения оптимальных параметров АС анализа и для разбирательства с возможными проблемами.

## Данные симуляции


**Traces (кривые).** Во время симуляции NL5 хранит данные в памяти. Данные для сохранения выбираются пользователем как кривые (traces) в диалоговом окне **AC Data**. Доступно несколько типов кривых: V (напряжение), I (ток), Z (импеданс), Gamma (коэффициент отражения), VSWR (коэффициент стоячей волны) и Function (функция).

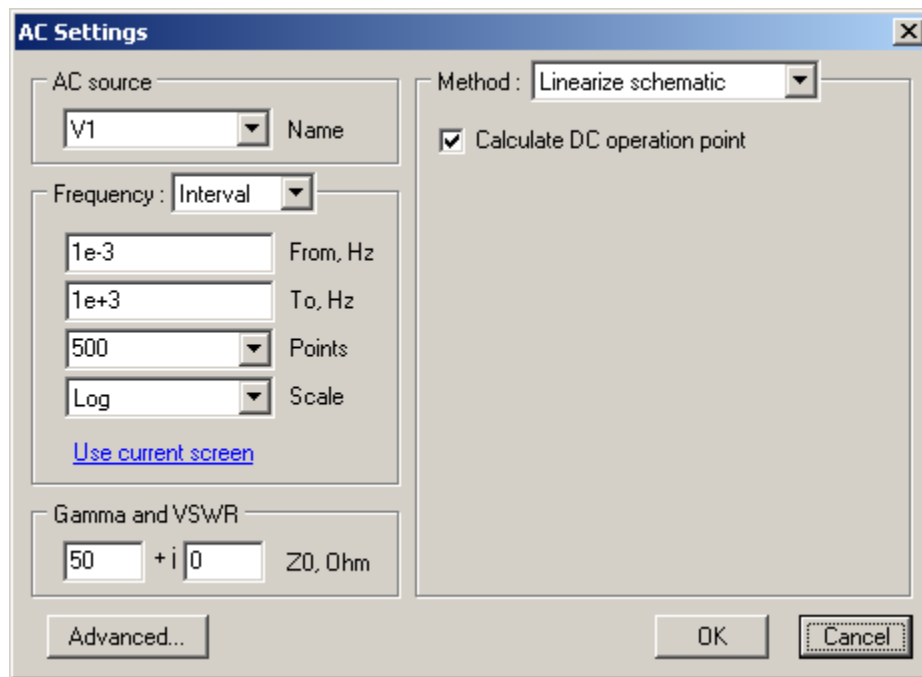
Когда начинается симуляция, все кривые автоматически очищаются, а затем начинается сохранение новых данных симуляции. Новые данные отображаются на закладке **Run** окна **AC Window**. Данные последней симуляции могут быть перемещены в **Storage** со специальной закладкой в окне **AC Window**. Сохраненные таким образом данные не очищаются автоматически и могут быть использованы для сравнения результатов разных запусков симуляции.


Кривые могут копироваться в буфер обмена, сохраняться в файлах данных «nlf» или экспортироваться в текстовый файл формата «csv». В свою очередь данные могут вставляться из буфера обмена, загружаться из файла данных «nlf» или импортироваться из текстового файла, как новый график. Такой график всегда отображается в окне **AC Window**, независимо от того, какая закладка выбрана. Он не очищается, когда запускается новая симуляция, и может использоваться в качестве опорной кривой для симуляции. Он также может быть переименован произвольным образом.

Данные симуляции хранятся только в оперативной памяти.

## Установки AC (AC Settings)

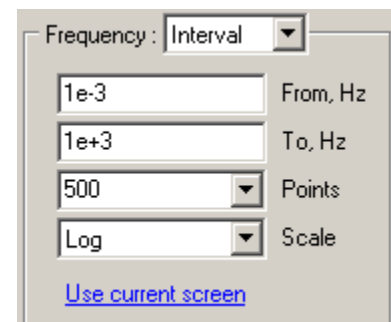
Щелкните по кнопке **AC settings**  , или выберите команду **AC | Settings** Основного Меню. Появится диалоговое окно **AC Settings**:




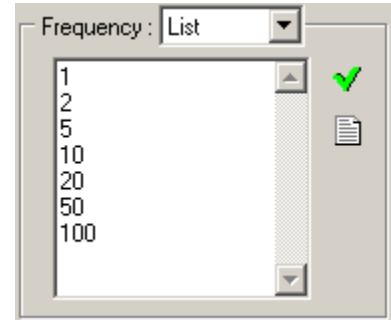
**AC Source (источник AC).** Введите имя источника переменного напряжения или выберите его из выпадающего списка. Любой источник напряжения, тока или этикетка (label) могут быть использованы в качестве AC источника. AC источник может иметь любую модель (кроме **File** и **SubCir**): нет необходимости менять модель на **Sin**, она изменится автоматически в процессе AC анализа и будет возвращена в первоначальную, когда анализ закончится. Во время AC анализа постоянное напряжение/ток AC источника будет установлено в его DC значение при  $t=0$ , а переменная составляющая, необходимая для AC анализа, будет добавлена к этому постоянному уровню. К имени компонента источника AC на схеме и в окне **Components** будет добавлен текст (AC). AC источник можно также выбрать из контекстного меню схемы и в окне **Components** кнопкой **Set AC source** .


**Frequency (частота).** Выберите метод задания частот AC анализа. Когда начнется симуляция, диапазон частот окна AC автоматически установится в заданный частотный диапазон.

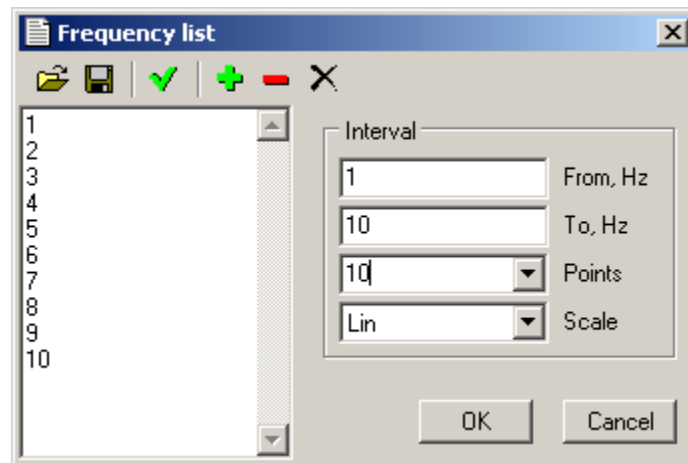
- **Interval.** Выберите диапазон частот, количество точек симуляции и вид шкалы..
  - **From, Hz.** Начальная частота.
  - **To, Hz.** Конечная частота.
  - **Points.** Количество точек симуляции.
  - **Scale.** Частотная шкала:
    - **Log.** Логарифмическая.
    - **Lin.** Линейная.









- **Use current screen.** Щелкните, чтобы использовать левую частоту, правую частоту и шкалу текущего экрана AC в качестве нового интервала симуляции. Параметры **From**, **To** и **Scale** будут установлены согласно тому, что отображено на графике AC в данный момент.
- **List (список).** Задайте список произвольных частот. Используйте этот метод, чтобы определить больше точек частоты в диапазоне, где характеристика меняется очень быстро, чтобы оптимизировать время симуляции (особенно для метода **Sweep AC Source**). Введите по одной частоте в строке, в любом порядке. Щелкните по кнопке **Sort and refresh** , чтобы проверить и упорядочить список.



Щелкните по кнопке **Edit frequency list** , чтобы редактировать список частот в диалоговом окне **Frequency List**.



Диалоговое окно позволяет ввести частоты вручную, а также выполнить следующие операции:

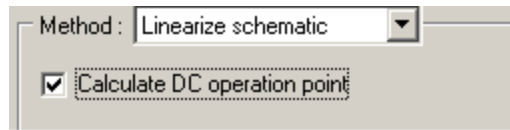
-  ○ **Open frequency list** – загрузить список из текстового файла.
-  ○ **Save frequency list** – сохранить список в текстовый файл.
-  ○ **Sort and refresh** – обновить и отсортировать список.
-  ○ **Add interval** – добавить в список частоты, заданные параметрами в окне **Interval** (**From** - от, **To** - до, **Points** - точек и **Scale** - шкала). Частоты, уже находящиеся в списке, удалены не будут.
-  ○ **Remove interval** - удалить из списка частоты, заданные параметрами в окне **Interval** (**From** - от, **To** - до).
-  ○ **Delete all** - удалить все частоты из списка.

**Gamma and VSWR** – параметры, используемые для вычисления коэффициента отражения и KСВН.

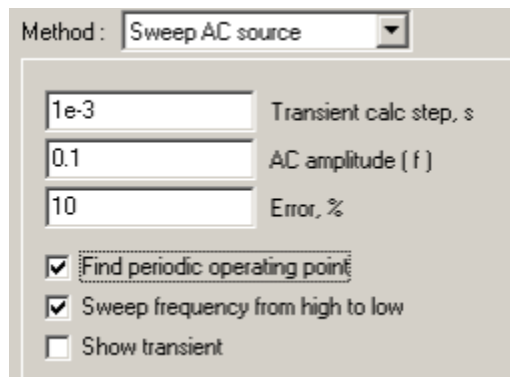
- **Z0, Ohm.** Характеристический импеданс.



- **Advanced.** Щелкните, чтобы открыть диалоговое окно **Advanced**. См. раздел «*Установки переходного процесса*» для деталей.
- **Method.** Метод АС симуляции. См. раздел «*Симуляция*» для подробностей.
  - **Linearize schematic** – метод линеаризации схемы.




- **Calculate DC operating point.** Если выбрано, перед АС анализом будет вычислена рабочая точка по постоянному току. Эта опция не требуется для линейных цепей или если начальные условия (Initial Conditions) для всех компонентов определены вручную.
- **Sweep AC source** – метод «включенного» источника.

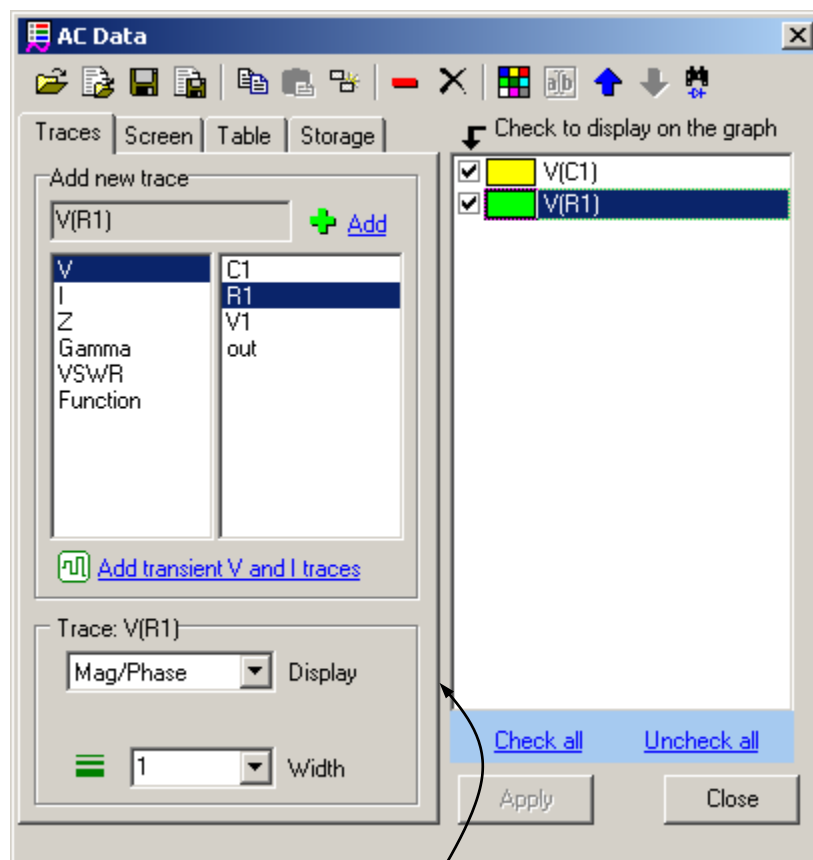


- **Transient calc step, s.** Шаг расчета переходного процесса.
- **AC amplitude ( f ).** Амплитуда источника АС. **V** для источника напряжения и этикетки, **A** для источника тока. Амплитуда может быть константой или функций частоты. Например:
  - 0.1
  - 1m\*f
  - 1000/f
- **Error, %.** Ожидаемая ошибка АС анализа. См. раздел «*Метод “включенного” источника*» для деталей.
- **Find periodic operating point.** Вычислять переходного процесс в начале АС анализа до тех пор, пока периодический процесс в схеме не установится.
- **Sweep frequency from high to low.** Выполнять АС анализа, начиная с высокой частоты.
- **Show transient.** Показывать данные переходного процесса во время АС анализа и сохранить их после окончания АС анализа.

## Данные AC (AC Data)

Щелкните по кнопке **AC data** , или выберите команду **AC | Data**. Появится окно **AC Data**. window will show up. В окне всегда показаны данные, относящиеся к активному документу (схеме): переключение на другой документ автоматически обновит данные в окне. Окно содержит инструментальную панель, список кривых (traces) и 4 страницы, используемые для следующих операций:

- **Traces (кривые):** добавляет кривые, задает индивидуальные масштаб и ширину для кривых.
- **Screen (экран):** задает масштаб, линии сетки и другие опции экрана для графика.
- **Table (таблица):** конфигурирует таблицу данных.
- **Storage («хранилище»):** обслуживает хранение данных.



Поместите мышку поверх «разделителя» , затем нажмите левую клавишу мышки и перетащите разделитель, чтобы изменить размер панелей.

**Trace list (список кривых)** показывает все текущие доступные кривые. Флажок показывает следующие свойства кривых, в зависимости от того, какая страница выбрана:

- Страницы **Trace** и **Screen** – кривая будет показана на графике.
- Страница **Table** – кривая будет показана в таблице.
- Страница **Storage** – кривая будет сохраняться в **storage**.

Щелкните по **Check all**, чтобы поставить флажок на всех кривых, или **Uncheck all**, чтобы убрать флажок

Большая часть команд панели применима только к выделенным кривым. Одна или больше кривых могут быть выбраны из списка с помощью мышки и клавиш **Ctrl** и **Shift**. Пожалуйста, заметьте, что выделенные кривые подсвечены в списке, и «выделенное» состояние кривой не зависит от состояния её флажка. На рисунке выше обе кривые имеют выставленный флажок, но только V(V1) выбрана.

Дважды щелкните по кривой, чтобы изменить ее цвет.

В этой главе описаны команды инструментальной панели, и работа со страницей **Traces**. Другие страницы описаны в главе **AC Window** (разделы **Graph**, **Data table** и **Storage**).

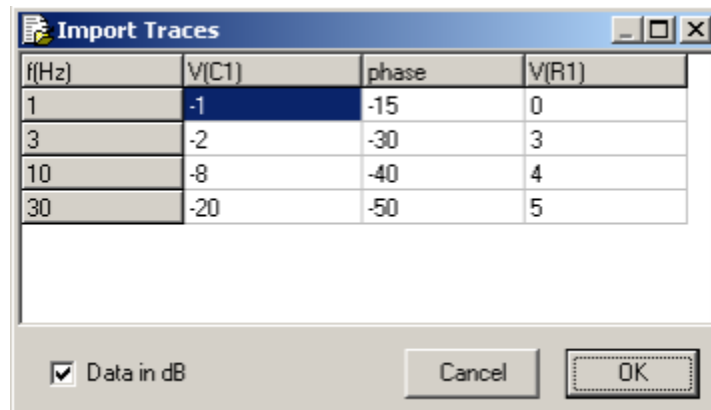
## Инструментальная панель

Команды кнопок инструментальной панели относятся ко всем или только выделенным кривым. Некоторые из этих команд также доступны через контекстные меню в окне **AC**.

- **Open file.** Загрузить кривые из файла данных типа “nlf”.
- **Import traces.** Импортировать кривые из текстового файла формата “txt” или “csv”. Формат данных файла должен быть аналогичен формату экспорта кривых. Первая строка — это строка заголовка: она может содержать любой текст в первой колонке, и имена кривых в других колонках. Первая колонка содержит частоту (в Гц), другие колонки содержат данные кривой. Данные кривой могут иметь только одну колонку на кривую или две колонки на кривую. Первая колонка кривой — это амплитуда (абсолютное значение или дБ), и она имеет имя кривой в строке заголовка. Вторая колонка кривой, если она есть, это фаза (в градусах), и колонка имеет в заголовке надпись «phase». Если вторая колонка кривой отсутствует, фаза кривой установлена в ноль. Если имя кривой состоит из символов иных, чем цифры и буквы, оно должно быть заключено в кавычки. Данные и имена могут быть разделены запятыми, пробелами или табуляцией. Например:

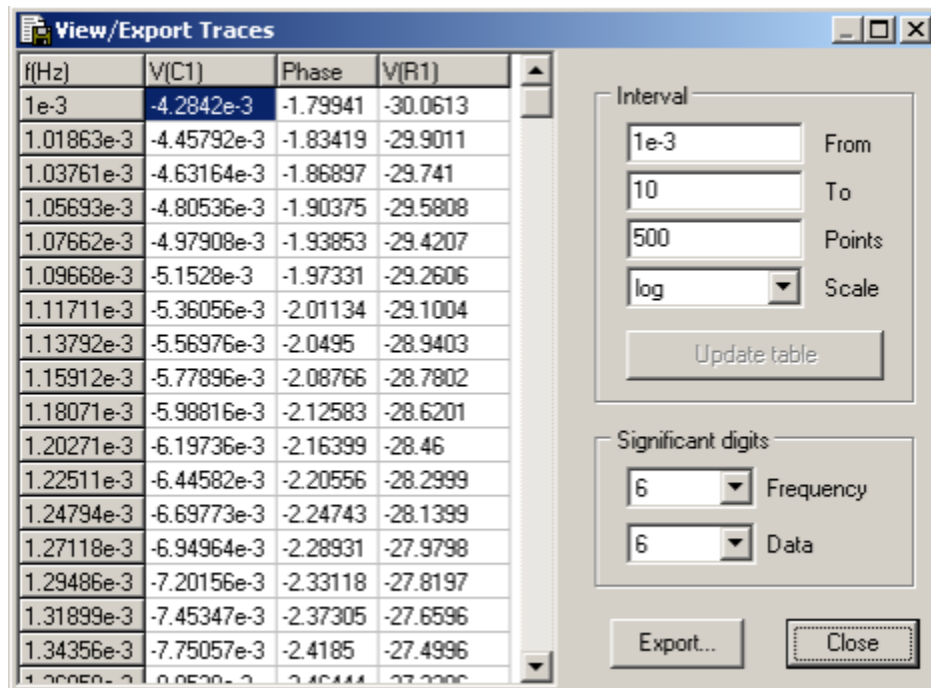
```
f (Hz) , V (C1) , phase , V (R1)
1, -1, -15, 0
3, -2, -30, 3
10, -8, -40, 4
30, -20, -50, 5
```

Когда файл загружается, его содержание отображается в диалоговом окне **Import Traces** для проверки:



Установите флажок **Data in dB**, если значения кривой в дБ. Щелкните по **OK**, чтобы принять импортированный файл. Новые кривые будут созданы и показаны на графике.








- **Save selected traces** - сохранить выделенные кривые в файл данных типа “nlf”. Только выделенные (подсвеченные) кривые из списка будут сохранены.
- **View/Export selected traces.** посмотреть и экспортировать выделенные кривые в текстовом виде (“txt” или “csv” формат). Только выделенные (подсвеченные) кривые из списка будут показаны. Появится диалоговое окно **View/Export**:

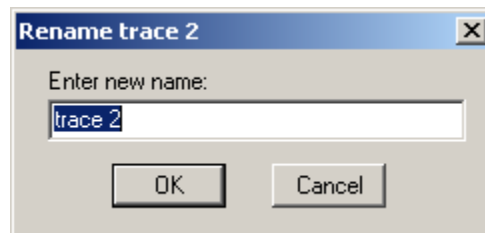


Выделенные кривые показаны как текст в таблице. Первоначально кривые показываются в частотном интервале между курсорами экрана или, если курсоры выключены, на всем видимом экране. Измените значения **From, To, Points** и **Scale** и нажмите **Enter** или щелкните по кнопке **Update table**, чтобы обновить данные в таблице. Кривые




отображаются с фиксированным временным шагом, заданным значением **Step**. Число значащих цифр для колонок времени и данных может быть задано.

Щелкните по **Export**, чтобы сохранить таблицу в текстовом файле, как значения, разделенные запятыми.

-  • **Copy selected traces** – скопировать выделенные кривые в буфер обмена. Только выделенные (подсвеченные) кривые из списка будут скопированы.
-  • **Paste traces** - вставить кривые из буфера обмена.
-  • **Duplicate selected traces** – сдублировать выделенные кривые. Эта операция эквивалентна операциям **Copy/Paste**. . Только выделенные (подсвеченные) кривые из списка будут сдублированы.
-  • **Remove selected traces** – удалить выделенные кривые. Только выделенные (подсвеченные) кривые из списка будут удалены.
-  • **Delete all traces** – удалить все кривые из списка.
-  • **Select color** – задать цвет выделенной кривой. Должна быть выделена только одна кривая. Двойной щелчок по кривой выполняет ту же операцию.
-  • **Rename trace** - переименовать выделенную кривую. Должна быть выделена только одна кривая. Переименованы могут быть только кривые, загруженные или импортированные из файла, сдублированные или вставленные из буфера обмена. Переименование кривой типа **Function** изменяет саму функцию. Появится окно **Rename**:

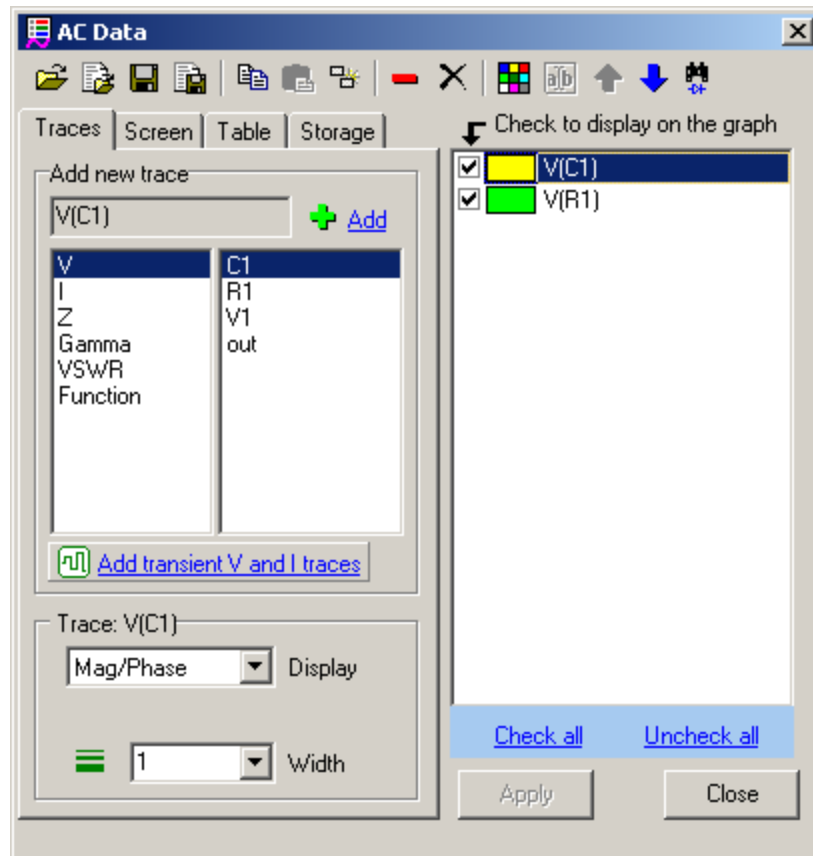


Введите новое имя кривой и щелкните по **OK**.

-  • **Move selected traces up** – передвинуть выделенные кривые вверх в списке. Эта операция меняет порядок кривых в списке, на графике и в таблице.
-  • **Move selected traces down**. передвинуть выделенные кривые вниз в списке. Эта операция меняет порядок кривых в списке, на графике и в таблице.
-  • **Find component**. Если выделенная кривая – это V, I, Z, Gamma, или VSWR, щелкните, чтобы показать компонент на схеме. Компонент будет выделен (подсвечен) и отцентрован на экране


## Traces (кривые)

Страница **Traces** окна **AC Data** используется для добавления и удаления кривых, и для установок для отдельных кривых масштаба и ширины.





**Add new trace** (добавить новую кривую). Выберите тип графика в списке слева:

- **V** – напряжение.
- **I** – ток.
- **Z** – импеданс.
- **Gamma** – коэффициент отражения.
- **VSWR** – коэффициент стоячей волны напряжения (КСВН).
- **Function** – произвольная функция.

Если выбрана кривая **V**, **I** или **P**, список справа покажет компоненты, доступные для графика этого типа: модель компонента должна поддерживать выбранный тип. Выделите компонент и щелкните по кнопке **Add** , или дважды щелкните по имени переменной, чтобы добавить новый график в список. Имя кривой состоит из буквы с последующим именем компонента в скобках:

V (R1) , I (C2)

Если выбраны кривые Z, Gamma или VSWR, щелкните по кнопке **Add**  , чтобы добавить новую кривую к списку. Кривые покажут Z, Gamma и VSWR относительно источника (AC source).

Если выбрана кривая **Function**, введите функцию в окне редактирования и щелкните по кнопке **Add**  to add new trace to the trace list. , чтобы добавить новую кривую в список. Функция может состоять из арифметических операторов и функций, параметров компонентов, текущего времени переходного процесса  $t$  и  $V$  и  $I$  на компоненте, и следующих переменных:

- $f$  – текущая частота.
- $w$  – круговая частота,  $w = 2\pi f$ .
- $s$  or  $p$  – параметр Лапласа,  $s = p = j*2\pi f$  .

Например:

```
V(C1) / I(C1)
V(X1.V1)
1 / (1+s)
V(R1) * f
```

Имя кривой — это сама функция, так что переименование кривой изменит функцию.

Графики  $V$  и  $I$  могут также добавляться из контекстного меню схемы и кнопками панели окна **Components**.

Для каждой кривой могут быть установлены следующие индивидуальные параметры:

- **Display.** Задаёт, что должно быть отображено на графике и показано в таблице данных:
  - **Mag/Phase.** Амплитуда и фаза.
  - **Mag.** Только амплитуда.
  - **Phase.** Только фаза.
  - **Re.** Действительная часть. Может быть использована для отображения  $R$  кривой  $Z$ .
  - **Im.** Мнимая часть. Может быть использована для отображения  $X$  кривой  $Z$ .
- **Width.** Ширина кривой в пикселях.

Выберите одну или более кривых в списке кривых, измените параметры и нажмите **Enter** или щелкните по кнопке **Apply**. Если выбранные кривые имеют разные значения одного и того же параметра, соответствующее поле будет оставаться пустым. Оставьте поле пустым, чтобы сохранить индивидуальные значения без изменения, или введите новое значение, чтобы применить его ко всем выделенным кривым.

Чтобы отобразить и **Re**, и **Im** части кривой, добавьте эту кривую в список дважды, затем выделите **Re** для одной кривой, а **Im** для другой.

## Симуляция

Используйте команды Основного Меню, кнопки инструментальной панели или горячие клавиши для выполнения симуляции.

- ➔ • **Start AC (AC | Start, or F9).** Запустить симуляцию. Когда AC симуляция стартует, открывается окно **AC Window** и частотный диапазон экрана и шкала частоты устанавливаются в значения, заданные в диалоговом окне **AC Settings**. Во время симуляции результаты немедленно отображаются в **AC Window**, вместе с индикатором прогресса симуляции.
- ✖ • **Stop AC.** Прекратить симуляцию. Во время симуляции это единственная доступная кнопка.
- 📄 • **AC Log (AC | Log).** Информация журнала регистрации, показываемая в диалоговом окне, может использоваться для поиска проблем. Последняя запись сохраняется в файле схемы. При отправке файла схемы в службу поддержки (Customer Service) для помощи, пожалуйста, сохраняйте схему после симуляции, чтобы последняя запись журнала регистрации была включена в файл. Щелкните по кнопке **Copy to clipboard**, чтобы скопировать эту текстовую запись в буфер обмена.
- 🔧 • **Sweep (AC | Sweep)** Позволяет запускать серии AC анализов, когда параметр компонента или переменная меняется в заданном диапазоне, и сохранять данные анализа. Режим **Sweep** конфигурируется на странице **Sweep** окна **Tools**.






## Окно AC (AC window)

Типичный вид окна AC и его основных компонент показан ниже:

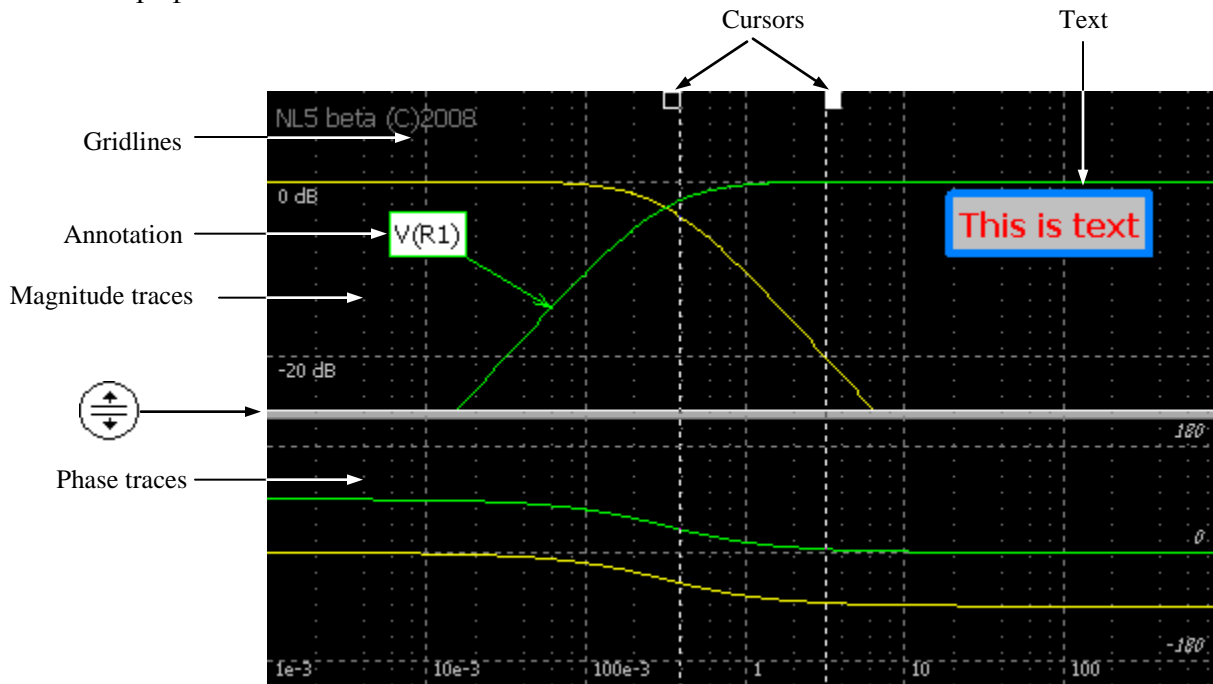
The screenshot shows the AC window interface with various components labeled:

- Right-click**: Points to context menus for the graph area and the legend.
- Graph area**: The main plot area showing magnitude and phase curves.
- Legend**: A legend box showing traces for V(C1) and V(R1).
- Data table**: A table below the plot showing cursor data for V(C1) and V(R1).
- Data selection**: Points to the 'Run' button.
- Simulation progress**: Points to the progress bar.
- Shift/Ctrl indicators**: Points to the 'Shift' and 'Ctrl' buttons.
- Status bar**: Points to the 'Click to move cursor' button.

|         | left          | right      | delta      | min        | max           | pp        | slope   |
|---------|---------------|------------|------------|------------|---------------|-----------|---------|
| Cursors | 12.6624e-3    | 281.024e-3 | 268.361e-3 |            |               |           |         |
| V(C1)   | -6.8692625e-3 | -2.5032154 | -2.4963461 | -2.5032154 | -6.8692625e-3 | 2.4963461 | -1.854E |
|         | -2.2782074    | -41.44036  | -39.162152 | -41.44036  | -2.2782074    | 39.162152 |         |
| V(R1)   | 28.013587     | 3.585363   | 24.428219  | 28.013587  | 3.585363      | 24.428219 | 18.145E |

- **Graph** area (область графика), содержит кривые с аннотациями, курсорами и текстом.
- **Legend** window (условные обозначения), содержит список кривых, показанных на графике. Щелкните по серому заголовку панели окна условных обозначений , чтобы перетащить окно.
- **Data table**, таблица данных, содержит информацию о курсоре/экране и расчетные данные кривой.
- **Data selection**, выбор данных, содержит закладки последних симуляций и сохраненных данных (**storage**). Щелкните по закладке, чтобы выбрать **Run** (результаты симуляции) или сохраненные данные.
- **Simulation progress**, прогресс симуляции.
- Индикатор **Shift/Ctrl** подсвечен, когда клавиши Shift и/или Ctrl нажаты.
- **Status bar** показывает подсказку, относящуюся к текущей позиции указателя мышки и состоянию **Shift/Ctrl**.
- Поместите указатель мышки поверх «разделителя» , затем нажмите левую клавишу мышки и перетащите разделитель, чтобы изменить размер области Data selection (выбор данных).
- Поместите указатель мышки поверх «разделителя» , затем нажмите левую клавишу мышки и перетащите, чтобы изменить размер таблицы данных.
- Щелкните правой клавишей мышки по графику, условным обозначениям, таблице данных или области выбора данных, чтобы увидеть контекстное меню с соответствующими командами.
- Общие свойства окна AC, такие как цвета, шрифты и некоторые другие опции, могут быть настроены на страницах **Graphs**, **Table**, **Annotation** и **Text** диалогового окна **Preferences**. Свойства, присущие документу (схеме), могут также устанавливаться в окне **AC Data**.




Область графика и ее компоненты показаны ниже:



## График

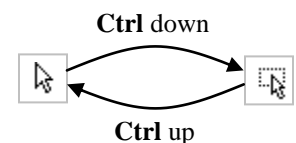
Навигация по графику может выполняться командами, доступными в контекстных меню окна переходного процесса, кнопками инструментальной панели переходного процесса, горячими клавишами, клавишами клавиатуры и мышкой. Очень часто некоторые операции могут выполняться разными способами. Так, например, масштабирование графика, *zoom in/out*, может быть сделано либо только с использованием клавиш клавиатуры, либо только мышкой, либо клавиатурой и мышкой вместе. Пользователя может сам выбрать наиболее эффективный и удобный для него способ.

Есть три режима операций с графиком:

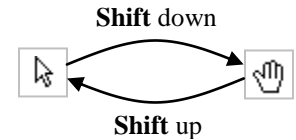
-  • **Cursors**. Перемещение курсоров.
-  • **Zoom**. Масштабирование.
-  • **Scrolling**. Прокрутка.

Режим может выбираться щелчком кнопки на панели переходного процесса. Также есть быстрые способы переключения из режима *Cursors* в *Zoom* и *Scrolling*:

- Нажмите и удержите клавишу **Ctrl**, щелкните и перетащите мышкой, чтобы масштабировать график. Отпустите клавишу **Ctrl**, чтобы вернуться в режим *Cursors*:



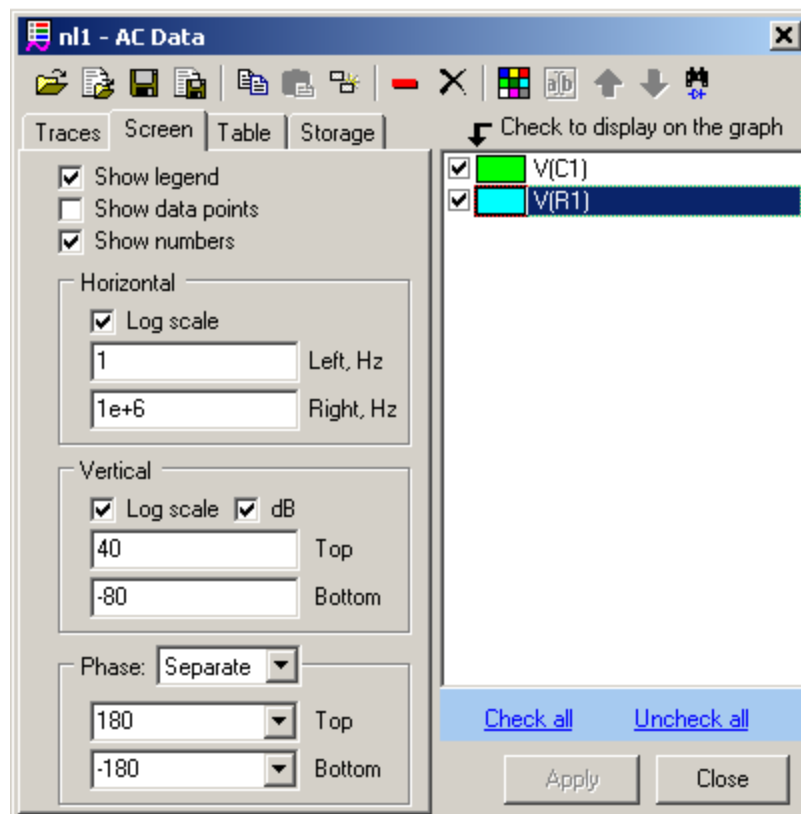
- Нажмите и удержите клавишу **Shift**, щелкните и перетащите мышкой чтобы прокрутить график. Отпустите клавишу **Shift**, чтобы вернуться в режим *Cursors*:




Кривые появляются на графике со своим индивидуальным масштабом, шириной и цветом, определенными на странице **Traces** окна **AC Data**. Все кривые имеют один и тот же вертикальный и горизонтальный масштаб.

Плотность линий сетки выбирается автоматически так, чтобы последняя значащая цифра шага была 1, 2 или 5 и дистанция между линиями сетки приблизительно равна значению, заданному на странице **Graphs** диалогового окна **Preferences**, как **Gridlines interval** в пикселях. Для логарифмической шкалы линии сетки автоматически подстраиваются, чтобы обеспечить наилучший вид графика.

Масштабы, линии сетки и некоторые другие опции графика можно изменить на странице **Screen** окна **AC Data**:



- **Show legend.** Выберите, чтобы показать окно условных обозначений. Также используйте кнопку **Legend**  на инструментальной панели или в контекстном меню.

- **Show data points.** Выберите, чтобы маркировать точки рассчитанных данных по всей кривой в виде маленьких квадратиков. Эта опция значительно замедляет вывод графиков, но может быть полезна при отладке и выборе шага вычислений.
- **Show numbers.** Выберите, чтобы выводить числа на шкалах.

**Horizontal.** Устанавливает горизонтальную шкалу и линии сетки.

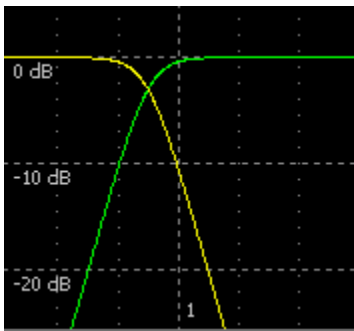
- **Log scale.** Выберите для логарифмической шкалы частоты.
- **Left.** Частота с левого края экрана.
- **Right.** Частота с правого края экрана.

**Vertical.** Установка вертикальную шкалу и линии сетки.

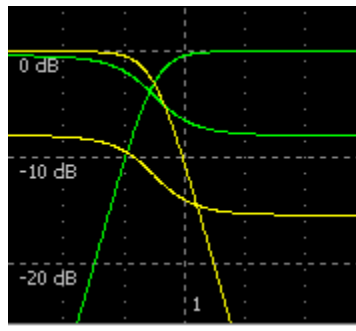
- **Log scale.** Выберите для логарифмической шкалы амплитуды.
- **dB.** Выберите, чтобы отображать вертикальную шкалу в децибелах.
- **Top.** Амплитуда в верхней части экрана.
- **Bottom.** Амплитуда в нижней части экрана.

**Phase.** Выбор режима отображения фазы:

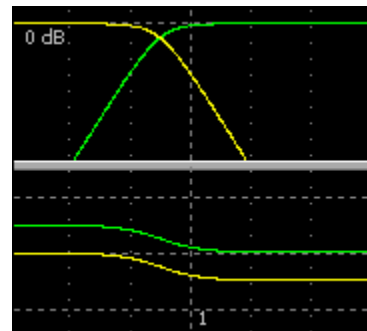
- **Off.** Не показывать фазу.
- **On.** Показывать амплитуду и фазу в той же области графика.
- **Separate.** Показывать амплитуду и фазу в отдельных областях графика.



Phase Off



Phase On



Separate

Нажмите **Tab** в окне **AC**, чтобы переключить режим отображения фазы.

- **Top.** Фаза в верхней части экрана.
- **Bottom.** Фаза в нижней части экрана.

## Legend (условные обозначения)

Окно **Legend** содержит список кривых, показанных на графике.


- Чтобы показать/скрыть условные обозначения, щелкните на



инструментальном меню по кнопке **Legend** , или используйте команду контекстного меню, или используйте флажок **Show legend** на странице **Screen** окна **AC Data**.

- Щелкните по строчке кривой, чтобы выбрать ее. Выделенная кривая будет показана поверх всех кривых.
- Дважды щелкните по строчке кривой, чтобы выделить ее и открыть окно **AC Data**.
- Щелкните правой клавишей мышки по строчке кривой, чтобы выделить ее и открыть контекстное меню. Меню будет содержать некоторые общие команды и команды, относящиеся к выбранной кривой.
- Щелкните на сером заголовке окна и перетащите окно.
- Размер шрифта условных обозначений и ширина окна могут быть выбраны на странице **Legend** диалогового окна **Preferences**.


## Курсыры

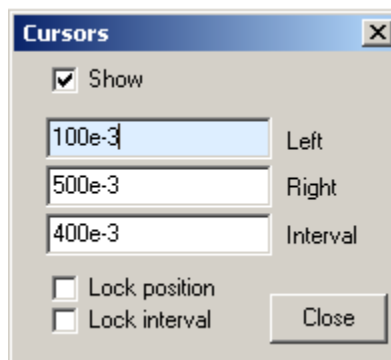
Курсыры используются в основном для выделения временного интервала на графике для расчета таблицы данных (**Data table**). Выделенный (активный) курсор показан с закрашенным квадратиком сверху. Чтобы показать/скрыть (show/hide) курсыры, щелкните по кнопке инструментальной панели **Show/hide cursors** .



Выберите режим **Cursors** (  ), чтобы перемещать курсор на графике.

- Дважды щелкните по графику, чтобы установить оба курсора в одну точку. Этим также активизируются курсыры, если они были выключены.
- Щелкните по графику, чтобы переместить ближайший курсор в эту точку.
- Щелкните и переместитесь, чтобы выделить и передвинуть курсор.







Чтобы поместить курсор в заданное место, и для других опций, щелкните правой клавишей мышки по графику и выберите команду **Cursors**  из контекстного меню. Появится диалоговое окно **Cursors**:




- **Show**. Установите флажок, чтобы показывать курсыры
- **Left, Right, Interval**. Введите новое положение курсора или расстояние между курсорами, нажмите **Enter**, чтобы применить, или **Esc**, чтобы отменить. Если изменен интервал, изменится положение активного курсора.

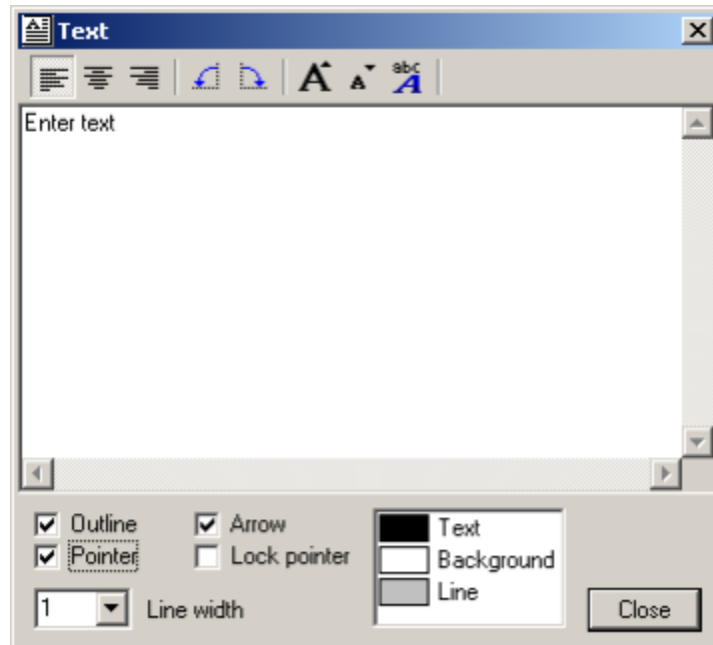
- **Lock position.** Закрепляет курсоры в текущей позиции, так что курсоры не могут перемещаться.
- **Lock interval.** Сохраняет текущий интервал между курсорами. Если один курсор будет передвинут, второй автоматически последует за ним, сохраняя заданный интервал.

Следующие кнопки инструментальной панели можно использовать для перемещений курсоров:

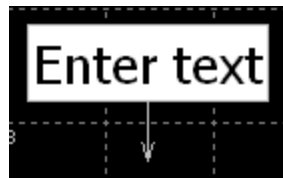
-  • **Right maximum.** Переместить активный курсор к ближайшему справа максимуму выделенной кривой.
-  • **Left maximum.** Переместить активный курсор к ближайшему слева максимуму выделенной кривой.
-  • **Right minimum.** Переместить активный курсор к ближайшему справа минимуму выделенной кривой.
-  • **Left minimum.** Переместить активный курсор к ближайшему слева минимуму выделенной кривой.
-  • **Right unity gain.** Переместить активный курсор к ближайшей справа точке выделенной кривой с единичной амплитудой.
-  • **Left unity gain.** Переместить активный курсор к ближайшей слева точке выделенной кривой с единичной амплитудой.

## Text.

Чтобы добавить текст на окно графиков, щелкните правой клавишей мышки по графику и выберите команду **Insert Text**  из контекстного меню. Появится диалоговое окно **Text**:






Введите текст в окно. Одновременно текст появится на окне графиков:





Текст можно форматировать, используя кнопки инструментальной панели:

**Alignment.** Устанавливает выравнивание для многострочного текста.

-  • **Align left.** Выравнивание по левому краю.
-  • **Center.** Выравнивание по центру.
-  • **Align right.** Выравнивание по правому краю.

**Orientation.** Изменить ориентацию текста.

-  • **Rotate left.** Повернуть влево.
-  • **Rotate right.** Повернуть вправо.

**Font.** Изменить размер шрифта или выбрать шрифт.

-  • **Larger font.** Увеличить размер шрифта.

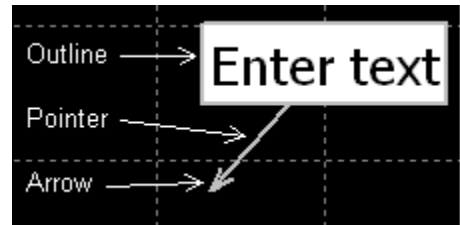





- **Smaller font.** Уменьшить размер шрифта.
- **Select font.** Выбрать шрифт.


**Outline and pointer options.** Опции указателя и рамки.

- **Outline.** Рисовать прямоугольную рамку.
- **Pointer.** Рисовать линию указателя из текста в заданную точку.
- **Arrow.** Рисовать линию указателя со стрелкой.
- **Lock pointer.** Закрепить конец указателя: конец указателя не будет перемещаться, даже если текст перемещается.
- **Line width.** Задаёт ширину линии для рамки и указателя.
- **Color.** Двойной щелчок по пункту в списке для изменения цвета.





Если график масштабировался или прокручивался, текст остается на том же месте, «заякоренный» в левом верхнем углу окна графика. Чтобы переместить текст, щелкните по тексту и перетащите его. Если указатель закреплен, переместится только текст. Чтобы переместить только указатель, щелкните по концу указателя и перетащите его.

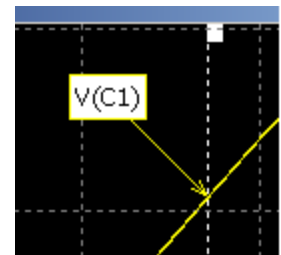
Чтобы отредактировать текст, дважды щелкните по тексту или щелкните правой клавишей мышки по тексту и выберите команду **Edit text**  из контекстного меню. Появится такое же диалоговое окно **Text**.


Чтобы удалить текст, щелкните правой клавишей мышки по тексту и выберите команду **Delete text**  из контекстного меню.

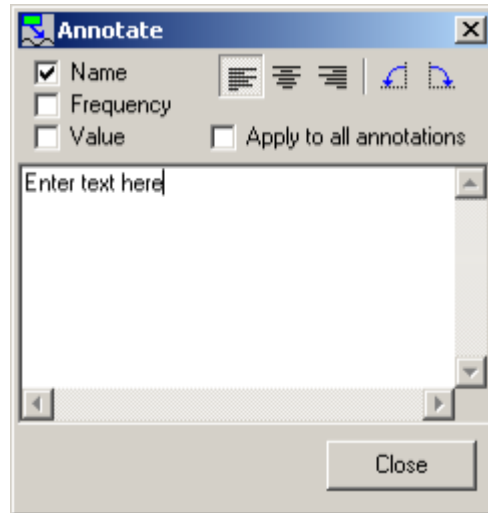
## Аннотация

Аннотации — это текст с указателем, который всегда указывает на ту же точку данных кривой, даже когда график масштабируется или прокручивается. Аннотации принадлежат кривой, так что если кривая удаляется, все аннотации к ней тоже удаляются. Аннотации также удаляются, если данные кривой очищаются. Например, если аннотация добавлена к графику симуляции (**Run**), и запускается новая симуляция, аннотация исчезнет, поскольку данные графика очищаются при новом запуске симуляции.

Чтобы добавить аннотацию, установите активный курсор на точку частоты, где нужна аннотация, щелкните правой клавишей мышки по графику, выберите **Annotate**, затем выберите команду **Selected trace**  или **All traces** . Эти же кнопки доступны на инструментальной панели АС окна. Аннотация будет добавлена только в том случае, когда данные кривой существуют для частоты, на которой находится курсор. Если курсоры выключены, аннотация будет добавлена приблизительно в месте 1/3 экрана.






Шрифт, цвета, количество значащих цифр и некоторые другие свойства аннотации могут быть заданы на странице **Annotation** диалога **Properties**. Чтобы изменить текст аннотации и какие-то свойства аннотации, дважды щелкните или щелкните правой клавишей мышки по ней, выберите команду **Edit annotation**  из контекстного меню, внесите изменения в появившемся диалоговом окне **Annotate**.





Введите текст в окне. Текст сразу отображается в аннотации. Доступны следующие опции и команды форматирования:

- **Name.** Отображает имя графика в тексте аннотации.
- **Frequency.** Отображает частоту в тексте аннотации.
- **Value.** Отображает значение амплитуды и фазы (если кривая фазы выведена на графике) в тексте аннотации

**Alignment.** Задаёт выравнивание многострочного текста.



-  • **Align left.** Выравнивание по левому краю.
-  • **Center.** Выравнивание по центру.
-  • **Align right.** Выравнивание по правому краю.

**Orientation.** Изменить ориентацию текста.

-  • **Rotate left.** Повернуть влево.
-  • **Rotate right.** Повернуть вправо.

- **Apply to all annotations.** Выберите, чтобы применить данные установки для всех аннотаций графика.

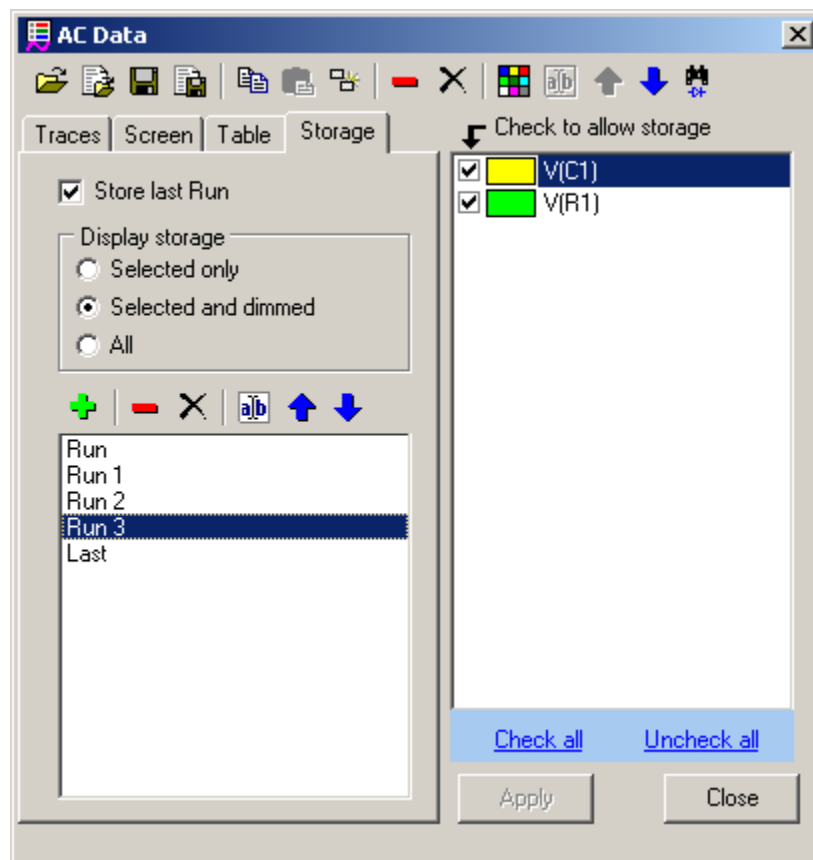
Чтобы переместить текст аннотации с сохранением привязки указателя к той же точке графика, щелкните по тексту аннотации и перетащите его. Чтобы переместить указатель, щелкните по концу указателя и перетащите его. Указатель изменит частоту, все еще следуя амплитуде графика. Текст аннотации будет перемещен с указателем.


Чтобы удалить аннотацию, щелкните правой клавишей мышки по аннотации и выберите команду **Delete annotation**  из контекстного меню. Чтобы удалить все аннотации, щелкните правой клавишей мышки по графику, выберите **Annotate**, затем выберите команду **Delete all** .

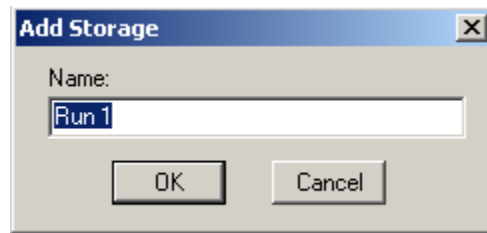
## Storage (накопитель данных)

Результаты последнего запуска симуляции всегда показаны на закладке **Run** окна **AC**. Данные последнего запуска могут быть перемещены в **storage** (накопитель данных), так что их можно будет сравнивать с другими запусками симуляции. Каждое сохранение данных имеет закладку в области **Data selection** (выбор данных). Сохраненные данные могут выбираться щелчком по закладке. Данные принадлежат кривой, так что, если кривая удалена, сохраненные данные будут также удалены.

Для доступа к командам, относящимся к сохранению данных, щелкните правой клавишей мышки по графику или области **Data selection**, затем выберите команду из контекстного меню. Список доступных сохраненных данных и команды, относящиеся к хранению и отображению данных, можно найти на странице **Storage** окна **AC Data**:

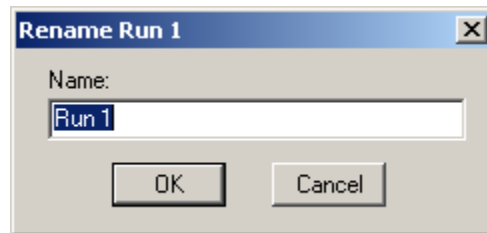


-  **Move Run to storage.** Перенести последние данные симуляции в **storage**. Откроется окно **Add Storage**:



Введите новое имя или оставьте предложенное по умолчанию и щелкните по **OK**. Будет создана новая закладка с заданным именем в области **Data selection** окна **AC**. Заметьте, что имена **Run** и **Last** являются зарезервированными и не должны использоваться.

- **Remove** – удалить выделенные данные из **storage**. Данные последней симуляции **Run** также могут быть удалены.
- ✕ • **Clear storage** – удалить все данные из **storage**.
- ajb • **Rename** – переименовать выделенные данные. Откроется окно **Rename**:



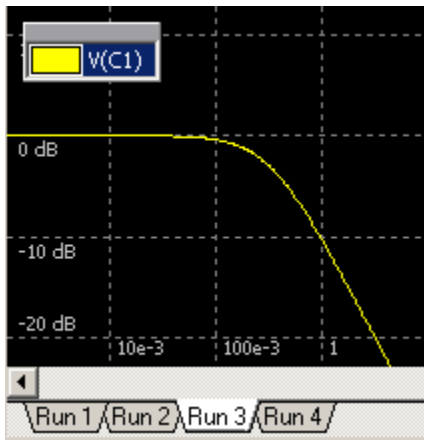
Введите новое имя и щелкните по **OK**. Заметьте, что имена **Run** и **Last** являются зарезервированными и не должны использоваться.

- ↑ • **Move selected up**. Передвинуть выделенные данные вверх (влево в области **Data Selection**)
- ↓ • **Move selected down**. Передвинуть выделенные данные вниз (вправо в области **Data Selection**)
- **Store last Run. Run (сохранить последние данные)**. Если выбрана эта опция, то в момент запуска новой симуляции предыдущие (последние) данные будут перемещены в специальную закладку **Last** («последний»), стерев при этом находящиеся там данные. Таким образом, закладка **Last** всегда содержит предыдущие данные, которые можно использовать для сравнения с данными последней симуляции **Run**.

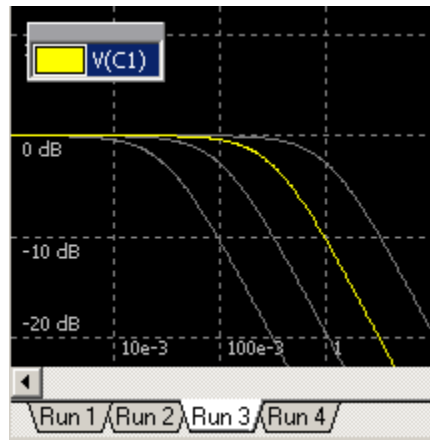
#### Display storage (отображать storage)

- **Selected only**. Только выделенные данные отображаются на графике.
- **Selected and dimmed**. Выделенные данные отображаются с нормальным цветом кривой, а другие данные отображаются с пониженной цветностью, заданной на странице **Graphs** окна **Preferences**.
- **All**. Все данные отображаются с нормальными цветом кривых.

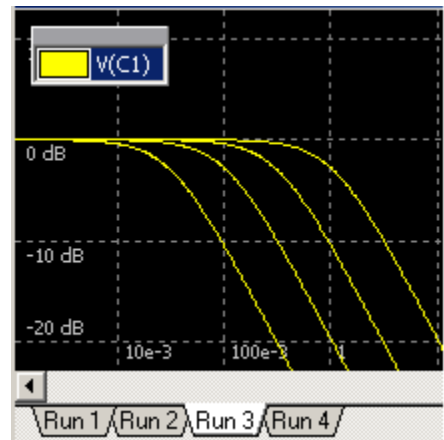
Например:



Selected only



Selected and dimmed



All

Когда страница **Storage** выбрана, флажки в списке кривых задают те кривые, для которых разрешено сохранение данных.

## Data table (Таблица данных)


Таблица данных показывает позиции курсоров, значения кривых и некоторые характеристики кривых, рассчитанных между курсорами, такие как: значение, максимум, минимум, и т.д. Если курсоры выключены, таблица показывает данные левого и правого краев экрана и значения, рассчитанные между левым и правым краями экрана:

|         | left       | right      | d  |
|---------|------------|------------|----|
| Cursors | 100.42e-3  | 1.0138     | 9  |
| V(C1)   | -10.395306 | -30.066285 | -1 |
|         | -72.411132 | -88.201624 |    |

Cursors enabled, active cursor is highlighted

|        | left        | right | d  |
|--------|-------------|-------|----|
| Screen | 10e-3       | 10e+3 | 9. |
| V(C1)  | -408.852e-3 |       |    |
|        | -17.441312  |       |    |

Cursors disabled, screen is used

- Чтобы показать/спрятать таблицу, щелкните по кнопке инструментальной панели **Table**  или щелкните правой клавишей мышки по графику и выберите команду **Table** из контекстного меню.
- Щелкните по строке кривой в таблице для выбора кривой. Выделенная кривая будет показана поверх всех других.
- Дважды щелкните по таблице, чтобы открыть окно **AC Data**.
- Щелкните правой клавишей мышки, чтобы открыть контекстное меню. Меню будет содержать некоторые общие команды и команды, относящиеся к выделенной кривой.

- Цвета, шрифты и количество значащих цифр, используемые в таблице, могут задаваться на странице **Table** диалогового окна **Preferences**.

Таблица может отображаться внизу окна **AC** или как отдельное окно: щелкните правой клавишей мышки по таблице и выберите команду **Separate window**:



Table in the AC Window

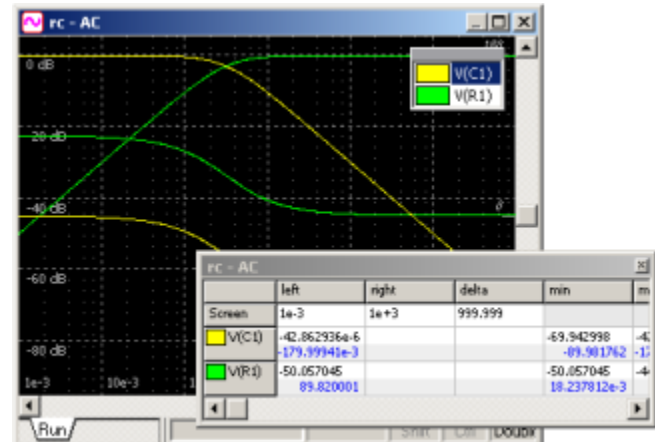
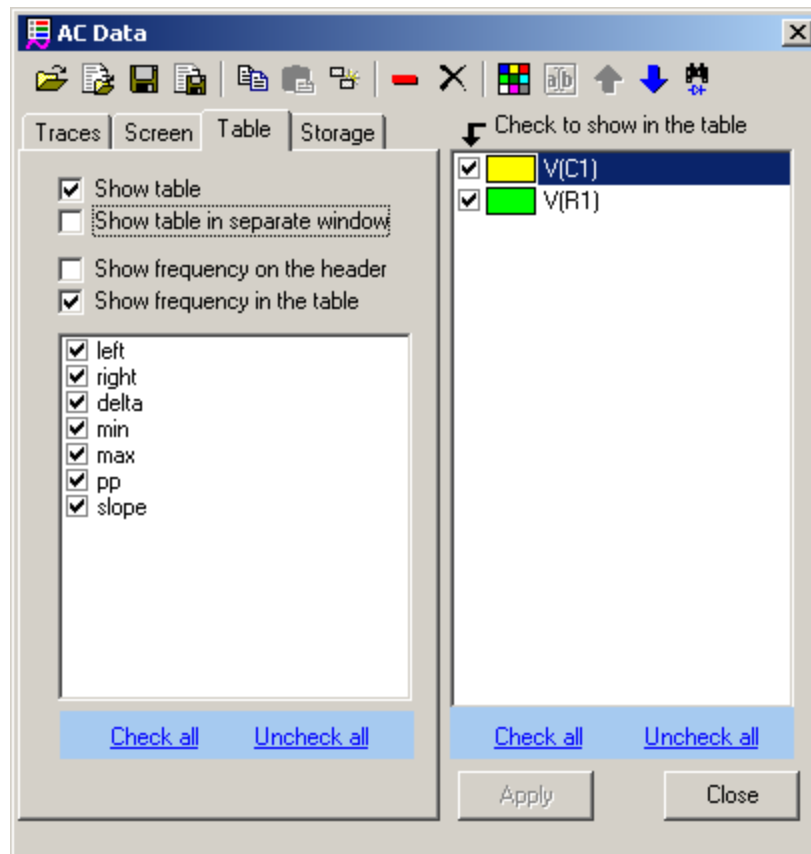


Table in separate window

Значения, показанные в таблице, также как и другие опции таблицы, могут выбираться на странице **Table** окна **AC Data**:



- **Show table.** Выберите, чтобы отображать таблицу.
- **Show table in separate window.** Если выбрано, таблица будет отображена в отдельном окне..

- **Show frequency on the header.** Если выбрано, позиция курсоров будет показана в заголовке строки, в колонках **left**, **right** и **delta**.

|         |                             |                         |                          |
|---------|-----------------------------|-------------------------|--------------------------|
| Cursors | 12.6624e-3                  | 281.024e-3              | 268.361e-3               |
| V(C1)   | -6.8692625e-3<br>-2.2782074 | -2.5032154<br>-41.44036 | -2.4963461<br>-39.162152 |
| V(R1)   | -28.013582<br>87.721793     | -3.585363<br>48.55964   | 24.428219<br>-39.162152  |

- **Show frequency in the table.** Если выбрано, позиции курсоров будут показаны в отдельной строке.



|         | left                        | right                   | delta                    |
|---------|-----------------------------|-------------------------|--------------------------|
| Cursors | 12.6624e-3                  | 281.024e-3              | 268.361e-3               |
| V(C1)   | -6.8692625e-3<br>-2.2782074 | -2.5032154<br>-41.44036 | -2.4963461<br>-39.162152 |
| V(R1)   | -28.013582<br>87.721793     | -3.585363<br>48.55964   | 24.428219<br>-39.162152  |


- **Table values.** Выберите значения для отображения в таблице:
  - **left** – значение кривой под левым курсором.
  - **right** – значение кривой под правым курсором.
  - **delta** – правое значение минус левое.
  - **min** – минимум кривой между курсорами.
  - **max** – максимум кривой между курсорами.
  - **pp** – значение кривой от пика до пика между курсорами.
  - **slope** – наклон амплитуды кривой между курсорами, в дБ/декаду. Если оба курсора находятся на одной частоте, наклон вычисляется как производная амплитуды по частоте в этой точке. Иначе наклон вычисляется как  $(\text{Mag}_{\text{right}} - \text{Mag}_{\text{left}}) / (f_{\text{right}} - f_{\text{left}})$ .

Когда страница **Table** выбрана, флажки в списке кривых задают те кривые, которые будут показаны в таблице.

## Прокрутка и масштабирование

Чтобы прокрутить график, используйте любой из следующих методов:



- Переместите указатель мышки к левому или правому краю графика. Указатель мышки примет вид «большой стрелки». Щелкните или удержите левую клавишу мышки, чтобы прокрутить график.
- В режиме **Cursors** : нажмите и держите клавишу **Shift**, затем щелкните и перетащите график.
- В режиме **Scrolling** : щелкните и перетащите график.
- Нажмите клавишу **Ctrl** и поверните колесико мышки, чтобы прокрутить горизонтально.
- Нажмите клавишу **Shift** и поверните колесико мышки, чтобы прокрутить вертикально.
- Нажмите клавиши **Right** и **Left** (курсорная панель клавиатуры).

- В режиме **Zoom**  : дважды щелкните по графику, чтобы отцентровать эту точку.

Чтобы масштабировать график, используйте любой из следующих методов:

- Поверните колесико мышки, чтобы масштабировать горизонтально.
- Нажмите клавиши **Ctrl** и **Shift** и поверните колесико мышки, чтобы масштабировать вертикально.
- Щелкните кнопку инструментальной панели, или используйте горячие клавиши, или щелкните правой клавишей мышки по графику, выберите **Zoom**, затем выберите команду:
  - ↔ **Horizontal Zoom-in (Ctrl-PgUp)** – горизонтальное увеличение.
  - ⇠ **Horizontal Zoom-out (Ctrl-PgDn)** – горизонтальное уменьшение.
  - ↔ **Fit the screen horizontal (Ctrl-Home)** – автомасштабировать горизонтально.
  - ⌘ **Fit cursors to screen** – привести курсоры к экрану.
  - ↑ **Vertical Zoom-in (PgUp)** - вертикальное увеличение.
  - ↓ **Vertical Zoom-out (PgDn)** - вертикальное уменьшение.
  - ↕ **Fit the screen vertical (Home)** – автомасштабировать вертикально.
  - ⌘ **Fit the screen (Shift-Home)** – автомасштабировать горизонтально и вертикально.
    - **Zoom-in (Shift-PgUp)** – увеличить (по вертикали и горизонтали).
    - **Zoom-out (Shift-PgDn)** – уменьшить (по вертикали и горизонтали).

Чтобы расширить выделенную область:

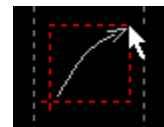
- В режиме **Zoom**  : щелкните и растащите выделенную область.
- В режиме **Cursors**  : нажмите и удержите клавишу **Ctrl**, затем щелкните и растащите выделенную область.

Выделение области экрана зависит от того, как указатель мышки перемещается относительно начальной точки.

- Если указатель мышки перемещается только вверх или вниз, будут показаны две горизонтальные линии, а когда левая клавиша мышки отпущена, выделенная область увеличится по вертикали.
- Если указатель мышки движется только влево и вправо, будут показаны две вертикальные линии, а когда левая клавиша мышки будет отпущена, выделенная область увеличится по горизонтали.
- Если указатель мышки движется диагонально, будет показан прямоугольник, а когда левая клавиша будет отпущена, выделенная область увеличится до заполнения экрана.



будет



В области графика, где выведена фаза, может быть выполнено только горизонтальное прокручивание и масштабирование.











Чтобы отменить или выполнить повторно ранее отмененное прокручивание и масштабирование, щелкните кнопки инструментальной панели:




















- **Undo** - отменить.
- **Redo** – выполнить повторно.

## Команды AC

Следующие команды, кнопки и горячие клавиши доступны в Основном Меню, на инструментальной панели AC анализа и в контекстных меню.

-  • **Open/Show AC window (F8)** – открыть/показать окно AC.
-  • **AC Settings** – открыть диалоговое окно **AC Settings**.
-  • **AC Data** – открыть окно **AC Data**.
-  • **Start AC (F9)** - запустить AC анализ.
-  • **Stop** - остановить анализ.
-  • **Log** - показать журнал регистрации анализа.
-  • **Sweep** – запустить серию анализов с изменяемым параметром.
-  • **Preferences** – открыть диалоговое окно **Preferences**.

### Toolbar and some context menus (инструментальная панель и контекстные меню:

-  • **Cursors.** Перемещение курсоров.
-  • **Zoom.** Масштабирование.
-  • **Scrolling.** Прокрутка.
-  • **Horizontal Zoom-in (Ctrl-PgUp)** – горизонтальное увеличение.
-  • **Horizontal Zoom-out (Ctrl-PgDn)** – горизонтальное уменьшение.
-  • **Fit the screen horizontal (Ctrl-Home)** – автомасштабировать горизонтально.
-  • **Fit cursors to screen** – привести курсоры к экрану.
-  • **Vertical Zoom-in (PgUp)** - вертикальное увеличение.
-  • **Vertical Zoom-out (PgDn)** - вертикальное уменьшение.
-  • **Fit the screen vertical (Home)** – автомасштабировать вертикально.
-  • **Fit the screen (Shift-Home)** – автомасштабировать горизонтально и вертикально.
-  • **Undo scale (Backspace).** Отменить прокрутку или масштабирование.
-  • **Redo scale.** Выполнить повторно отмененную прокрутку или масштабирование.
-  • Show/hide **Cursors** – скрыть/показать курсоры
-  • Show/hide **Data Table** – скрыть/показать таблицу данных.
-  • Show/hide **Legend** – скрыть/показать условные обозначения (список кривых).
-  • **Right maximum.** Переместить активный курсор к ближайшему справа максимуму выделенной кривой.
-  • **Left maximum.** Переместить активный курсор к ближайшему слева максимуму выделенной кривой.
-  • **Right minimum.** Переместить активный курсор к ближайшему справа минимуму выделенной кривой.

- **Left minimum.** Переместить активный курсор к ближайшему слева минимуму выделенной кривой.
- **Right unity gain.** Переместить активный курсор к ближайшей справа точке выделенной кривой с единичной амплитудой.
- **Left unity gain.** Переместить активный курсор к ближайшей слева точке выделенной кривой с единичной амплитудой.






### Graph commands (команды контекстного меню окна графиков):

- **Cursors** - открыть диалоговое окно **Cursors**.
- **Phase (Tab)** Выбор режима отображения фазы ►
  - **Off.** Не показывать фазу.
  - **On.** Показывать амплитуду и фазу в той же области графика.
  - **Separate.** Показывать амплитуду и фазу в отдельных областях графика.
- **Traces** ► (Следующие команды применяются только к кривым, показанным на графике)
  - **Open** - загрузить кривые из файла данных типа “nlf”.
  - **Import** - импортировать кривые из текстового файла формата “txt” или “csv”.
  - **Save** - сохранить кривые в файл данных типа “nlf”.
  - **View/Export** - посмотреть и экспортировать кривые в текстовом виде (“txt” или “csv” формат).
  - **Copy** - скопировать кривые в буфер обмена.
  - **Paste** - вставить кривые из буфера обмена.
- **Image** ►
  - **Copy to clipboard.** Скопировать изображение окна переходного процесса в буфер обмена.
  - **Save as BMP.** Сохранить изображение окна переходного процесса в файле формата «bmp».
  - **Save as JPG.** Сохранить изображение окна переходного процесса в файле формата «jpg».




### Storage command (команды хранения данных):

- **Move Run to storage** - перенести последние данные симуляции в **storage**.
  - **Remove** – удалить выделенные данные.
  - **Clear storage** - удалить все данные.
  - **Rename** - переименовать выделенные данные.
  - **Store last Run** – в начале анализа сохранять последние данные под именем **Last**.
  - **Selected only**
  - **Selected and dimmed**
  - **All**
- } Виды отображения **storage**

### Annotation commands (команды аннотаций):

-  • **Annotate selected trace** – аннотировать выделенную кривую.
-  • **Annotate all traces** – аннотировать все кривые.
-  • **Edit annotation** – редактировать аннотацию.
-  • **Delete annotation** – удалить аннотацию.
-  • **Delete all** – удалить все аннотации.







### Text commands (команды текста):

-  • **Insert text** – вставить текст на графике.
-  • **Edit text** – редактировать текст.
-  • **Delete text** – удалить текст.

### Data table commands (команды таблицы данных):

- **Hide trace name** - не показывать данную кривую в таблице.
- **Separate window** – показать таблицу в отдельном окне.

### Legend commands (команды legend – условных обозначений):

- **Hide trace name** - не показывать данную кривую на графике.
-  • **Remove** – удалить данную кривую.
-  • **Rename** – переименовать данную кривую.
-  • **Duplicate** – сдублировать данную кривую.
-  • **Copy** – скопировать данную кривую в буфер обмена.
-  • **Paste** – вставить кривые из буфера обмена.
-  • **Find component** - показать компонент на схеме.

## Клавиатура и горячие клавиши

Следующие клавиши клавиатуры и горячие клавиши можно использовать в окне переходного процесса:

- **Tab**. Переключить режим вывода фазы.
- **Left, Right** – прокрутить горизонтально.
- **Up, Down** – выбрать кривую.
- **Shift-PgUp** – увеличить (по вертикали и горизонтали).
- **Shift-PgDn** – уменьшить (по вертикали и горизонтали).

## Операции с мышкой

Следующие операции с мышкой можно использовать в окне переходного процесса.

- **Щелчок** (левая клавиша) - открыть контекстное меню.
- **Колесико мышки** – горизонтальное увеличение/уменьшение.
- **Ctrl- Колесико мышки** – горизонтальная прокрутка.
- **Shift- Колесико мышки** – вертикальная прокрутка.
- **Ctrl-Shift- Колесико мышки** – вертикальное увеличение/уменьшение.

В режиме *Cursors*  :

- **Щелчок** (левая клавиша) – если курсоры показаны, поставить ближайший курсор в эту точку.
- **Щелчок и перетаскивание (Click and drag)**.
  - На аннотации – передвинуть текст или указатель аннотации.
  - На тексте – передвинуть текст или указатель текста.
  - Иначе – передвинуть курсор.
- **Двойной щелчок (Double-click)**.
  - На аннотации – редактировать аннотацию.
  - На тексте – редактировать текст.
  - Иначе – показать курсоры, поставить оба курсора в эту точку.

В режиме *Zoom*  :

- **Щелчок и перетаскивание (Click and drag)** – выбрать и отмасштабировать.
- **Двойной щелчок (Double-click)** - центрировать экран.






В режиме *Scrolling*  :

- **Щелчок и перетаскивание (Click and drag)** – прокрутить график.
- **Двойной щелчок (Double-click)** - центрировать экран.

## Инструменты АС (AC Tools)

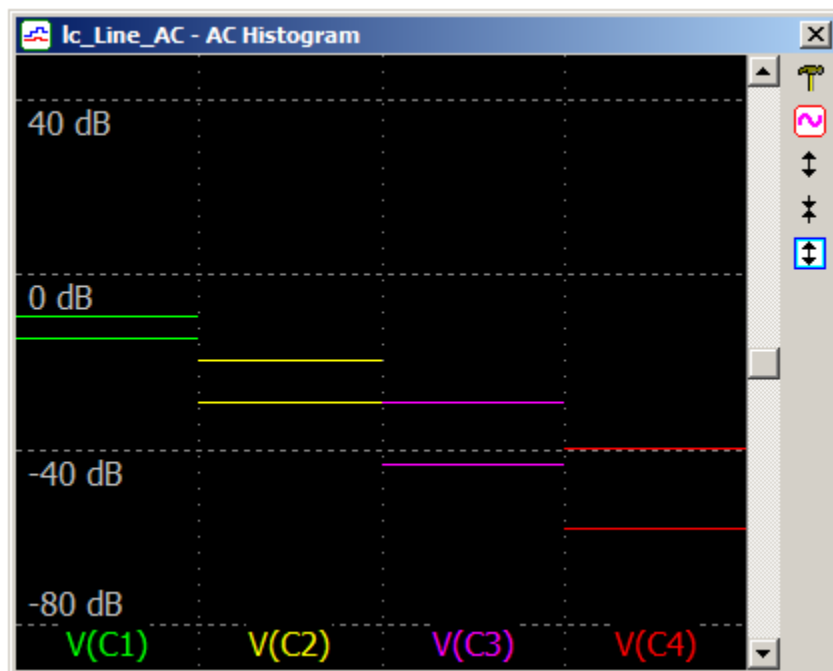
**AC Tools** позволяют сделать различные виды анализа данных АС и представить данные в разных форматах. Выберите **AC | Tools** и затем строку с требуемым инструментом.

В настоящее время доступны следующие инструменты:

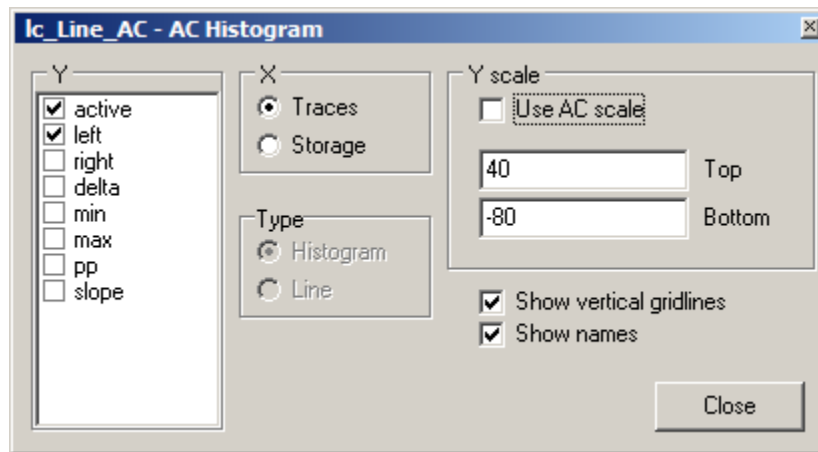
-  • Histogram (гистограмма)
-  • Smith chart (диаграмма Смита)
-  • Nyquist chart (диаграмма Найквиста)
-  • Nichols chart (диаграмма Николса)
-  • Markers (маркеры)

### Histogram (гистограмма)


Гистограмма представляет значения кривой и некоторые ее расчетные характеристики, полученные между курсорами (или на всем экране, если курсоры выключены), в графическом формате. Гистограмма может также показать «поперечное сечение» кривых или сохраненных данных.



-  • **Settings.** Открывает диалог **Settings**:




- **Y values.** Выберите переменные, отображаемые по оси **Y**. **active** — это выделенный в настоящий момент курсор (левый или правый). Другие значения – это то, что отображено в таблице данных AC.
- **X.** Выбирает режим гистограммы: что будет показано по оси **X**.
  - **Traces.** Показать «поперечное сечение» всех кривых, отображаемых сейчас на графике.
  - **Storage.** Показывает «поперечное сечение» **Storage** для всех кривых, отображаемых сейчас на графике.
- **Type.** Выбирает тип гистограммы для режима **Storage**:
  - **Histogram** - гистограмма.
  - **Line** - линии.
- **Y scale.** По умолчанию шкала **Y** диаграммы имеют тот же масштаб, что и вертикальная шкала экрана AC. Снимите флажок **Use AC scale** и введите нужные **Top** и **Bottom**.
- **Show vertical gridlines** – показать вертикальные линии, разделяющие данные гистограммы.
- **Show names** – показать имена кривых или **storage** на оси **X**.

 • **Use AC scale.**

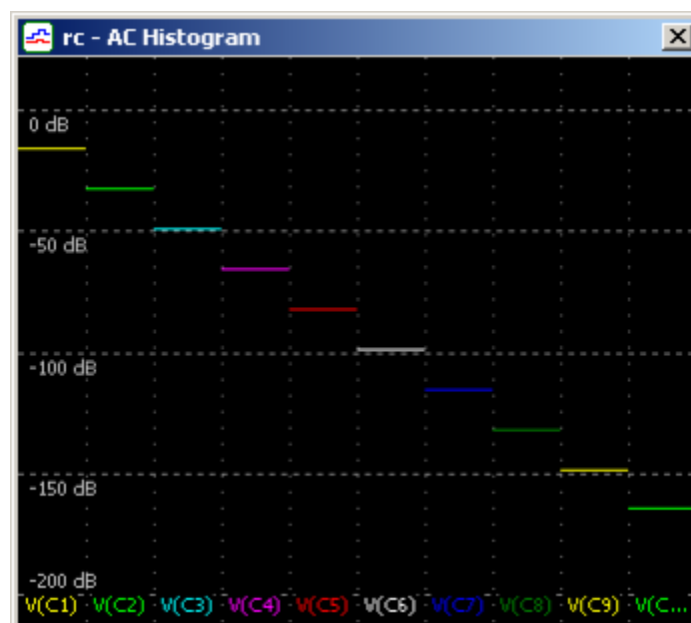
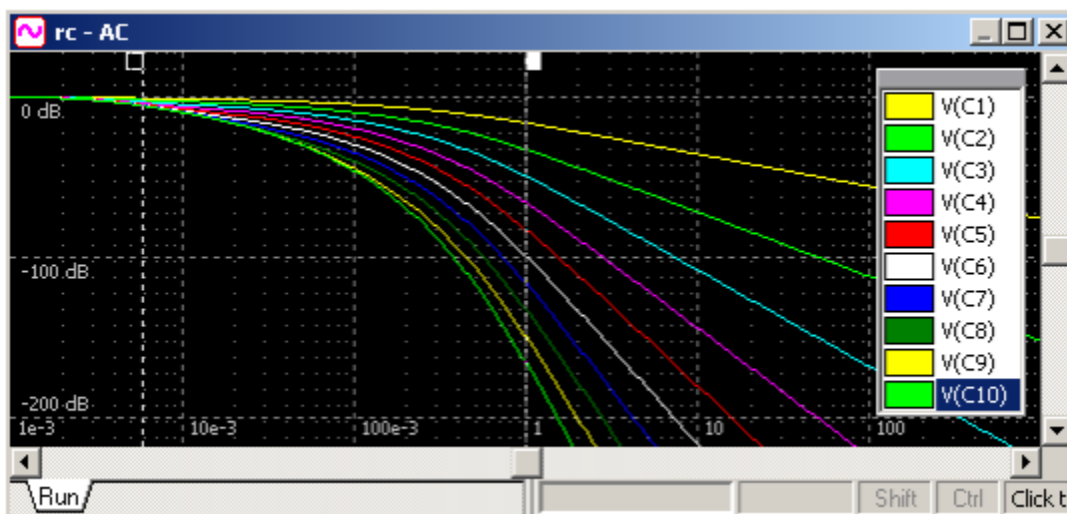
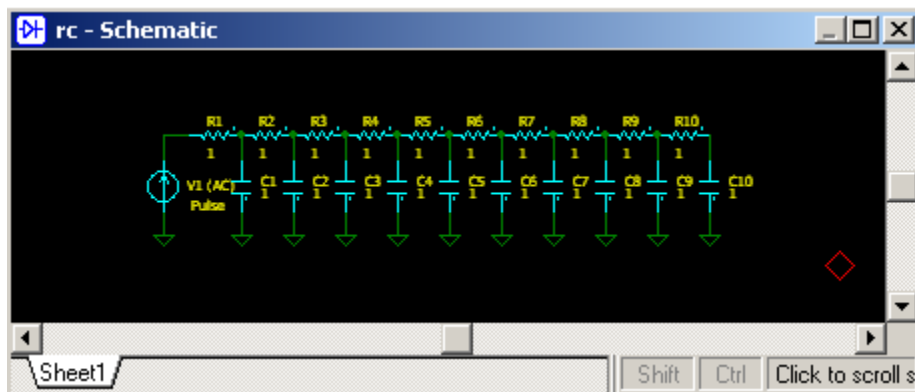
 • **Zoom-in.**

 • **Zoom-out.**

 • **Fit the screen vertical.**

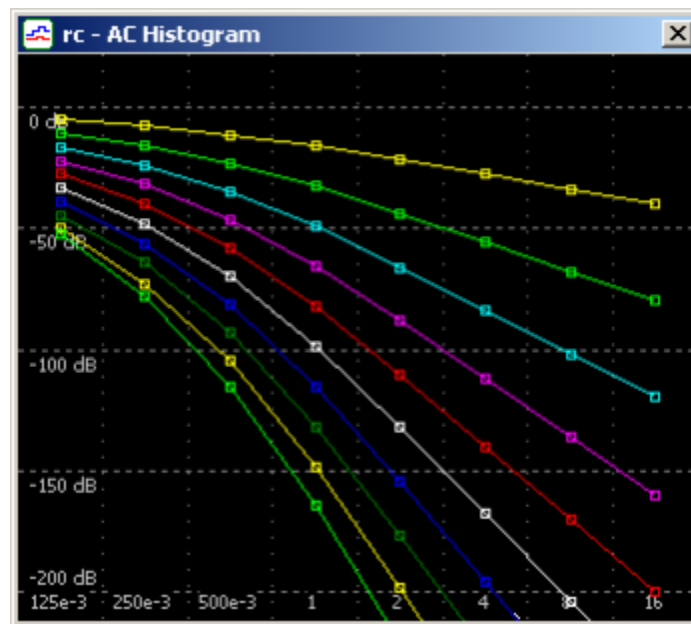
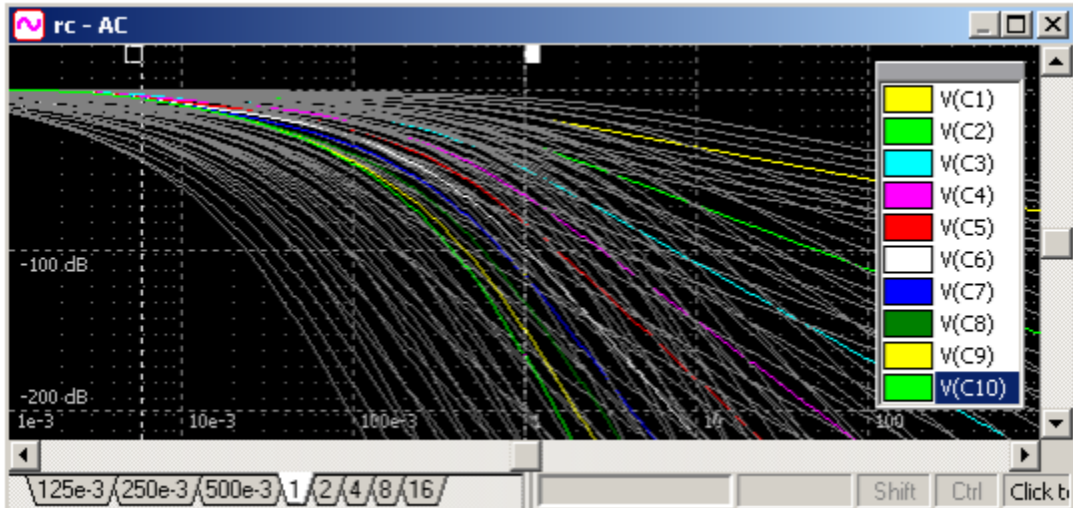
- Щелкните правой клавишей мышки по окну, чтобы увидеть доступные команды.
- Дважды щелкните по окну, чтобы открыть диалоговое окно **Settings**.

Режим **Traces** или «поперечное сечение кривых» могут использоваться для отображения «пространственного» распространения сигнала по схеме. Следующий пример показывает затухание на каждом звене RC фильтра на частоте 1 Гц.





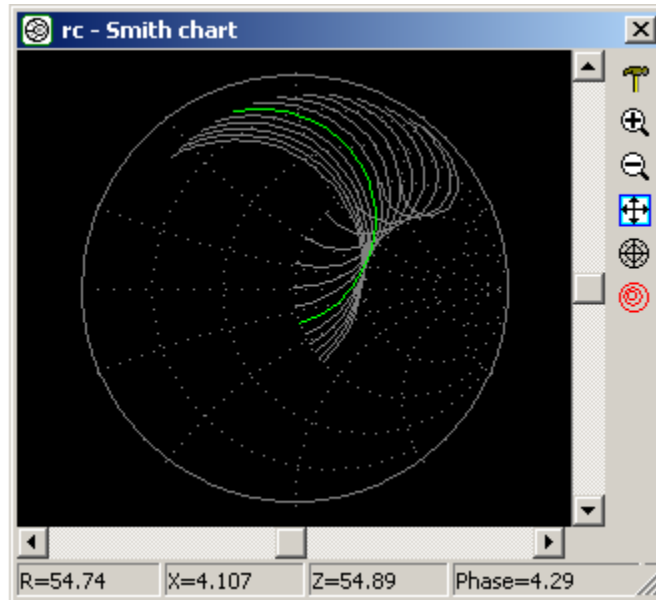
Режим **Storage** или «поперечное сечение **Storage**» может быть использован, чтобы показать, как значения кривых в какое-то время зависят от параметров схемы. Следующий пример показывает моделирование предыдущей схемы с сопротивлением, меняющимся от 0.125 до 16, с шагом X2 с помощью **sweep AC**. Каждый запуск запоминается в **Storage**. Ось X гистограммы — данные **Storage** (то есть сопротивление). Линии разного цвета показывают затухание на каждом звене RC фильтра на частоте 1 Гц (активный курсор), как функцию сопротивления.



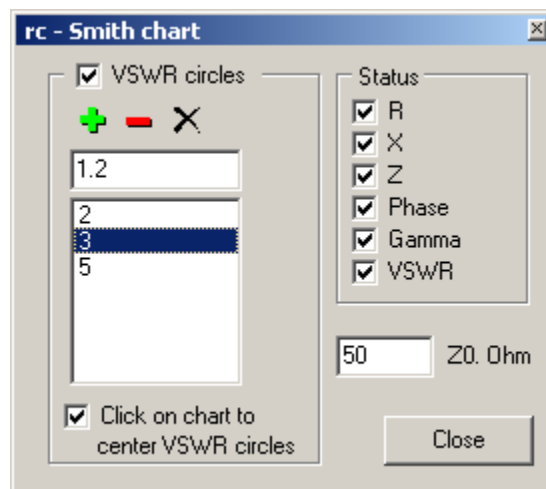
Можно заметить, что «поперечное сечение» гистограммы Storage при  $R=1$  то же, что и верхняя линия гистограммы Trase, показанной в предыдущем примере

## Диаграмма Смита

Это стандартная Smith Chart, рисующая комплексный коэффициент отражения. Заметьте, что диаграмма Смита предполагает выводить только Z кривые (импеданс). Фактически, она рисует все типы кривых AC, интерпретируя их как комплексный импеданс. Диаграмма показывает кривые только между курсорами (или на всем экране, если курсоры выключены).



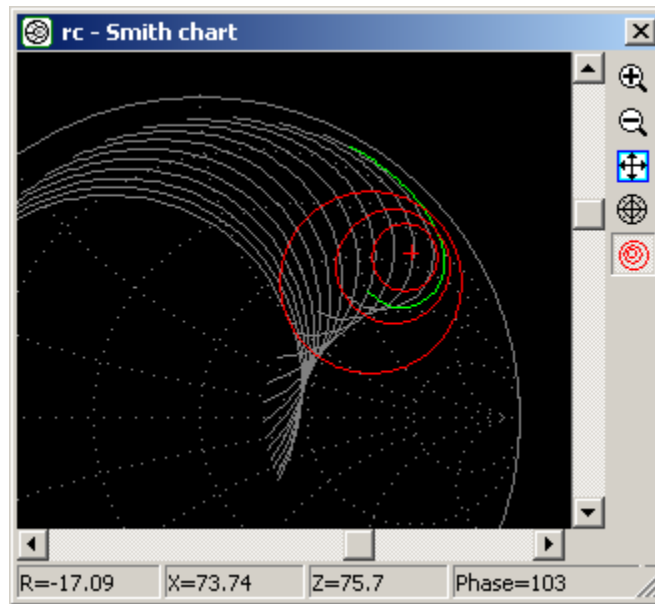
**T** • **Settings.** Открывает диалог **Settings**:



○ **VSWR circles.** Показывает окружности с заданными значениями VSWR вокруг выбранных точек. Значения показаны в списке VSWR.

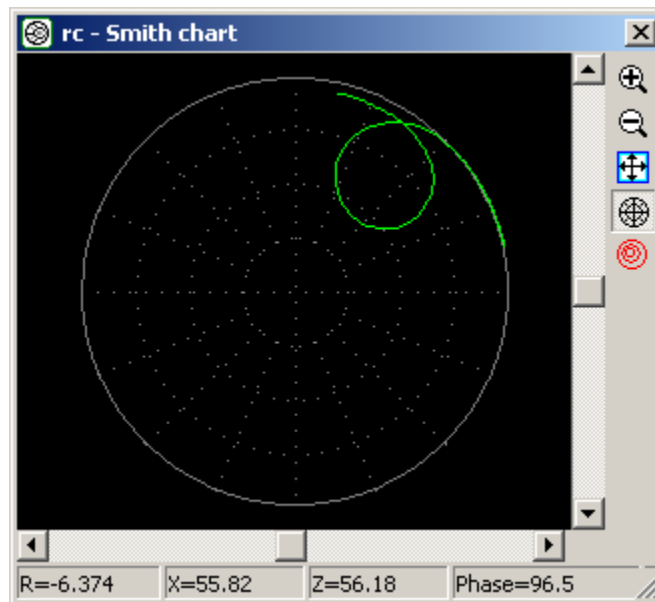
- +** ▪ **Add** – добавить VSWR окружность к списку.
- ▪ **Remove** – удалить VSWR окружность из списка.
- X** ▪ **Delete** – удалить все окружности из списка.

- **Click on chart to center VSWR circles (щелкните по диаграмме, чтобы отцентровать окружности VSWR).** Если выбрано, окружности VSWR могут быть показаны вокруг произвольно выбранной точки на диаграмме:








Если флажок снят, окружности VSWR будут показаны вокруг центра диаграммы.

- **Status.** Показывает выбранные значения на панели состояния. Значения показаны для позиции указателя мышки.
- **Z0, Ohm.** Характеристический импеданс диаграммы.

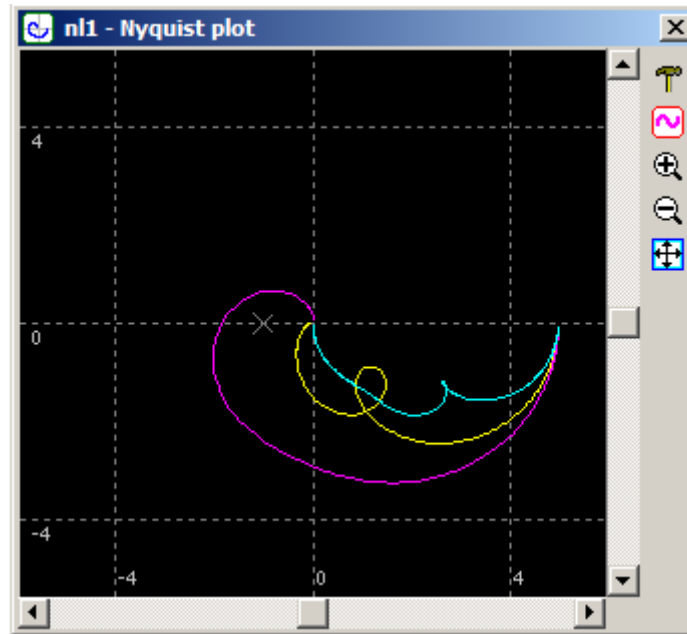


Вид диаграммы с полярной шкалой

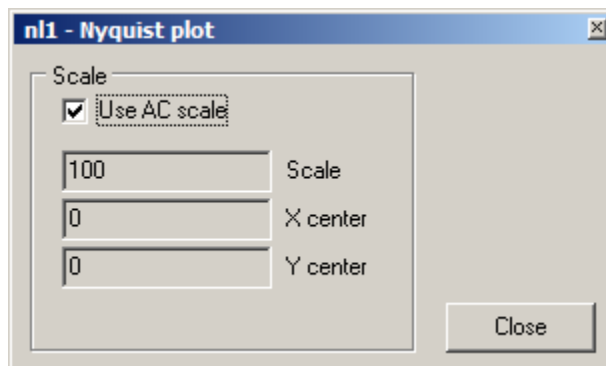
-  • **Zoom-in.**
-  • **Zoom-out.**
-  • **Fit the screen.**
-  • **Polar grid.**
-  • **Show/hide VSWR circles.**
  - Щелкните правой клавишей мышки по окну, чтобы увидеть доступные команды.
  - Дважды щелкните по окну, чтобы открыть диалоговое окно **Settings**.

## Диаграмма Найквиста




Диаграмма Найквиста (Nyquist plot) показывает комплексный АС отклик в полярных координатах. Диаграмма показывает кривые только между курсорами (или на всем экране, если курсоры выключены). «X» маркер показывает точку единичного усиления с фазой -180 градусов.



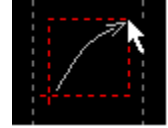
-  • **Settings.** Открывает диалог **Settings**:



По умолчанию **Scale**, **X center** и **Y center** диаграммы получаются из установок вертикальной шкалы АС графика. Снимите флажок **Use AC scale** и введите шкалу и индивидуальные значения **X** и **Y center**.

-  • **Use AC scale.**
-  • **Zoom-in.**
-  • **Zoom-out.**

- **Fit the screen.**
- Щелкните правой клавишей мышки по окну, чтобы увидеть доступные команды.
- Дважды щелкните по окну, чтобы отцентровать кривые.
- Щелкните левой клавишей мышки по окну и растащите выделенную область. Когда левая клавиша будет отпущена, выделенная область увеличится до заполнения экрана.

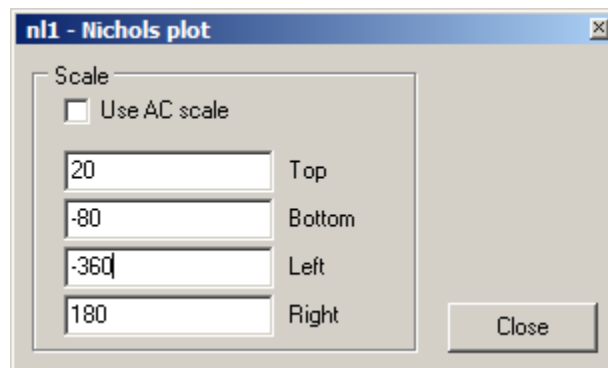


## Диаграмма Николса

Диаграмма Николса (Nichols plot) показывает логарифм модуля, как функцию фазы. Диаграмма показывает кривые только между курсорами (или на всем экране, если курсоры выключены).



- **Settings.** Открывает диалог **Settings**:



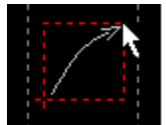
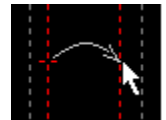
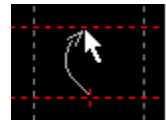
По умолчанию **Top**, **Bottom**, **Left** и **Right** диаграммы те же, что и для шкал модуля и фазы АС графика. Если нужно, снимите флажок **Use AC scale** и введите новые шкалы.

- **Use AC scale.**
- **Horizontal Zoom-in.**
- **Horizontal Zoom-out.**
- **Fit the screen horizontal.**

- **Vertical Zoom-in.**
- **Vertical Zoom-out.**
- **Fit the screen vertical.**
- **Fit the screen.**

- Щелкните правой клавишей мышки по окну, чтобы увидеть доступные команды.
- Дважды щелкните по окну, чтобы отцентровать кривые.
- Щелкните левой клавишей мышки по окну и растащите выделенную область. Выделение области экрана зависит от того, как указатель мышки перемещается относительно начальной точки.

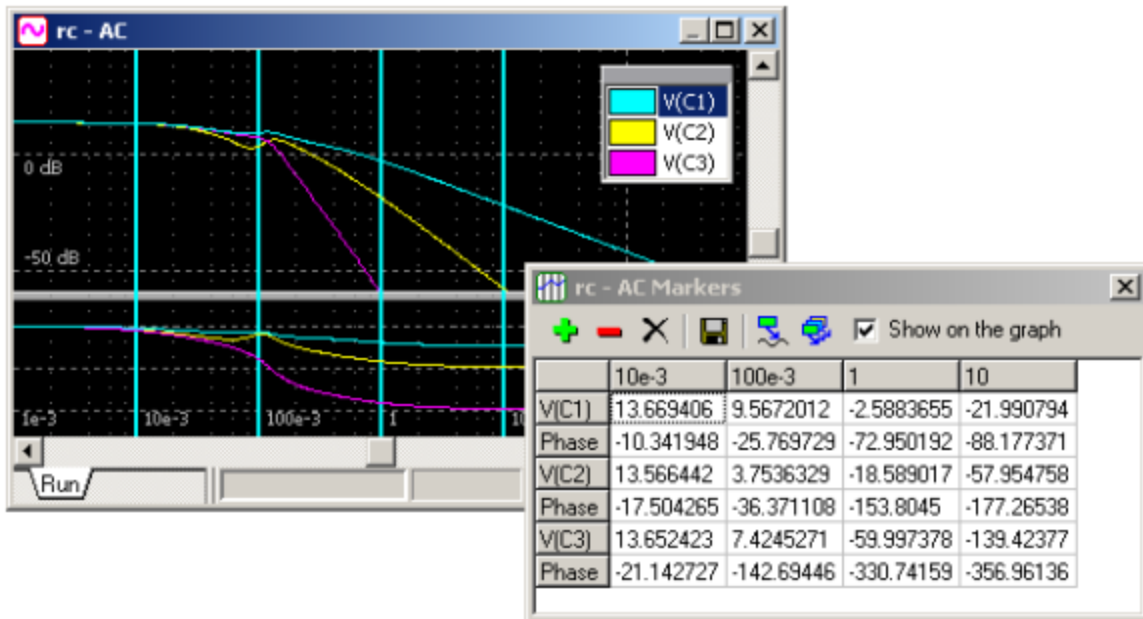
- Если указатель мышки перемещается только вверх или вниз, будут показаны две горизонтальные линии, а когда левая клавиша мышки будет отпущена, выделенная область увеличится по вертикали.
- Если указатель мышки движется только влево и вправо, будут показаны две вертикальные линии, а когда левая клавиша мышки будет отпущена, выделенная область увеличится по горизонтали.
- Если указатель мышки движется диагонально, будет показан прямоугольник, а когда левая клавиша будет отпущена, выделенная область увеличится до заполнения экрана.





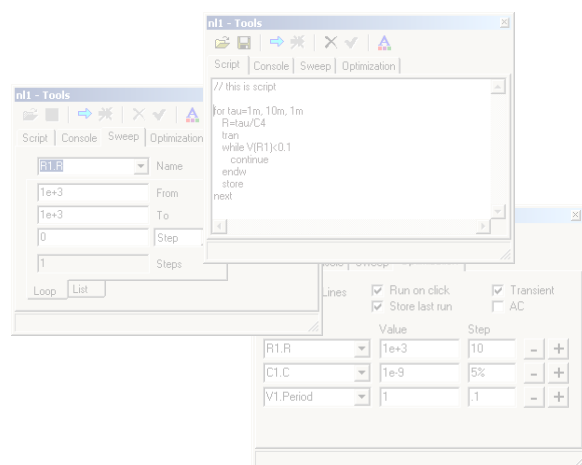
## Markers (маркеры)

Маркеры предлагают удобный способ отслеживания амплитуд кривых в заданных точках. В отличие от курсоров, маркеры всегда остаются в заданном положении. Количество маркеров не ограничено. Ниже вы можете видеть 4 маркера, показанных на графике АС, с амплитудами и фазами кривых, отображаемыми в таблице **Markers table**.








- **Таблица маркеров** содержит кривые, изображенные на графике.
- **Add** – добавить новый маркер. Частота маркера – это частота активного курсора.
- **Remove** – удалить маркер из таблицы. Щелкните любую ячейку в таблице, принадлежащую колонке маркера, затем щелкните по кнопке. На примере вверху выделен первый маркер.
- **Delete** – удалить все маркеры.
- **Export** – экспортировать таблицу маркеров в файл в текстовом (“csv”) формате.
- **Annotate** аннотировать выделенную кривую на позициях маркеров на графике АС.
- **Annotate all traces** аннотировать все кривые на позициях маркеров на графике АС.
- **Show on the graph** – выберите, чтобы показывать маркеры на графике АС. Ширина и цвет маркеров задаются на странице **Graphs** диалогового окна **Preferences**.

# VI. Инструменты (Tools)




**Инструменты** используются для выполнения и автоматизации некоторых сложных задач со схемой посредством языка скрипта. Большинство Инструментов расположено в окне **Tools**. Используйте следующие команды Основного Меню (меню **Tools**) или кнопки инструментальной панели, чтобы открыть инструменты:

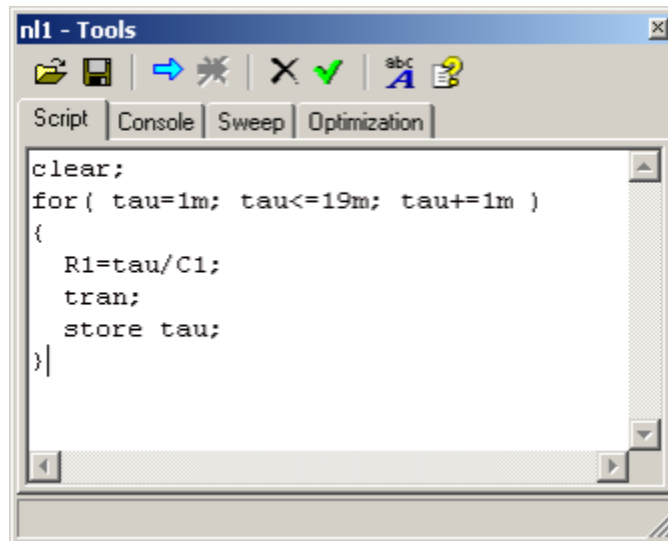
-  • **Script** – открыть, сохранить, редактировать и запустить скрипт.
-  • **Console** – показать журнал (log) выполнения скрипта. Здесь же есть командная строка, которая работает как калькулятор и позволяет выполнять некоторые команды скрипта.
-  • **Sweep** - выполнить серию запусков анализа переходного процесса и/или АС изменяя параметр компонента.
-  • **Optimization** - выполнять анализ переходного процесса и/или АС итерируя выбранные параметры компонентов вручную.
-  • **HTTP Link** – сконфигурировать и запустить NL5 HTTP сервер для работы с внешними программами по HTTP ссылке.

## Скрипт








На странице **Script** можно открывать, редактировать, сохранять и выполнять скрипт: программу, написанную на упрощенном языке программирования **C**. В дополнение к стандартным операторам и математическим функциям в скрипте можно использовать специальные команды NL5, позволяющие изменять параметры компонентов, запускать анализ, анализировать и сохранять данные симуляции.

Команды скрипта можно также выполнять из командной строки приложения NL5, на странице **Console** окна инструментов и из внешней программы через **HTTP ссылку**. Это позволяет использовать NL5 как «дополнение» к популярным инженерным средствам, таким как MATLAB®, PYTHON и другим.

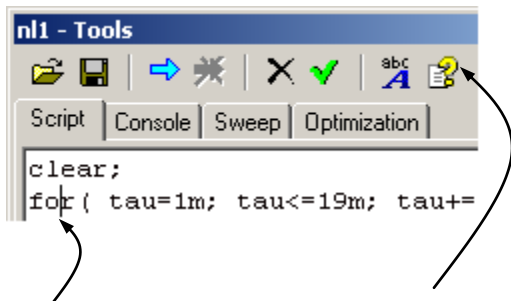
Щелкните по кнопке **Script**  или выберите команду **Tools | Script** в Основном Меню, чтобы открыть страницу **Script** окна **Tools**:



Кнопки инструментальной панели выполняют следующие операции:

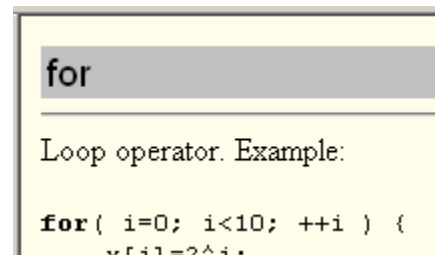
-  • **Open script** – загрузить скрипт из текстового файла.
-  • **Save script** – сохранить скрипт в текстовом файле.
-  • **Run script** – выполнить скрипт.
-  • **Stop script** – остановить выполнение скрипта.
-  • **Clear** – очистить текст.
-  • Проверить скрипт на наличие синтаксических ошибок.
-  • **Font** – выбрать шрифт.

- **Help (F1).** - Помощь. Чтобы увидеть раздел Помощи для определенного оператора, функции или команды, поместите курсор на слово в тексте скрипта и щелкните по кнопке **Help** или нажмите **F1**.



Place cursor on **for** operator

Click **Help** button  
or press **F1**



Help on **for** operator is displayed

## Синтаксис скрипта

В качестве языка скрипта использован упрощенный язык программирования С (см. **Приложение 4** для справки и примеров). Все операторы (**Приложение 2**) и функции (**Приложение 3**) могут использоваться в скрипте. Специальные команды NL5 (**Приложение 5**) могут также быть использованы для запуска анализа, анализа и сохранения данных симуляции.

**Параметры компонентов** и переменные схемы могут использоваться в выражениях и изменяться аналогично локальным переменным:

```
R1=tau/C1;
freq=1./V1.period;          // "freq" is schematic variable
```

**Данные кривых и курсоров** могут использоваться в выражениях. Эти данные соответствуют тому, что показывается в таблице данных переходного процесса или АС. Чтобы использовать данные кривой, кривая **должна** быть добавлена в список кривых, но не обязательно, чтобы она отображалась на графике или в таблице данных. Например:



```
x = V(out).max - V(out).min;
if( V(C1).(3.45)>threshold ) break;
charge=I(C1).mean * delta;
```

**Команды скрипта.** Команды скрипта используются, чтобы загружать схему, управлять процессом симуляции, сохранять данные анализа в различных форматах. См. **Приложение 5** для полного списка и описания команд. Заметьте, что в отличие от стандартного формата вызова функций в языке С, параметры команд скрипта не обязательно заключать в скобки. Однако для общности кода можно и заключать. Например::


```
open(rc.nl5);
tran(0,1,1m);
close();
```

## Выполнение скрипта

Выполнение скрипта можно запустить любым из следующих способов:

- Запуск скрипта из окна **Tools**:
  - Выберите страницу **Script**.
  - Введите код скрипта или щелкните по кнопке **Open** , выберите файл и загрузите скрипт из файла.
  - Щелкните по кнопке **Run script**  .
- Запуск из среды Windows:
  - Перетащите и сбросьте иконку файла с текстом скрипта на иконку NL5.
- Запуск из командной строки или другого приложения:
  - Запустите NL5 с именем файла скрипта в качестве параметра. Например:

```
n15.exe myscript.txt
```

Когда скрипт запущен, никакие изменения в схеме, окне переходного процесса или АС не могут быть сделаны вручную. Доступна только кнопка **Stop script**  . Журнал (**log**) и сообщения об ошибках отображаются на странице **Console**. Команды скрипта применяются к активному документу.

Если скрипт выполняет команды **tran** или **cont**, «непрерывный» режим анализа переходного процесса автоматически выключается: переходной процесс будет приостановлен в конце экрана. Следующая команда скрипта не будет выполняться, пока анализ переходного процесса не приостановлен. После этого анализ может быть продолжен командой **cont**.

Если скрипт выполняет команду **AC**, следующая команда скрипта не будет выполняться, пока анализ АС не закончен.

## Примеры скрипта

**Установить параметры компонентов.** Параметры компонентов могут быть вычислены во внешней программе (например, в Excel), или введены вручную и сохранены в текстовом файле в формате *name=value*:

```
R1=5.1;  
C1=12e-9;  
V3.period=0.01
```

При запуске скрипта новые параметры будут присвоены компонентам.

**Цикл с изменением параметра.** Параметр компонента меняется в заданном диапазоне, для каждого значения выполняется анализ переходного процесса, результаты помещаются в **storage**:

```

for( R1=1; R1<=10; R1+=1 )
{
    tran;
    store R1;
}

```

**Изменение параметра из списка.** Параметр компонента принимает значения из списка, для каждого значения выполняется АС анализ, результаты помещаются в **storage**:

```

for( V1.period=1m, 2m, 10m, 50, 100m )
{
    ac;
    store V1.period;
}

```

**Цикл с изменением локальной переменной.** Локальная переменная меняется в заданном диапазоне, для каждого значения переменной вычисляются новые значения параметров компонентов, выполняется анализ переходного процесса, результаты помещаются в **storage**:

```

for( freq=1; freq <=10; freq*=1.1 )
{
    V2.period=1/freq;
    R2=1/(freq*C5);
    tran;
    store freq;
}

```

**Ожидание выполнения условия.** Переходной процесс выполняется до тех пор, пока значение от пика до пика кривой меньше, чем заданный порог. Когда это происходит, сохраняются начальные условия (Initial Conditions).

```

threshold=1e-6;
tran;
while( v(c1).pp>threshold ) cont;
saveic;

```

**Выполнение анализа для заданного файла, сохранение данных, выход из приложения.** Файл схемы загружается в NL5, меняются параметры компонента, выполняется анализ переходного процесса, кривая экспортируется в «csv» файл, NL5 закрывается. Этот скрипт может быть выполнен из командной строки.

```

open lcr.nl5;
R1=100;
C1=1n5;
tran;
export data.csv;
exit;

```


**Выполнение анализа для заданного файла, регистрация данных, выход из приложения.** Файл схемы загружается в NL5, выполняется цикл с изменением параметра компонента,

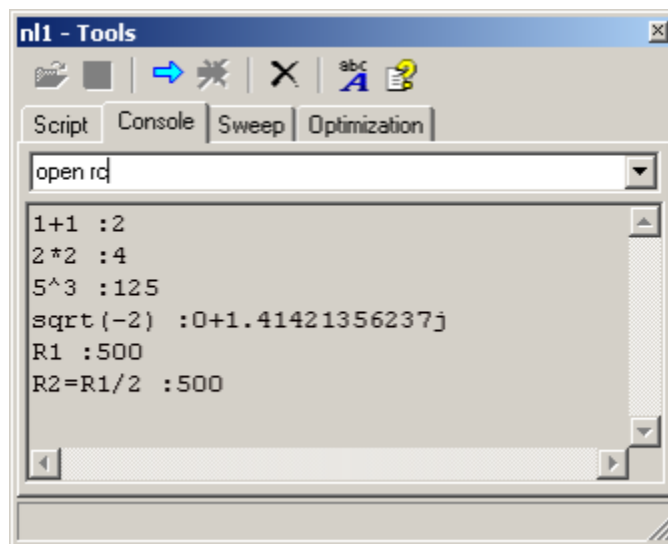
выполняется анализ переходного процесса, данные кривой записываются в текстовый файл, NL5 закрывается. Этот скрипт может быть выполнен из командной строки.

```
open lcr.nl5;
logdata lcrdata.csv, r1, v(r1).mean, v(r1).rms;
for( R1=100; R1<=1000; R1+=100 )
{
  tran;
  logdata;
}
exit;
```




## Console (консоль)

Страница **Console** отображает журнал выполнения скрипта. Она также имеет командную строку, которая работает как калькулятор и позволяет выполнять некоторые команды скрипта.

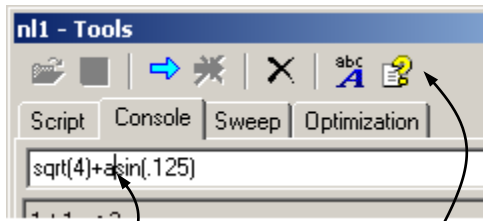
Щелкните по кнопке **Console**  или выберите команду **Tools | Script** основного меню, чтобы открыть страницу **Console** окна **Tools**.



Кнопки инструментальной панели выполняют следующие операции:

-  • **Clear** – очистить текст.
-  • **Font** – выбрать шрифт.
-  • **Help (F1)**. Помощь. Чтобы увидеть раздел Помощи для определенного оператора, функции или команды, поместите курсор на слово в командной строке и щелкните по кнопке **Help** или нажмите **F1**.





Place cursor on **asin** function

Click **Help** button  
or press **F1**

 A screenshot of a help window titled 'asin, acos, atan, atan2'. Below the title, it says 'C column indicates if function supports cc'. There is a table with two columns: 'Function' and 'C'. The first row shows 'asin(x)' and 'arcsin(x)'.
 

| Function | C         |
|----------|-----------|
| asin(x)  | arcsin(x) |

Help on **asin** function is displayed

## Командная строка

Командная строка используется для вычисления выражений. Выражение может содержать:

- Числа (включая комплексные).
- Параметры компонентов и переменные схемы активного документа.
- Арифметические операторы и функции.
- Данных кривых.

Введите выражение в командную строку и нажмите **Enter**. Результаты будут отображены в текстовом окне. Например:

```

2*2           : 2*2=4
sin(45)       : sin(45)=707.106781187e-3
R1*C1         : R1*C1=15
sqrt(-2)      : sqrt(-2)=0+1.41421356237j
V(R1).mean    : V(R1).mean=0.15425
  
```

Чтобы изменить параметры компонента или переменные активного документа, введите имя параметра со знаком равенства и выражением:

```

R1=1k
R1=1000/C1
V1.model=pulse
  
```

Вы можете также выполнить команды скрипта из командной строки: доступны все команды скрипта, за исключением *logdata* и *exit*. Также доступна команда *pause*: приостановить выполнение анализа переходного процесса. Команды применяются к активному документу:


```

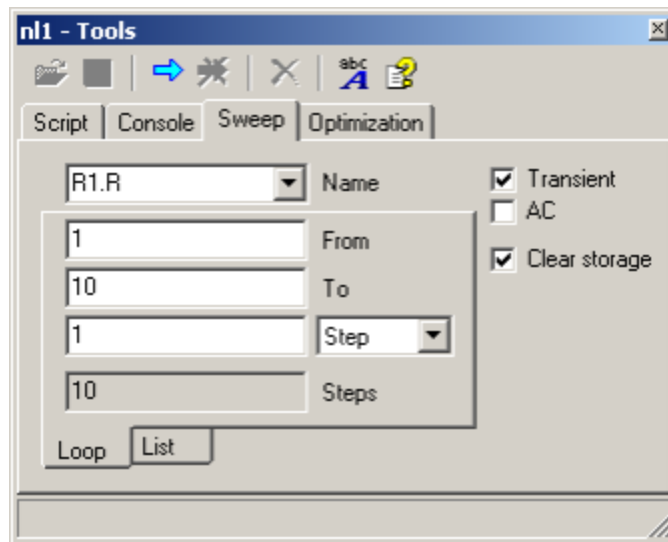
open rc.nl5
tran 0,1m,.1u
store
cont
  
```

Предыдущие выражения и команды могут быть найдены в выпадающем списке или с помощью клавиш клавиатуры **Up** и **Down**. Нажмите **Esc**, чтобы очистить командную строку.




## Sweep (серия)

Страница **Sweep** предоставляет возможность автоматически выполнить серию анализов переходного процесса и/или АС анализа при изменении параметра компонента в заданном диапазоне и сохранить результаты анализа в **storage**.

Щелкните по кнопке **Sweep**  или выберите команду **Tools | Sweep** в Основном Меню, чтобы открыть страницу **Sweep** окна **Tools**:

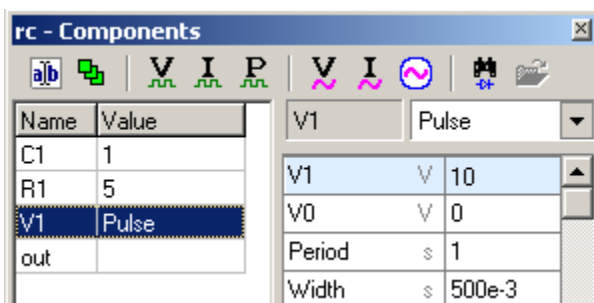


Кнопки инструментальной панели выполняют следующие операции:

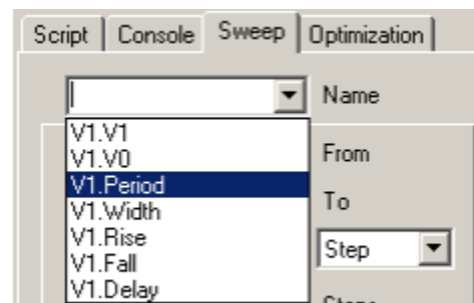
-  • **Run** – запустить серию.
-  • **Stop** - остановить.
-  • **Clear** – очистить список параметров (режим “list”).

Есть два режима выполнения серии: **Loop** (цикл) и **List** (список). Установите следующие параметры для обоих режимов:

- **Name**. Имя параметра для изменения. Введите имя параметра вручную или выберите из выпадающего списка. Список показывает все числовые параметры компонента, который сейчас выбран в схеме или в окне **Components**. Например:



Select component V1



Select parameter V1.Period

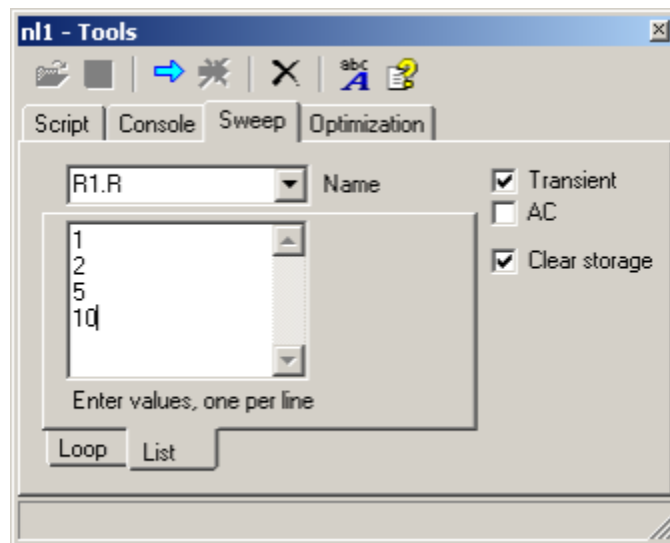
- **Transient** – выберите для анализа переходного процесса.
- **AC** - выберите для AC анализа.
- **Clear storage** – если выбрано, данные **storage** будут стираться перед началом новой серии. Иначе результаты новой серии будут добавлены к существующим данным **storage**.
- Выберите режим серии: **Loop** или **List**.


Для режима **Loop** задайте следующие параметры:

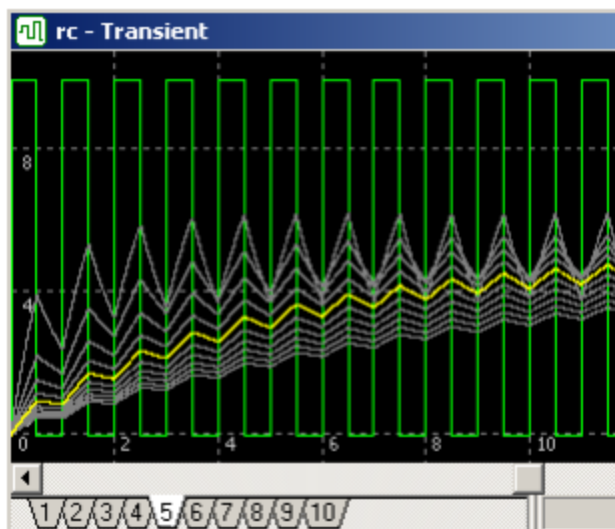
- **From** - начальное значение изменяемого параметра.
- **To** - конечное значение изменяемого параметра.
- **Step type** (тип шага):
  - **Step**. Параметр увеличивается (или уменьшается) со значением **Step**.
  - **Step, %**. Параметр меняется в заданном процентном соотношении.
  - **Step, X**. Параметр умножается на заданное значение.
- В поле **Steps** будет показано полное количество шагов, которое будет выполнено.

Для режима **List** задайте следующие параметры:

- **Список** значений изменяемого параметра (одно значение в строке):



Щелкните по кнопке **Run** , чтобы запустить серию. Результаты анализа будут записаны в **storage**. (Только кривые с разрешенной записью в **storage** будут сохранены). Страница **Script** содержит текст скрипта, который при этом выполнялся, а страница **Console**: журнал выполнения скрипта.



Sweep results in the storage

```
R1.R=1;
tran;
store R1.R;
R1.R=2;
tran;
store R1.R;
R1.R=5;
tran;
store R1.R;
```

Script text


```
<Script started>
R1.R=1 :1
tran :OK
store R1.R :OK
R1.R=2 :2
tran :OK
store R1.R :OK
```

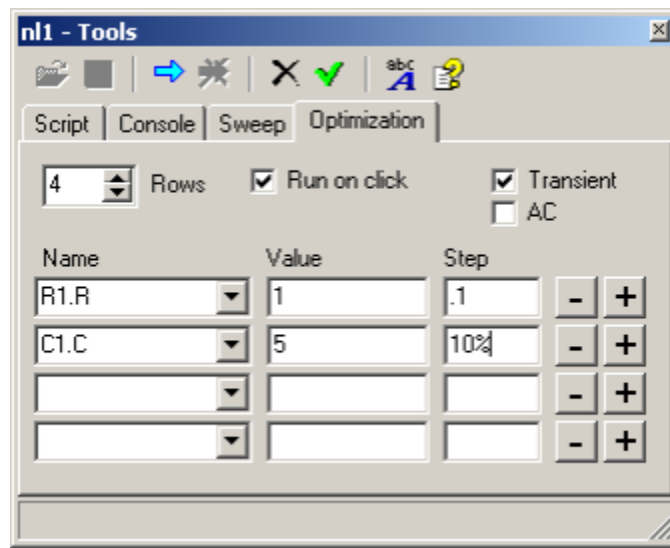
Execution log

Команды **Transient | Sweep**  и **AC | Sweep**  открывают страницу **Sweep** с установленными **Transient** или **AC** флажками, соответственно.





## Оптимизация

Оптимизация выполняет анализ переходного процесса и/или АС анализ при ручной итерации выбранных параметров.


Щелкните кнопку **Optimization**  или выберите команду **Tools | Optimization** в Основном Меню, чтобы открыть страницу **Optimization** окна **Tools**:



Кнопки инструментальной панели выполняют следующие операции:

-  • **Run** – запустить анализ.
-  • **Stop** – остановить.
-  • **Clear** – очистить таблицу оптимизации.
-  • **Clean up and update** – привести в порядок таблицу оптимизации: удалить строки с пустым именем параметра, при необходимости передвинуть строки вверх. Выставить текущие значения компонентов в поле **Value**.

Могут быть заданы следующие параметры:

- **Rows**. Количество параметров для итераций, от 1 до 16. Если все строки не помещаются в окне, в строке состояния появится предупреждающее сообщение: «*Resize window to see more rows* — измените размер окна, чтобы увидеть больше строк».
- **Run on click**. Если выбрано, заданный анализ будет стартовать немедленно, сразу после щелчка по кнопкам **Plus** или **Minus** любого параметра. Иначе, щелкните по кнопке , чтобы запустить анализ.
- **Transient** - выбрать для запуска анализа переходного процесса.
- **AC** - выбрать для запуска АС анализа.
- Поля **Name** содержат имена параметров, которые будут меняться. Введите имя параметра вручную или выберите из выпадающего списка. Список показывает все числовые

параметры компонента, которые в настоящее время выделены на схеме или в окне **Components** (аналогично выбору параметра для **sweep**).

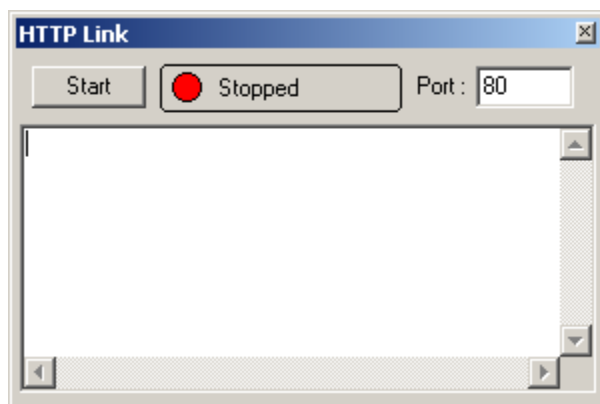
- Поля **Value** содержат текущее значение заданного параметра. Щелкните кнопку **Clean up and update** чтобы обновить значения.
- Поля **Step** задают, как значение будет меняться, когда нажимаются кнопки **Plus** или **Minus**. Если **Step** — это число, к текущему значению будет добавлено число. Если **Step** — это число с символом «%» на конце, значение будет меняться на заданный процент.
- Кнопки **Plus/Minus**. When clicked, the value will be modified by specified step. Щелчок по ним модифицирует значение на заданный шаг. Если выбрана опция **Run on click**, будет выполняться заданный анализ.

## HTTP link

**HTTP link** позволяет установить связь («линк») NL5 с внешними программами. NL5 выступает в роли «сервера» - в ней работает встроенный HTTP сервер. Внешние программы являются «клиентами». NL5 и программа-«клиент» могут работать на одно и том же компьютере или на разных компьютерах, общающихся друг с другом через локальную сеть или Интернет.

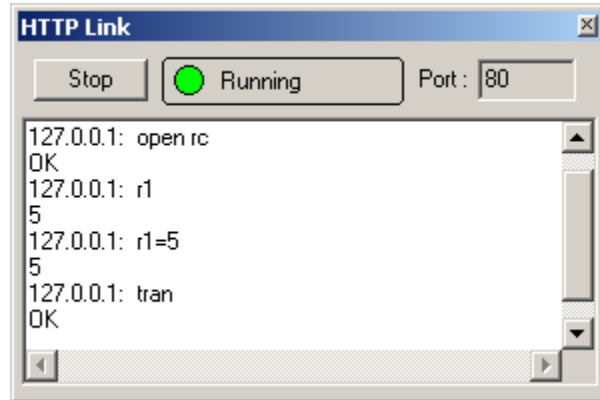
### Запуск HTTP сервера

Выберите команду **Tools | HTTP Link** Основного Меню, чтобы открыть окно **HTTP Link**:



Значение параметра **Port** по умолчанию - 80, что является стандартным номером порта для HTTP протокола. Если этот порт уже используется другой программой (скорее всего HTTP сервером), выберите любой другой номер порта в диапазоне, допустимом для TCP портов. Удостоверьтесь, что программа-«клиент» использует такой же номер в команде URL запроса. Заметьте, что номер порта может быть изменен только до того, как HTTP сервер был запущен. Чтобы изменить номер порта после того, как HTTP сервер был запущен, необходимо закрыть программу и запустить ее снова. Если NL5 и программа-«клиент» работают на разных компьютерах, удостоверьтесь, что используемый порт разрешен для TCP обмена всевозможными антивирусами, файрволами, роутерами и т.д.

Чтобы запустить HTTP сервер, щелкните по кнопке **Start**. Когда сервер запущен, в текстовом окне будет показываться IP адрес «клиента», полученная команда и результат, посланный «клиенту»:



Окно **HTTP link** может быть закрыто в любой момент, не влияя при этом на работоспособность сервера. Чтобы остановить сервер, щелкните по кнопке **Stop**.

HTTP сервер может также быть автоматически запущен при старте NL5 из командной строки с ключом “-http”:

```
>n15.exe -http
```

## Посылка URL запроса

Программа-«клиент» может посылать команды и получать данные из NL5, посылая URL запрос с параметрами и получая в ответ данные в текстовом формате. Имя, формат и параметры команды URL запроса (URL чтения) могут быть разными в разных программах. Например, в системе MATLAB используется функция `urlread`:

```
s = urlread('url')
```

Для других программ ищите информацию о командах URL запроса с параметрами в Руководстве или Помощи.

Параметр URL запроса (URL строка) имеет следующий формат:

```
http://host_name:port/?cmd=command
```

где:

- **host\_name** – имя или IP адрес компьютера, на котором запущена программа NL5 с HTTP сервером. Если программа-«клиент» работает на том же самом компьютере, *host\_name* может принимать значение “127.0.0.1” или “localhost”.
- **port** - номер порта. Должен быть таким же, как номер порта, определенный в конфигурации HTTP сервера в NL5. Если используется стандартный HTTP порт 80, номер порта можно опустить, так что URL строка будет выглядеть так:

```
http://host_name/?cmd=command
```



- **command** – команда скрипта или выражение.

NL5 выполняет команду или вычисляет выражение и возвращает текст, который может быть содержать следующее:

- текст “OK”,
- результат вычисленного выражения в текстовом виде (число),
- строка чисел, разделенных запятыми, для запроса данных кривой,
- сообщение об ошибке.

Например:

**Запрос:**

```
http://localhost/?cmd=open rc.n15
http://127.0.0.1/?cmd=R1
http://192.168.0.1/?cmd=C1=1n
http://public025:2119/?cmd=V(C1) 0,1,.2
http://localhost/?cmd=open test.n15
```

**Ответ:**

```
“OK”
“100”
“2.2e-9”
“0,9.99999424754,9.9999944731,...”
“Error opening file test”
```

Если позволяет программа-«клиент», к URL строке могут быть применены следующие упрощения, делающие ее проще и легче читаемой:

- Текст “http://” может быть опущен.
- Текст “cmd=” может быть опущен.
- Несколько команд могут быть посланы в одной URL строке. Команды должны быть разделены символом ‘&’, а ответ на запрос будет состоять из ответов на каждую команду, разделенных запятыми:

Например:

**Запрос:**

```
127.0.0.1/?open rc
192.168.0.101/?R1&C1&L1
http://public025:222/?store R1
localhost/?V(C1) .mean
```

**Ответ:**

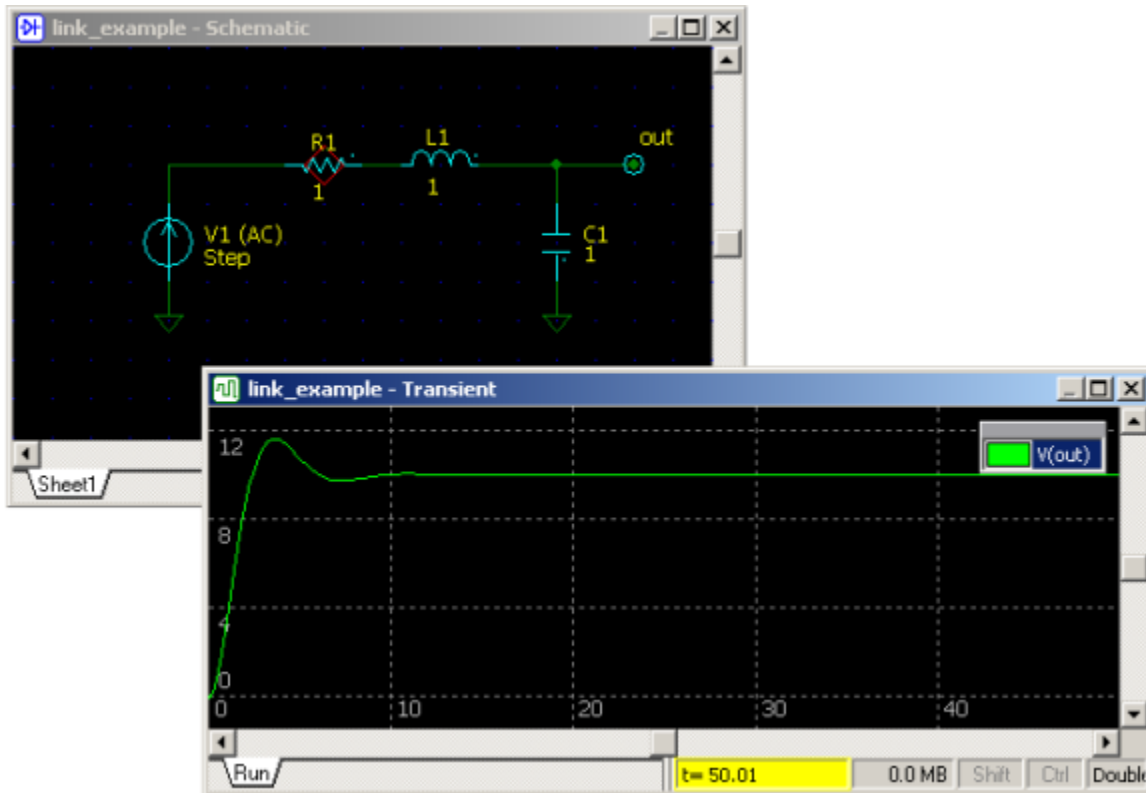
```
“OK”
“100,2.2e-9,100e-3”
“OK”
“1.27978684602”
```

## Запуск симуляция

Если через HTTP линк запрошена команда анализа переходного процесса **tran** (или **cont**) или AC анализа **AC**, NL5 посылает ответ “OK” немедленно, без ожидания окончания анализа. Это сделано для того, чтобы избежать возможных тайм-аутов в случае долгого процесса симуляции. Программа-«клиент» должна сама проверять, окончена ли симуляция, периодически посылая команду **ready** и проверяя ответ NL5. Ответ “0” означает, что симуляция еще не окончена, “1” – симуляция окончена и может быть послана следующая команда. Если во время симуляции произошла ошибка, NL5 возвратит сообщение об ошибке.

## Пример связи NL5-MATLAB

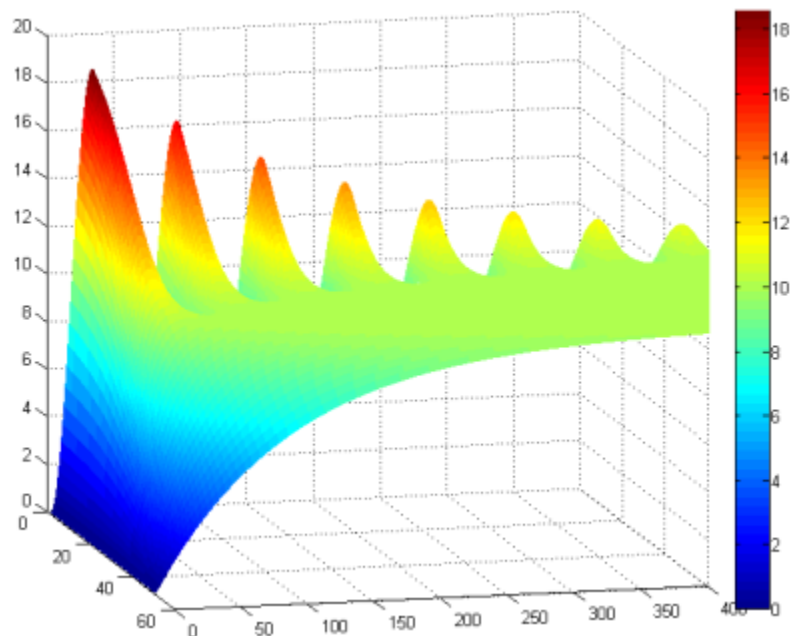
Следующий пример показывает использование NL5 совместно с системой MATLAB. В NL5 запущен HTTP сервер и загружена схема из файла “link\_example.nl5”. В окне переходного процесса показан отклик схемы на ступеньку напряжения на источнике V1 для R1=1:



Приведенная ниже программа загружена в MATLAB из файла “link\_example.m” и выполнена::

```
clear
clc
close all
R=logspace(-1,1);
Header='http://127.0.0.1/?';
Cmd2=[Header,'tran'];
Cmd3=[Header,'ready'];
Cmd4=[Header,'V(out) 0,50,.1'];
for k=1:length(R)
    Cmd1=[Header,'R1=',num2str(R(k))];
    urlread(Cmd1);
    urlread(Cmd2);
    Response='0';
    while strcmp(Response,'0')
        Response=urlread(Cmd3);
    end
    Graph(k,:)=str2num(urlread(Cmd4));
end
end
Graph=Graph';
surf(Graph)
shading flat
colormap jet
colorbar
ylim([0 400])
```

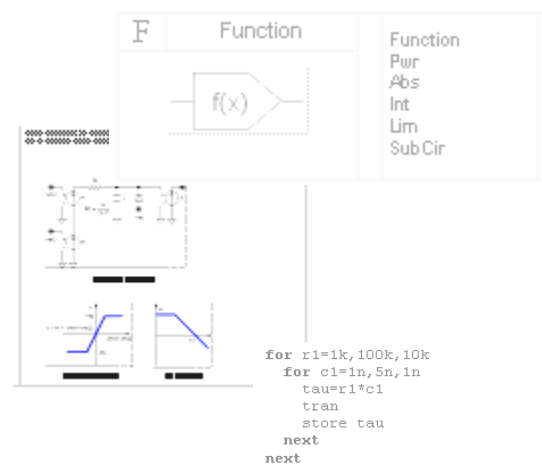
Программа изменяет сопротивление R1 в диапазоне 0.1...10 с логарифмическим шагом, вычисляет переходный процесс, считывает кривую V(out) и отображает V(out) в виде трехмерной поверхности, как функции времени и величины R1:



Файлы-примеры схемы и MATLAB программы находятся в директории Examples/MATLAB полного загрузочного пакета NL5 (zip-файл).

## VII. Attachments

Приложения, как содержащие сугубо техническую информацию, пока не переведены.



# 1. Component Types, Models and Parameters

Component types are described in the following format:

|               |             |                          |               |                                    |               |        |                              |
|---------------|-------------|--------------------------|---------------|------------------------------------|---------------|--------|------------------------------|
| <b>Letter</b> | <b>Type</b> |                          | <b>Models</b> | <b>Model 1<br/>Model 2<br/>...</b> | <b>Traces</b> | Symbol | Picture of<br>V, I, P traces |
| Symbol        |             |                          |               |                                    |               |        |                              |
| <b>Views</b>  | View 1      | View 2<br><br>(optional) | ...           | Description of views (optional)    |               |        |                              |

Model and parameters are presented as follows:

| Letter                         | Type  |        |                   | Examples/Components/Example file name |
|--------------------------------|---|--------|-------------------|---------------------------------------|
| Model                          | Parameter   | Units  | Description       |                                       |
| <b>Model name</b>              | Par 1   | Unit 1 | Par 1 description |                                       |
|                                | Par 2   | Unit 2 | Par 2 description |                                       |
|                                | ...   | ...    | ...               |                                       |
| Model specific view (optional) | Detailed description of the model and parameters. |        |                   |                                       |

Model example files are located in the `Examples/Components` folder of the NL5 complete package download zip file.

|  |  |        |   |   |
|--|--|--------|---|---|
|  | Label  |        |   |   |
|  | Label<br> | Models | Label<br>V<br>Pulse<br>Step<br>Sin<br>PWL<br>Function<br>File<br>SubCir | Traces  |
|  |  |        |   |  |

|  | Label     |       |                                     | Label/Label_Label.n15 |
|--|-----------|-------|-------------------------------------|-----------------------|
| Model  | Parameter | Units | Description                         |                       |
| Label  | VIC       | V     | Initial condition: initial voltage. |                       |
|  | R         | Ohm   | Initial resistance.                 |                       |
| <p><b>Label.</b> This model can be used:</p> <ul style="list-style-type: none"> <li>- As a voltage trace probe point.</li> <li>- For connecting schematic points without wires, including points at different sheets.</li> <li>- To apply initial voltage during DC operating point calculation.</li> </ul> <p>When calculating DC operating point, if initial voltage “VIC” is not blank, the temporary voltage source “VIC” is connected to the label through initial resistor “R”. When DC operating point is found, the voltage source is removed.</p> <p><b>Please note:</b> “VIC” is not a constant voltage source! This voltage will be removed after DC operating point calculation, and the label will be floating! For constant voltage source use <b>V</b> model.</p> |           |       |                                     |                       |

|       | Label  |       |             | Label/Label_V.n15 |
|-------|--|-------|-------------|-------------------|
| Model | Parameter                                      | Units | Description |                   |
| V     | V  | V     | Voltage.    |                   |
|       | <b>Constant voltage source.</b> Voltage = “V”. |       |             |                   |

| Label        |           | Label/Label_Pulse.n15 |                                  |
|--------------|-----------|-----------------------|----------------------------------|
| Model        | Parameter | Units                 | Description                      |
| <b>Pulse</b> | V1        | V                     | Pulse On voltage.                |
|              | V0        | V                     | Pulse Off voltage.               |
|              | Period    | s                     | Period.                          |
|              | Width     | s                     | Pulse width.                     |
|              | Slope     |                       | Slope type: Linear/Cos/Exp       |
|              | Rise      | s                     | Pulse rise length.               |
|              | Fall      | s                     | Pulse fall length.               |
|              | Delay     | s                     | Delay before first pulse starts. |

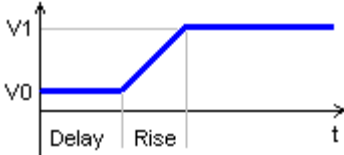
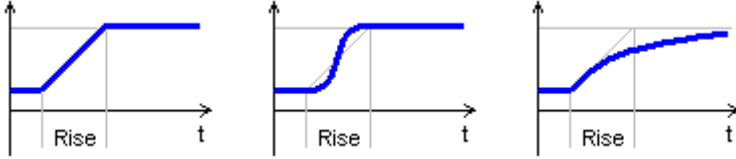
**Pulse voltage source.** Pulses start after “Delay” time. “Rise” time is included into “Width”, “Fall” time is not included into “Width”. Almost every parameter can be set to zero and infinity (“inf”); otherwise the error message will be displayed.

Slope type applies both to pulse rise and fall. The following slope types are available:

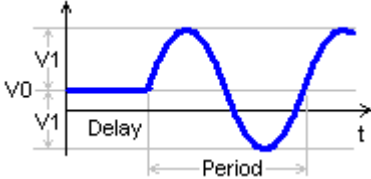
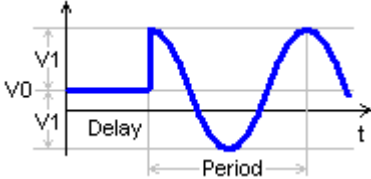
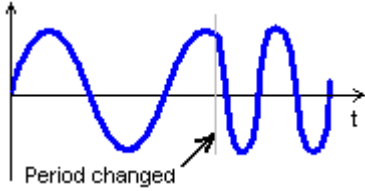
**Linear**

**Cos (cosine)**

**Exp (exponential)**

| Label   |           | Label/Label_Step.n15 |                            |
|---|-----------|----------------------|----------------------------|
| Model   | Parameter | Units                | Description                |
| <b>Step</b>   | V1        | V                    | Step On voltage.           |
|   | V0        | V                    | Step Off voltage.          |
|   | Slope     |                      | Slope type: Linear/Cos/Exp |
|   | Rise      | s                    | Step rise length.          |
|   | Delay     | s                    | Delay before step starts.  |
| <p><b>Step voltage source.</b> Step starts after “Delay” time.</p>  <p>The following slope types are available:</p>  <p style="text-align: center;"> <span>Linear</span>                      <b>Cos (cosine)</b>                      <b>Exp (exponential)</b> </p> |           |                      |                            |



| Label   |           |       | Label/Label_Sin.n15              |
|---|-----------|-------|----------------------------------|
| Model   | Parameter | Units | Description                      |
| Sin   | V1        | V     | Voltage amplitude.               |
|   | V0        | V     | Voltage baseline.                |
|   | Period    | s     | Period.                          |
|   | Phase     | deg   | Phase.                           |
|   | Delay     | s     | Delay before sine signal starts. |
| <p><b>Sine voltage source.</b> Sine signal starts after “Delay” time. “Phase” is sine phase in degrees at the moment when signal starts:</p> <div style="display: flex; justify-content: space-around;"> <div style="text-align: center;">  <p>Phase = 0</p> </div> <div style="text-align: center;">  <p>Phase = 90</p> </div> </div> <p>If transient is paused, sine period changed, then transient is continued, the phase of the signal remains continuous, providing smooth sine signal of variable frequency:</p>  |           |       |                                  |

| Label |   |       | Label/Label_PWL.n15       |
|-------|---|-------|---------------------------|
| Model | Parameter   | Units | Description               |
| PWL   | pwl   |       | Comma-separated string.   |
|       | Cycle   |       | Cycling (repeat): No/Yes. |
|       | Delay   | s     | Delay.                    |
|       | <p><b>Piece-wise linear voltage source.</b> Signal is defined by “pwl” parameter in the csv (“comma-separated values”) format, as follows:</p> $t_0, V_0, t_1, V_1, \dots, t_n, V_n$ <p>where all <math>t</math> and <math>V</math> can be numerical values or expressions. If <math>t &lt; t_0</math>, signal is <math>V_0</math>. If <math>t_0 &lt; t &lt; t_1</math>, signal value is linearly interpolated between <math>V_0</math> and <math>V_1</math>, etc. If <math>t &gt; t_n</math>, then signal value is <math>V_n</math> if “Cycle” parameter is set to “No”, otherwise signal defined in <math>t_0 \dots t_n</math> interval is repeated continuously. In addition, the whole signal is delayed by “Delay” time.</p> <p>Example:</p> $pwl = 0, 0, 1, 2, 4, 3, 5, 0, 8, 0$ <p>If “Cycle” = Yes, “Delay” = 0, the following voltage will be generated:</p> |       |                           |

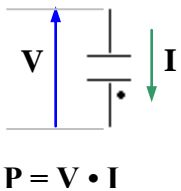

| Label    |   |       | Label/Label_Function.n15 |
|----------|---|-------|--------------------------|
| Model    | Parameter   | Units | Description              |
| Function | F(t)  | V     | Function                 |
|          | <p><b>Arbitrary function.</b> F(t) defines voltage as a function of the following variables:</p> <ul style="list-style-type: none"> <li><math>t</math> - current time</li> <li><math>V(name)</math> - voltage on the component <i>name</i></li> <li><math>I(name)</math> - current through the component <i>name</i></li> <li><math>P(name)</math> – power on the component <i>name</i></li> </ul> <p>where <i>name</i> is the name of any component in the schematic. If F(t) is blank, voltage is zero.</p> <p>Example:</p> $F(t) = \sin(t) * (1 + \cos(t * .01))$ $F(t) = V(R1) * I(R1)$ <p>Please note that <b>V</b>, <b>I</b>, and <b>P</b> variables are taken at previous calculation step. This may affect stability of the schematic with closed loop.</p> |       |                          |

| Label       |   |       | Label/Label_File.n15      |
|-------------|---|-------|---------------------------|
| Model       | Parameter   | Units | Description               |
| <b>File</b> | File  |       | File name.                |
|             | Cycle   |       | Cycling (repeat): No/Yes. |
|             | Delay   | s     | Delay.                    |
|             | <p><b>Voltage source defined in the text file.</b> "File" parameter is a file name, with full path to the file. If the file is located in the same directory as schematic file, the path can be omitted. Signal is defined in the csv ("comma-separated values") format, as follows:</p> <pre> &lt;if first line does not start with a number, it is ignored &gt; t0,V0 t1,V1 ..... tn,Vn                 </pre> <p>where all t and V can be numerical values or expressions. If <math>t &lt; t_0</math>, signal is <math>V_0</math>. If <math>t_0 &lt; t &lt; t_1</math>, signal value is linearly interpolated between <math>V_0</math> and <math>V_1</math>, etc. If <math>t &gt; t_n</math>, then signal value is <math>V_n</math> if "Cycle" parameter is set to "No", otherwise signal defined in <math>t_0 \dots t_n</math> interval is repeated continuously. In addition, the whole signal is delayed by "Delay" time.</p> <p>Example:</p> <pre> 0,0 1,2 4,3 5,0 8,0                 </pre> <p>If "Cycle" = Yes, "Delay" = 0, the following voltage will be generated:</p> |       |                           |

| Label  |           |       | Label/Label_SubCir.n15                      |
|--|-----------|-------|---|
| Model  | Parameter | Units | Description                                 |
| <b>SubCir</b>  | File      |       | File name of subcircuit schematic.          |
|  | Pin1      |       | Name of subcircuit label connected to pin 1 |
|  | Cmd       |       | Subcircuit start-up command string          |
|  | IC        |       | Subcircuit Initial conditions string        |
| <p><b>Subcircuit.</b> See <i>Working with Subcircuits</i> chapter for details.</p> |           |       |   |

|   |             |             |   |
|---|-------------|-------------|---|
| <b>A</b>  | Amperemeter | Amperemeter |   |
|  |             | Models      |  |

|             |   |                                  |
|-------------|---|----------------------------------|
| <b>A</b>    | Amperemeter   | A/ A_Amperemeter_Amperemeter.nl5 |
| Model       | No parameters   |                                  |
| Amperemeter | <p><b>Amperemeter.</b> Short circuit. In addition to current, amperemeter can measure voltage relative to ground, and power to grounded load.</p> |                                  |

|          |   |        |                                |        |   |
|----------|---|--------|--------------------------------|--------|---|
| <b>C</b> | <b>Capacitor</b>  | Models | C<br>PWL<br>Function<br>SubCir | Traces |  |
|          |  |        |                                |        |   |

|   |           |                     |  |  |
|---|-----------|---------------------|--|--|
| <b>C</b>  | Capacitor | C/C_Capacitor_C.n15 |  |  |
| Model   | Parameter | Units               | Description  |  |
| <b>C</b>  | C         | F                   | Capacitance  |  |
|   | IC        | V                   | Initial condition: voltage. Leave blank if IC not defined. |  |
| <p><b>Linear capacitor.</b> <math>I = C \cdot dV/dt</math>.</p> <p>When calculating DC operating point, if "IC" is defined, capacitor is replaced with voltage source equal to IC. If "IC" is not defined (blank), capacitor is temporarily removed (open circuit), DC operating point calculated, and then the voltage found across the capacitor is assigned to the capacitor as its initial voltage.</p> |           |                     |  |  |

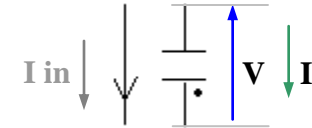
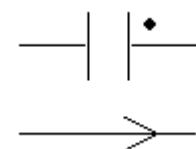
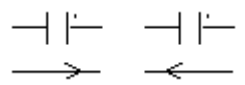
|  |           |                       |   |  |
|--|-----------|-----------------------|---|--|
| <b>C</b>   | Capacitor | C/C_Capacitor_PWL.n15 |   |  |
| Model  | Parameter | Units                 | Description   |  |
| <b>PWL</b>   | pwl       |                       | Comma-separated string, C(V)                              |  |
|  | IC        | V                     | Initial condition: voltage. Leave blank if no IC defined. |  |
| <p><b>Piece-wise linear capacitor.</b> "pwl" string defines capacitance as a function of voltage. See <i>Working with PWL model</i> chapter for details.</p> <p>When calculating DC operating point, if "IC" is defined, capacitor is replaced with voltage source equal to IC. If "IC" is not defined (blank), capacitor is temporarily removed (open circuit), DC operating point calculated, and then the voltage found across the capacitor is assigned to the capacitor as its initial voltage.</p> |           |                       |   |  |

| C        |   | Capacitor |  | C/C_Capacitor_Function.n15 |
|----------|---|-----------|--|----------------------------|
| Model    | Parameter   | Units     | Description                                    |                            |
| Function | Z(s)  | Ohm       | Impedance as a function of <b>s</b> parameter. |                            |
|          | <p><b>Impedance function in s domain.</b> For transient, constant impedance Z(0) is used. For linearized AC analysis, complex impedance Z(s) is used. The following variables can be used in the function:</p> <p><b>f</b> – current AC frequency, Hz<br/> <b>w</b> – angular AC frequency, <math>w = 2\pi f</math>.<br/> <b>s</b> or <b>p</b> – Laplace parameter, <math>s = p = j*2\pi f</math>.</p> <p>Example:<br/> <math>Z(s) = 1.0/3p/s + 1.5p</math> - 3 pF capacitor in series with 1.5 pH inductor.<br/> <math>Z(s) = (1.0/10n/s)  1k</math> - 10 nF capacitor in parallel with 1 kOhm resistor.</p> |           |  |                            |

| C  |           | Capacitor |   | C/C_Capacitor_SubCir.n15 |
|--|-----------|-----------|---|--------------------------|
| Model  | Parameter | Units     | Description                                 |                          |
| SubCir   | File      |           | File name of subcircuit schematic.          |                          |
|  | Pin1      |           | Name of subcircuit label connected to pin 1 |                          |
|  | Pin2      |           | Name of subcircuit label connected to pin 2 |                          |
|  | Cmd       |           | Subcircuit start-up command string          |                          |
|  | IC        |           | Subcircuit Initial conditions string        |                          |
| <p><b>Subcircuit.</b> See <i>Working with Subcircuits</i> chapter for details.</p> |           |           |   |                          |

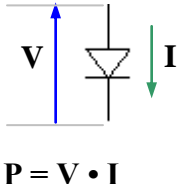
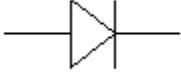
|          |                              |        |     |        |  |
|----------|------------------------------|--------|-----|--------|--|
| <b>C</b> | Voltage controlled capacitor | Models | PWL | Traces |  |
|          |                              |        |     |        |  |
| Views    |                              |        |     |        |  |

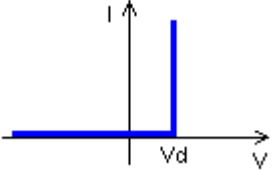
|  |                              |       |  |                 |
|--|------------------------------|-------|--|-----------------|
| <b>C</b>   | Voltage controlled capacitor |       |  | C/C_VCC_PWL.nl5 |
| Model  | Parameter                    | Units | Description  |                 |
| <b>PWL</b>   | pwl                          |       | Comma-separated string, C(Vin)                             |                 |
|  | IC                           | V     | Initial condition: voltage. Leave blank if IC not defined. |                 |
| <p><b>Piece-wise linear voltage controlled capacitor.</b> At any moment, capacitor is a <b>linear</b> capacitor. Its capacitance C is a function of control voltage:</p> $I = C(Vin) \cdot dV/dt.$ <p>“pwl” string defines capacitance as a function of control voltage <i>Vin</i>. See <i>Working with PWL model</i> chapter for details.</p> <p>When calculating DC operating point, if “IC” is defined, capacitor is replaced with voltage source equal to IC. If “IC” is not defined (blank), capacitor is temporarily removed (open circuit), DC operating point calculated, and then the voltage found across the capacitor is assigned to the capacitor as its initial voltage.</p> |                              |       |  |                 |

|          |   |        |     |        |   |
|----------|---|--------|-----|--------|---|
| <b>C</b> | Current controlled capacitor  | Models | PWL | Traces |  |
|          |  |        |     |        | $P = V \cdot I$   |
| Views    |  |        |     |        |   |

|  |                              |                 |  |  |
|--|------------------------------|-----------------|--|--|
| <b>C</b>   | Current controlled capacitor | C/C_CCC_PWL.n15 |  |  |
| Model  | Parameter                    | Units           | Description  |  |
| <b>PWL</b>   | pwl                          |                 | Comma-separated string, C( <i>I<sub>in</sub></i> )         |  |
|  | IC                           | V               | Initial condition: voltage. Leave blank if IC not defined. |  |
| <p><b>Piece-wise linear current controlled capacitor.</b> At any moment, capacitor is a <b>linear</b> capacitor. Its capacitance C is a function of control current:</p> $I = C(I_{in}) \cdot dV/dt.$ <p>“pwl” string defines capacitance as a function of control current <i>I<sub>in</sub></i>. See <i>Working with PWL model</i> chapter for details.</p> <p>When calculating DC operating point, if “IC” is defined, capacitor is replaced with voltage source equal to IC. If “IC” is not defined (blank), capacitor is temporarily removed (open circuit), DC operating point calculated, and then the voltage found across the capacitor is assigned to the capacitor as its initial voltage.</p> |                              |                 |  |  |



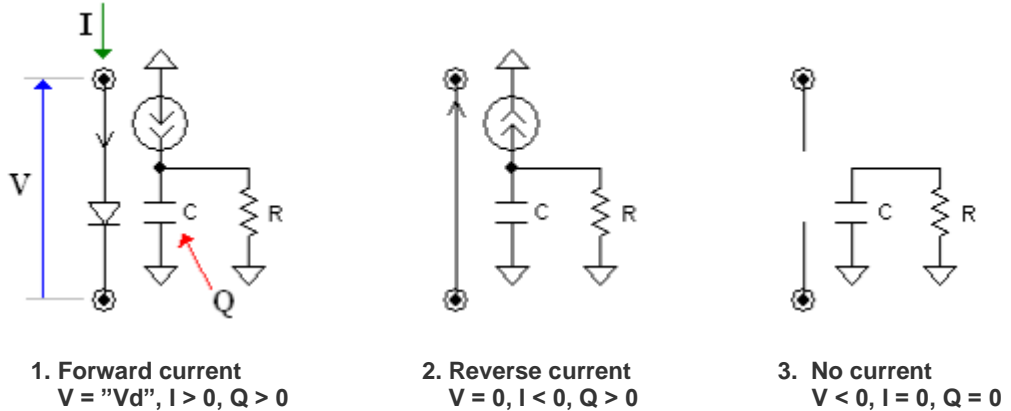
|   |       |   |   |
|---|-------|---|---|
| <b>D</b>  | Diode | <div style="writing-mode: vertical-rl; transform: rotate(180deg);">Models</div> Diode<br>Storage<br>Soft<br>PWL<br>SubCir | <div style="writing-mode: vertical-rl; transform: rotate(180deg);">Traces</div>  |
|  |       |   |   |

|   |           |                     |                            |
|---|-----------|---------------------|----------------------------|
| <b>D</b>  | Diode     | D/D_Diode_Diode.nl5 |                            |
| Model   | Parameter | Units               | Description                |
| <b>Diode</b>  | Vd        | V                   | Forward voltage drop.      |
|   | IC        |                     | Initial condition: On/Off. |
| <p><b>Ideal diode.</b> If <math>V \geq "Vd"</math>, diode is On (short circuit). Otherwise diode is Off (open circuit, <math>I=0</math>).</p>  <p>When calculating DC operating point diode is set to the state specified in "IC".</p> |           |                     |                            |

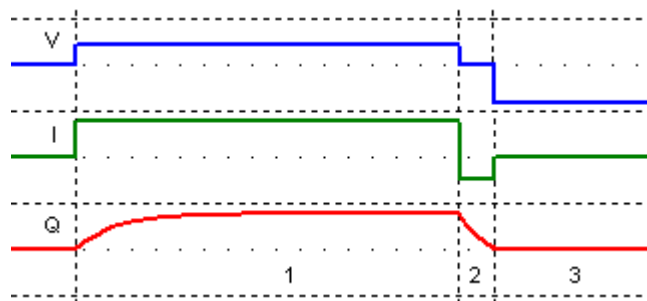
|          |       |                       |  |
|----------|-------|-----------------------|--|
| <b>D</b> | Diode | D/D_Diode_Storage.nl5 |  |
|----------|-------|-----------------------|--|

| Model   | Parameter | Units   | Description                  |
|---------|-----------|---------|------------------------------|
| Storage | Vd        | V       | Forward voltage drop.        |
|         | t         | s       | Recombination time constant. |
|         | IC        |         | Initial condition: Off/On.   |
|         | ICQ       | C (A*s) | Initial condition: charge.   |

**Charge storage diode.** Simplified equivalent schematic of the model is the following:



The diode has internal capacitor C and resistor R, with the time constant  $RC = "t"$ , Q is the charge on the capacitor. In mode 1, forward current flows through the diode and forward voltage drop is "Vd". At the same time, the current equal to forward current is charging capacitor C. In mode 2, reverse current is applied to the diode, and capacitor C is being discharged by the current equal to reverse current. As long as charge Q on the capacitor is positive, the diode is a short circuit with zero voltage drop. Finally, when charge drops to zero, the diode switches to mode 3, with zero current and negative voltage drop (open circuit). V, I, and Q waveforms are shown on the graph:



When calculating DC operating point the diode is set to the state specified in "IC", and internal charge Q is set to specified "ICQ" value.

| D Diode |           | D/D_Diode_Soft.nl5 |                              |
|---------|-----------|--------------------|------------------------------|
| Model   | Parameter | Units              | Description                  |
| Soft    | Vd        | V                  | Forward voltage drop.        |
|         | t         | s                  | Recombination time constant. |
|         | ts        | s                  | Soft recovery time constant. |
|         | IC        |                    | Initial condition: Off/On.   |
|         | ICQ       | C (A*s)            | Initial condition: charge.   |

**Soft recovery charge storage diode.** Simplified equivalent schematic of the model is the following:

**1. Forward current**  
 $V = "Vd", I > 0, Q > 0$   
 $C = 1, R = "t"$

**2. Reverse current**  
 $V = "Vd", I < 0, Q > 0$   
 $C = 1, R = "t"$

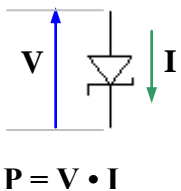
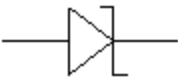
**3. Soft recovery**  
 $V < "Vd", I < 0, Q > 0$   
 $C = 1, R = "ts"$

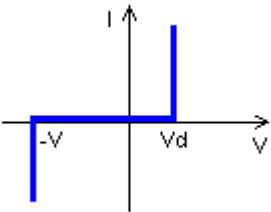
The diode has internal capacitor  $C=1$  and resistor  $R$ . Time constant  $RC$  is equal either recombination time constant  $"t"$ , or soft recovery time constant  $"ts"$ .  $Q$  is the charge on the capacitor. In mode 1, forward current flows through the diode and forward voltage drop is  $"Vd"$ . At the same time, the current equal to forward current is charging capacitor  $C$ . In mode 2, reverse current is applied to the diode, and capacitor  $C$  is being discharged by the current equal to reverse current. Voltage drop on the diode is still  $"Vd"$ . At the moment when reverse current is equal or less than charge divided by soft recovery time constant  $"ts"$ , a mode 3 is turned on. In mode 3, capacitor  $C$  is being exponentially discharged by the current through resistor  $R$  with time constant  $"ts"$  (plus small constant current to ensure full discharge - not shown on the picture). Reverse diode current is proportional to the charge. As soon as charge drops to zero, the diode switches to mode 4 (not shown), with zero current and negative voltage drop (open circuit). See demo schematic D/D\_Diode\_Soft.nl5 for examples of soft recovery waveforms.

When calculating DC operating point the diode is set to the state specified in  $"IC"$ , and internal charge  $Q$  is set to specified  $"ICQ"$  value.

| <b>D</b> |  | Diode |                              | D/D_Diode_PWL.n15 |
|----------|--|-------|------------------------------|-------------------|
| Model    | Parameter  | Units | Description                  |                   |
| PWL      | pwl  |       | Comma-separated string, R(V) |                   |
|          | <b>Piece-wise linear diode.</b> “pwl” string defines resistance as a function of voltage. See <i>Working with PWL model</i> chapter for details. |       |                              |                   |

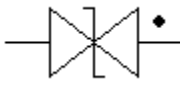
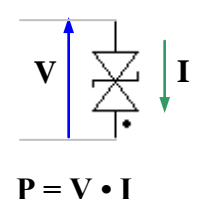
| <b>D</b>  |           | Diode |   | D/D_Diode_SubCir.n15 |
|---|-----------|-------|---|----------------------|
| Model   | Parameter | Units | Description                                 |                      |
| SubCir  | File      |       | File name of subcircuit schematic.          |                      |
|   | Pin1      |       | Name of subcircuit label connected to pin 1 |                      |
|   | Pin2      |       | Name of subcircuit label connected to pin 2 |                      |
|   | Cmd       |       | Subcircuit start-up command string          |                      |
|   | IC        |       | Subcircuit Initial conditions string        |                      |
| <b>Subcircuit.</b> See <i>Working with Subcircuits</i> chapter for details. |           |       |   |                      |

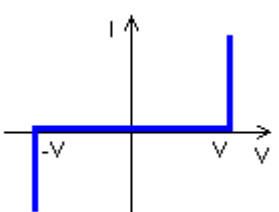
|          |   |        |                        |        |   |
|----------|---|--------|------------------------|--------|---|
| <b>D</b> | <b>Zener</b>  | Models | Zener<br>PWL<br>SubCir | Traces |  |
|          |  |        |                        |        |   |

|   |           |                     |                                    |  |
|---|-----------|---------------------|------------------------------------|--|
| <b>D</b>  | Zener     | D/D_Zener_Zener.n15 |                                    |  |
| Model   | Parameter | Units               | Description                        |  |
| <b>Zener</b>  | V         | V                   | Breakdown voltage drop.            |  |
|   | Vd        | Vd                  | Forward voltage drop.              |  |
|   | IC        |                     | Initial condition: Minus/Off/Plus. |  |
| <p><b>Ideal zener.</b> If <math>V \leq -"V"</math> or <math>V \geq "Vd"</math>, zener is On (short circuit). Otherwise zener is Off (open circuit, <math>I=0</math>).</p> |           |                     |                                    |  |
|   |           |                     |                                    |  |
| <p>When calculating DC operating point zener is set to the state specified in "IC".</p>   |           |                     |                                    |  |

|            |   |                   |                              |  |
|------------|---|-------------------|------------------------------|--|
| <b>D</b>   | Zener   | D/D_Zener_PWL.n15 |                              |  |
| Model      | Parameter   | Units             | Description                  |  |
| <b>PWL</b> | pwl   |                   | Comma-separated string, R(V) |  |
|            | <p><b>Piece-wise linear zener.</b> "pwl" string defines resistance as a function of voltage. See <i>Working with PWL model</i> chapter for details.</p> |                   |                              |  |


|  |           |                      |   |  |
|--|-----------|----------------------|---|--|
| <b>D</b>   | Zener     | D/D_Zener_SubCir.n15 |   |  |
| Model  | Parameter | Units                | Description                                 |  |
| <b>SubCir</b>  | File      |                      | File name of subcircuit schematic.          |  |
|  | Pin1      |                      | Name of subcircuit label connected to pin 1 |  |
|  | Pin2      |                      | Name of subcircuit label connected to pin 2 |  |
|  | Cmd       |                      | Subcircuit start-up command string          |  |
|  | IC        |                      | Subcircuit Initial conditions string        |  |
| <p><b>Subcircuit.</b> See <i>Working with Subcircuits</i> chapter for details.</p> |           |                      |   |  |

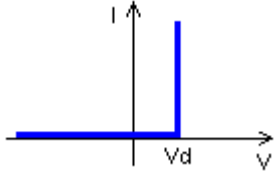
|          |                     |        |                        |        |   |   |
|----------|---------------------|--------|------------------------|--------|---|---|
| <b>D</b> | Bidirectional zener | Models | Zener<br>PWL<br>SubCir | Traces |  |  |
|          |                     |        |                        |        |   |   |

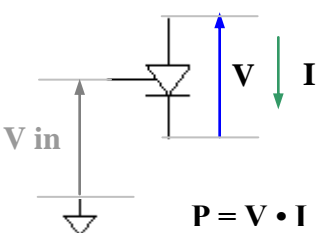
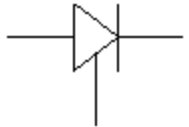
|  |                     |                      |                                    |  |
|--|---------------------|----------------------|------------------------------------|--|
| <b>D</b>   | Bidirectional zener | D/D_BZener_Zener.n15 |                                    |  |
| Model  | Parameter           | Units                | Description                        |  |
| Zener  | V                   | V                    | Breakdown voltage drop.            |  |
|  | IC                  |                      | Initial condition: Minus/Off/Plus. |  |
| <p><b>Ideal bidirectional zener.</b> If <math>V \leq -V</math> or <math>V \geq V</math>, zener is On (short circuit). Otherwise zener is Off (open circuit, <math>I=0</math>).</p> |                     |                      |                                    |  |
|    |                     |                      |                                    |  |
| <p>When calculating DC operating point zener is set to the state specified in "IC".</p>  |                     |                      |                                    |  |

|          |   |                    |                              |  |
|----------|---|--------------------|------------------------------|--|
| <b>D</b> | Bidirectional zener   | D/D_BZener_PWL.n15 |                              |  |
| Model    | Parameter   | Units              | Description                  |  |
| PWL      | pwl   |                    | Comma-separated string, R(V) |  |
|          | <p><b>Piece-wise linear zener.</b> "pwl" string defines resistance as a function of voltage. See <i>Working with PWL model</i> chapter for details.</p> |                    |                              |  |

|  |                     |                       |   |  |
|--|---------------------|-----------------------|---|--|
| <b>D</b>   | Bidirectional zener | D/D_BZener_SubCir.n15 |   |  |
| Model  | Parameter           | Units                 | Description                                 |  |
| SubCir   | File                |                       | File name of subcircuit schematic.          |  |
|  | Pin1                |                       | Name of subcircuit label connected to pin 1 |  |
|  | Pin2                |                       | Name of subcircuit label connected to pin 2 |  |
|  | Cmd                 |                       | Subcircuit start-up command string          |  |
|  | IC                  |                       | Subcircuit Initial conditions string        |  |
| <p><b>Subcircuit.</b> See <i>Working with Subcircuits</i> chapter for details.</p> |                     |                       |   |  |

|          |   |        |       |        |   |
|----------|---|--------|-------|--------|---|
| <b>D</b> | Bridge rectifier  | Models | Diode | Traces | $P = V_1 \cdot I_1 + V_2 \cdot I_2 + V_3 \cdot I_3 + V_4 \cdot I_4$ |
|          |  |        |       |        |   |

|          |  |                      |                       |  |
|----------|--|----------------------|-----------------------|--|
| <b>D</b> | Bridge rectifier   | D/D_Bridge_Diode.nl5 |                       |  |
| Model    | Parameter  | Units                | Description           |  |
| Diode    | Vd   | V                    | Forward voltage drop. |  |
|          | <p><b>Bridge rectifier with ideal diodes.</b> For each diode, if <math>V \geq "Vd"</math>, diode is On (short circuit). Otherwise diode is Off (open circuit, <math>I=0</math>).</p> <div style="text-align: center;">  </div> <p>When calculating DC operating point all diodes are Off.</p> |                      |                       |  |

|          |   |        |                     |        |   |
|----------|---|--------|---------------------|--------|---|
| <b>D</b> | Logic controlled thyristor  | Models | Thyristor<br>SubCir | Traces |  |
|          |  |        |                     |        |   |

| <b>D</b>   | Logic controlled thyristor | D/D_LCT_Thyristor.n15 |                            |  |
|--|----------------------------|-----------------------|----------------------------|--|
| Model  | Parameter                  | Units                 | Description                |  |
| <b>Thyristor</b>   | Vd                         | V                     | Forward voltage drop.      |  |
|  | Ihold                      | A                     | Holding current.           |  |
|  | IC                         |                       | Initial condition: Off/On. |  |
| <p><b>Ideal thyristor.</b> Thyristor has two states:</p> <ul style="list-style-type: none"> <li>- Off state (non-conducting): open circuit.</li> <li>- On state (conducting): ideal diode with “Vd” forward voltage drop.</li> </ul> <p>If control voltage “Vin” is greater than logical threshold, thyristor is in On state (ideal diode). When control voltage drops below logical threshold, thyristor stays in On state as long as current I exceeds holding current “Ihold”, and voltage V is not negative.</p> <p>To see and set logical levels and threshold go to <b>Transient   Settings</b>, or <b>AC   Settings</b>, then click <b>Advanced</b> button.</p> <p>When calculating DC operating point thyristor is set to the state specified in “IC”.</p> |                            |                       |                            |  |

| <b>D</b>   | Logic controlled thyristor | D/D_LCT_SubCir.n15 |   |  |
|--|----------------------------|--------------------|---|--|
| Model  | Parameter                  | Units              | Description                                 |  |
| <b>SubCir</b>  | File                       |                    | File name of subcircuit schematic.          |  |
|  | Pin1                       |                    | Name of subcircuit label connected to pin 1 |  |
|  | Pin2                       |                    | Name of subcircuit label connected to pin 2 |  |
|  | Pin3                       |                    | Name of subcircuit label connected to pin 3 |  |
|  | Pin4                       |                    | Name of subcircuit label connected to pin 4 |  |
|  | Cmd                        |                    | Subcircuit start-up command string          |  |
|  | IC                         |                    | Subcircuit Initial conditions string        |  |
| <p><b>Subcircuit.</b> See <i>Working with Subcircuits</i> chapter for details.</p> |                            |                    |   |  |



|          |                              |        |                  |        |                 |
|----------|------------------------------|--------|------------------|--------|-----------------|
| <b>D</b> | Voltage controlled thyristor | Models | Thyristor SubCir | Traces |                 |
|          |                              |        |                  |        |                 |
| Views    |                              |        |                  |        | $P = V \cdot I$ |

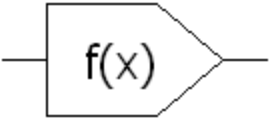
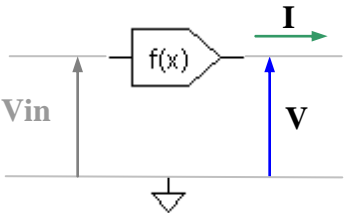
|  |                              |                       |                            |  |
|--|------------------------------|-----------------------|----------------------------|--|
| <b>D</b>   | Voltage controlled thyristor | D/D_VCT_Thyristor.nl5 |                            |  |
| Model  | Parameter                    | Units                 | Description                |  |
| <b>Thyristor</b>   | Vd                           | V                     | Forward voltage drop.      |  |
|  | Ihold                        | A                     | Holding current.           |  |
|  | Threshold                    | V                     | Voltage threshold.         |  |
|  | IC                           |                       | Initial condition: Off/On. |  |
| <p><b>Ideal thyristor.</b> Thyristor has two states:</p> <ul style="list-style-type: none"> <li>- Off state (non-conducting): open circuit.</li> <li>- On state (conducting): ideal diode with “Vd” forward voltage drop.</li> </ul> <p>If control voltage “Vin” is greater than “Threshold”, thyristor is in On state (ideal diode). When control voltage drops below “Threshold”, thyristor stays in On state as long as current I exceeds holding current “Ihold”, and voltage V is not negative.</p> <p>When calculating DC operating point thyristor is set to the state specified in “IC”.</p> |                              |                       |                            |  |

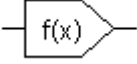
|  |                              |                    |   |  |
|--|------------------------------|--------------------|---|--|
| <b>D</b>   | Voltage controlled thyristor | D/D_VCT_SubCir.nl5 |   |  |
| Model  | Parameter                    | Units              | Description                                 |  |
| <b>SubCir</b>  | File                         |                    | File name of subcircuit schematic.          |  |
|  | Pin1                         |                    | Name of subcircuit label connected to pin 1 |  |
|  | Pin2                         |                    | Name of subcircuit label connected to pin 2 |  |
|  | Pin3                         |                    | Name of subcircuit label connected to pin 3 |  |
|  | Pin4                         |                    | Name of subcircuit label connected to pin 4 |  |
|  | Cmd                          |                    | Subcircuit start-up command string          |  |
|  | IC                           |                    | Subcircuit Initial conditions string        |  |
| <p><b>Subcircuit.</b> See <i>Working with Subcircuits</i> chapter for details.</p> |                              |                    |   |  |


|          |                              |        |                  |        |  |
|----------|------------------------------|--------|------------------|--------|--|
| <b>D</b> | Current controlled thyristor | Models | Thyristor SubCir | Traces |  |
|          |                              |        |                  |        |  |
| Views    |                              |        |                  |        |  |


|  |                              |                       |                            |  |
|--|------------------------------|-----------------------|----------------------------|--|
| <b>D</b>   | Current controlled thyristor | D/D_CCT_Thyristor.nl5 |                            |  |
| Model  | Parameter                    | Units                 | Description                |  |
| <b>Thyristor</b>   | Vd                           | V                     | Forward voltage drop.      |  |
|  | Ihold                        | A                     | Holding current.           |  |
|  | Threshold                    | A                     | Current threshold.         |  |
|  | IC                           |                       | Initial condition: Off/On. |  |
| <p><b>Ideal thyristor.</b> Thyristor has two states:</p> <ul style="list-style-type: none"> <li>- Off state (non-conducting): open circuit.</li> <li>- On state (conducting): ideal diode with “Vd” forward voltage drop.</li> </ul> <p>If control current “Iin” is greater than “Threshold”, thyristor is in On state (ideal diode). When control current drops below “Threshold”, thyristor stays in On state as long as current I exceeds holding current “Ihold”, and voltage V is not negative.</p> <p>When calculating DC operating point thyristor is set to the state specified in “IC”.</p> |                              |                       |                            |  |


|  |                              |                    |   |  |
|--|------------------------------|--------------------|---|--|
| <b>D</b>   | Current controlled thyristor | D/D_CCT_SubCir.nl5 |   |  |
| Model  | Parameter                    | Units              | Description                                 |  |
| <b>SubCir</b>  | File                         |                    | File name of subcircuit schematic.          |  |
|  | Pin1                         |                    | Name of subcircuit label connected to pin 1 |  |
|  | Pin2                         |                    | Name of subcircuit label connected to pin 2 |  |
|  | Pin3                         |                    | Name of subcircuit label connected to pin 3 |  |
|  | Pin4                         |                    | Name of subcircuit label connected to pin 4 |  |
|  | Cmd                          |                    | Subcircuit start-up command string          |  |
|  | IC                           |                    | Subcircuit Initial conditions string        |  |
| <p><b>Subcircuit.</b> See <i>Working with Subcircuits</i> chapter for details.</p> |                              |                    |   |  |


| F | Function  |   |  |
|---|---|---|--|
|   |  | Models<br>Function<br>Pwr<br>Abs<br>Int<br>Lim<br>Table<br>SubCir | <div style="text-align: center;">  </div> <p><math>P = V \cdot I</math></p> |


| F   | Function  |       |  | F/F_Function_Function.nl5 |
|---|-----------|-------|--|---------------------------|
| Model   | Parameter | Units | Description                              |                           |
| <b>Function</b><br>  | F(x)      | V     | Output as function of the input.         |                           |
|   | F(s)      |       | AC transfer function in <b>s</b> domain. |                           |
|   | IC        | V     | Initial condition: output voltage.       |                           |
| <p><b>Arbitrary function.</b></p> <p><b>Transient analysis.</b> F(x) defines output voltage as a function of the following variables:</p> <ul style="list-style-type: none"> <li><b>x</b> – input voltage <math>V_{in}</math></li> <li><b>t</b> - current time</li> <li><b>V(name)</b> - voltage on the component <b>name</b></li> <li><b>I(name)</b> - current through the component <b>name</b></li> <li><b>P(name)</b> – power on the component <b>name</b></li> </ul> <p>where <b>name</b> is the name of any component in the schematic. If F(x) is blank, output is zero. F(s) is ignored.</p> <p>Example:</p> <ul style="list-style-type: none"> <li>F(x) = <math>x^3</math></li> <li>F(x) = <math>x * \sin(t)</math></li> <li>F(x) = <math>P(r1)+P(r2)</math></li> </ul> <p>Please note that input voltage <b>x</b> and variables <b>V</b>, <b>I</b>, and <b>P</b> are taken at previous calculation step. This may affect stability of the schematic with closed loop.</p> <p><b>AC analysis.</b> F(s) defines transfer function in <b>s</b> domain. Only operators and functions that support complex numbers can be used in this function. The following variables can be used in the function:</p> <ul style="list-style-type: none"> <li><b>f</b> – current AC frequency, Hz</li> <li><b>w</b> – angular AC frequency, <math>w = 2\pi f</math>.</li> <li><b>s</b> or <b>p</b> – Laplace parameter, <math>s = p = j*2\pi f</math>.</li> </ul> <p>Example:</p> <ul style="list-style-type: none"> <li>F(s) = <math>1/(1+s)</math></li> <li>F(s) = <math>\exp(-1mk*s)</math></li> </ul> <p>F(s) is calculated at each frequency. If F(s) is blank, it is assumed to be 1. Also, if F(x) is not blank, it is linearized at DC operating point, and F(s) is multiplied by linearized gain at each frequency.</p> <p>When calculating DC operating point for transient or AC analysis, output is set to specified output voltage "IC".</p> |           |       |  |                           |

| F  |           | Function |                                    | F/F_Function_Pwr.n15 |
|--|-----------|----------|------------------------------------|----------------------|
| Model  | Parameter | Units    | Description                        |                      |
| <b>Pwr</b><br>  | power     |          | Power.                             |                      |
|  | K         | V/V      | Gain.                              |                      |
|  | IC        | V        | Initial condition: output voltage. |                      |
| <p><b>“Signed” power function.</b> <math>V = K * \text{pwr}( \text{Vin}, \text{power} )</math>.</p> <p>The function is calculated as follows:</p> <p>if power = 0:</p> <ul style="list-style-type: none"> <li>if <math>\text{Vin} &lt; 0</math> . . . : <math>V = -K</math></li> <li>if <math>\text{Vin} = 0</math> . . . : <math>V = 0</math></li> <li>if <math>\text{Vin} &gt; 0</math> . . . : <math>V = K</math></li> </ul> <p>if power <math>\neq 0</math>:</p> <ul style="list-style-type: none"> <li>if <math>\text{Vin} &lt; 0</math> . . . : <math>V = -K * (-\text{Vin})^{\text{power}}</math></li> <li>if <math>\text{Vin} = 0</math> . . . : <math>V = 0</math></li> <li>if <math>\text{Vin} &gt; 0</math> . . . : <math>V(= K * \text{Vin}^{\text{power}}</math></li> </ul> <p>When calculating DC operating point output is set to specified output voltage “IC”. When calculating transient, output voltage is always delayed by one calculation step. This may affect stability of the schematic with closed loop.</p> |           |          |                                    |                      |

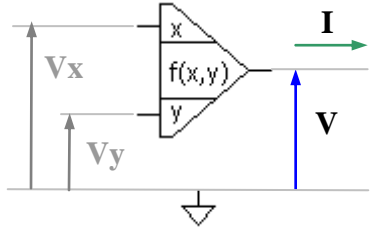
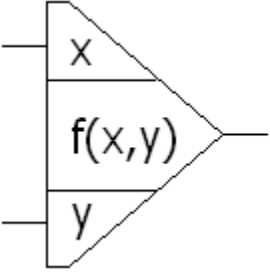
| F  |           | Function |                                    | F/F_Function_Abs.n15 |
|--|-----------|----------|------------------------------------|----------------------|
| Model  | Parameter | Units    | Description                        |                      |
| <b>Abs</b><br>  | K         | V/V      | Gain.                              |                      |
|  | IC        | V        | Initial condition: output voltage. |                      |
| <p><b>Absolute value.</b> <math>V = K * \text{abs}( \text{Vin} )</math>.</p> <p>When calculating DC operating point output is set to specified output voltage “IC”. When calculating transient, output voltage is always delayed by one calculation step. This may affect stability of the schematic with closed loop.</p> |           |          |                                    |                      |

| F   |   | Function |                                    | F/F_Function_Int.n15 |
|---|---|----------|------------------------------------|----------------------|
| Model   | Parameter   | Units    | Description                        |                      |
| <b>Int</b><br> | resolution  | V        | Resolution.                        |                      |
|   | K   | V/V      | Gain.                              |                      |
|   | IC  | V        | Initial condition: output voltage. |                      |
|   | <p><b>Rounding function.</b> <math>V = K * \text{round}(V_{in}, \text{resolution})</math>.</p> <p>Round to the nearest multiple of “resolution”. If resolution = 1, round to the nearest integer.</p> <p>When calculating DC operating point output is set to specified output voltage “IC”. When calculating transient, output voltage is always delayed by one calculation step. This may affect stability of the schematic with closed loop.</p> |          |                                    |                      |

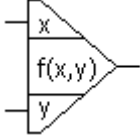
| F   |  | Function |                                    | F/F_Function_Lim.n15 |
|---|--|----------|------------------------------------|----------------------|
| Model   | Parameter  | Units    | Description                        |                      |
| <b>Lim</b><br> | Max  | V        | Maximum.                           |                      |
|   | Min  | V        | Minimum.                           |                      |
|   | IC   | V        | Initial condition: output voltage. |                      |
|   | <p><b>Limiting function.</b> The function is calculated as follows:</p> <p>if <math>V_{in} &lt; \text{Min}</math> . . . : <math>V = \text{Min}</math><br/> if <math>V_{in} &gt; \text{Max}</math> . . . : <math>V = \text{Max}</math><br/> Otherwise . . . . . : <math>V = V_{in}</math></p> <p>When calculating DC operating point output is set to specified output voltage “IC”. When calculating transient, output voltage is always delayed by one calculation step. This may affect stability of the schematic with closed loop.</p> |          |                                    |                      |

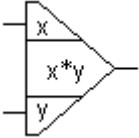
| F  | Function  |       |   | F/F_Function_Table.nl5 |
|--|-----------|-------|---|------------------------|
| Model  | Parameter | Units | Description                             |                        |
| <b>Table</b><br>  | Table     |       | Comma-separated string, Vin/Vout pairs. |                        |
|  | IC        | V     | Initial condition: output voltage.      |                        |
| <p><b>Look-up table.</b> Function output is defined by “Table” parameter in the csv (“comma separated values”) format, as follows:</p> $X1,Y1,X2,Y2,\dots,XN,YN$ <p>where <math>X_i, Y_i</math> pair defines input value (X) and output value (Y). Output value between specified points is linearly interpolated. Output value below <math>X_1</math> is linearly extrapolated using <math>X_1 \dots X_2</math> interval data, output value above <math>X_N</math> is linearly extrapolated using <math>X_{(N-1)} \dots X_N</math> interval data. Values <math>X_1 \dots X_N</math> should be given in an ascending order.</p> <p>See <i>Working with Table model</i> chapter for more details.</p> <p>When calculating DC operating point output is set to specified output voltage “IC”. When calculating transient, output voltage is always delayed by one calculation step. This may affect stability of the schematic with closed loop.</p> |           |       |   |                        |

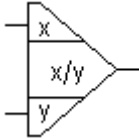
| F  | Function  |       |   | F/F_Function_SubCir.nl5 |
|--|-----------|-------|---|-------------------------|
| Model  | Parameter | Units | Description                                 |                         |
| <b>SubCir</b>  | File      |       | File name of subcircuit schematic.          |                         |
|  | Pin1      |       | Name of subcircuit label connected to pin 1 |                         |
|  | Pin2      |       | Name of subcircuit label connected to pin 2 |                         |
|  | Cmd       |       | Subcircuit start-up command string          |                         |
|  | IC        |       | Subcircuit Initial conditions string        |                         |
| <p><b>Subcircuit.</b> See <i>Working with Subcircuits</i> chapter for details.</p> |           |       |   |                         |

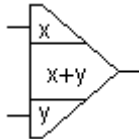
|   |                   |  |   |
|---|-------------------|--|---|
| <b>F</b>  | <b>Function-2</b> | Models<br>Function<br>Mul<br>Div<br>Sum<br>Sub<br>Max<br>Min<br>GT<br>LT<br>Pwr<br>Mag<br>Phase<br>Table<br>SubCir | Traces<br> <p><b>P = V • I</b></p> |
|  |                   |  |   |

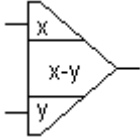


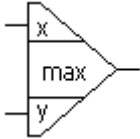
| F  | Function-2   |       | F/F_Function-2_Function.nl5              |
|--|--|-------|--|
| Model  | Parameter  | Units | Description                              |
| <b>Function</b><br> | F(x,y)   | V     | Output as function of the inputs.        |
|  | F(s)   |       | AC transfer function in <b>s</b> domain. |
|  | IC   | V     | Initial condition: output voltage.       |
|  | <p><b>Arbitrary function.</b></p> <p><b>Transient analysis.</b> F(x,y) defines output voltage as a function of the following variables:</p> <p><b>x</b> – input voltage V<sub>x</sub><br/> <b>y</b> – input voltage V<sub>y</sub><br/> <b>t</b> - current time<br/> <b>V(name)</b> - voltage on the component <b>name</b><br/> <b>I(name)</b> - current through the component <b>name</b><br/> <b>P(name)</b> – power on the component <b>name</b></p> <p>where <b>name</b> is the name of any component in the schematic. If F(x,y) is blank, output is zero. F(s) is ignored.</p> <p>Example:<br/> <math>F(x,y) = \sqrt{x*x+y*y}</math><br/> <math>F(x,y) = x * y * \sin(t)</math><br/> <math>F(x,y) = P(r1)+P(r2)</math></p> <p>Please note that input voltages <b>x</b>, <b>y</b>, and variables <b>V</b>, <b>I</b>, <b>P</b> are taken at previous calculation step. This may affect stability of the schematic with closed loop.</p> <p><b>AC analysis.</b> F(s) defines transfer function in <b>s</b> domain. Only operators and functions that support complex numbers can be used in this function. The following variables can be used in the function:<br/> <b>f</b> – current AC frequency, Hz<br/> <b>w</b> – angular AC frequency, <math>w = 2\pi f</math> .<br/> <b>s</b> or <b>p</b> – Laplace parameter, <math>s = p = j*2\pi f</math>.</p> <p>Example:<br/> <math>F(s) = 1/(1+s)</math><br/> <math>F(s) = \exp(-1mk*s)</math></p> <p>F(s) is calculated at each frequency. If F(s) is blank, it is assumed to be 1. Also, if F(x,y) is not blank, it is linearized at DC operating point, and F(s) is multiplied by linearized gain..</p> <p>When calculating DC operating point for transient or AC analysis, output is set to specified output voltage “IC”.</p> |       |  |

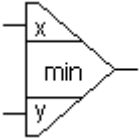
| F   |   | Function-2 |                                    | F/F_Function-2_Mul.n15 |
|---|---|------------|------------------------------------|------------------------|
| Model   | Parameter   | Units      | Description                        |                        |
| <b>Mul</b><br> | K   | V/V        | Gain.                              |                        |
|   | IC  | V          | Initial condition: output voltage. |                        |
|   | <p><b>Multiplication.</b> <math>V = K * V_x * V_y</math>.</p> <p>When calculating DC operating point output is set to specified output voltage "IC". When calculating transient, output voltage is always delayed by one calculation step. This may affect stability of the schematic with closed loop.</p> |            |                                    |                        |

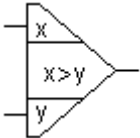
| F   |   | Function-2 |                                    | F/F_Function-2_Div.n15 |
|---|---|------------|------------------------------------|------------------------|
| Model   | Parameter   | Units      | Description                        |                        |
| <b>Div</b><br> | K   | V/V        | Gain.                              |                        |
|   | IC  | V          | Initial condition: output voltage. |                        |
|   | <p><b>Division.</b> <math>V = K * V_x / V_y</math>.</p> <p>If <math>V_y = 0</math>, <math>V = 0</math>.</p> <p>When calculating DC operating point output is set to specified output voltage "IC". When calculating transient, output voltage is always delayed by one calculation step. This may affect stability of the schematic with closed loop.</p> |            |                                    |                        |

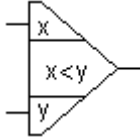
| F   |   | Function-2 |                                    | F/F_Function-2_Sum.n15 |
|---|---|------------|------------------------------------|------------------------|
| Model   | Parameter   | Units      | Description                        |                        |
| <b>Sum</b><br> | K   | V/V        | Gain.                              |                        |
|   | IC  | V          | Initial condition: output voltage. |                        |
|   | <p><b>Addition.</b> <math>V = K * (V_x + V_y)</math>.</p> <p>When calculating DC operating point output is set to specified output voltage "IC". When calculating transient, output voltage is always delayed by one calculation step. This may affect stability of the schematic with closed loop.</p> |            |                                    |                        |

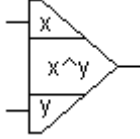
| F  |           | Function-2 |                                    | F/F_Function-2_Sub.n15 |
|--|-----------|------------|------------------------------------|------------------------|
| Model  | Parameter | Units      | Description                        |                        |
| <b>Sub</b><br>  | K         | V/V        | Gain.                              |                        |
|  | IC        | V          | Initial condition: output voltage. |                        |
| <p><b>Subtraction.</b> <math>V = K * (Vx - Vy)</math>.</p> <p>When calculating DC operating point output is set to specified output voltage “IC”. When calculating transient, output voltage is always delayed by one calculation step. This may affect stability of the schematic with closed loop.</p> |           |            |                                    |                        |

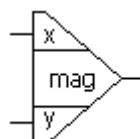
| F   |           | Function-2 |                                    | F/F_Function-2_Max.n15 |
|---|-----------|------------|------------------------------------|------------------------|
| Model   | Parameter | Units      | Description                        |                        |
| <b>Max</b><br>   | K         | V/V        | Gain.                              |                        |
|   | IC        | V          | Initial condition: output voltage. |                        |
| <p><b>Maximum.</b> <math>V = K * \max(Vx, Vy)</math>.</p> <p>if <math>Vx \geq Vy</math> . . . : <math>V = K * Vx</math><br/> if <math>Vx &lt; Vy</math> . . . : <math>V = K * Vy</math></p> <p>When calculating DC operating point output is set to specified output voltage “IC”. When calculating transient, output voltage is always delayed by one calculation step. This may affect stability of the schematic with closed loop.</p> |           |            |                                    |                        |

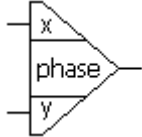
| F   |           | Function-2 |                                    | F/F_Function-2_Min.n15 |
|---|-----------|------------|------------------------------------|------------------------|
| Model   | Parameter | Units      | Description                        |                        |
| <b>Min</b><br>   | K         | V/V        | Gain.                              |                        |
|   | IC        | V          | Initial condition: output voltage. |                        |
| <p><b>Minimum.</b> <math>V = K * \min(Vx, Vy)</math>.</p> <p>if <math>Vx \geq Vy</math> . . . : <math>V = K * Vy</math><br/> if <math>Vx &lt; Vy</math> . . . : <math>V = K * Vx</math></p> <p>When calculating DC operating point output is set to specified output voltage “IC”. When calculating transient, output voltage is always delayed by one calculation step. This may affect stability of the schematic with closed loop.</p> |           |            |                                    |                        |

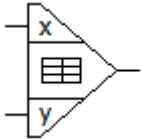
| F   |  | Function-2 |                                    | F/F_Function-2_GT.n15 |
|---|--|------------|------------------------------------|-----------------------|
| Model   | Parameter  | Units      | Description                        |                       |
| GT  | IC   | V          | Initial condition: output voltage. |                       |
|  | <p><b>Greater than.</b> <math>V = Vx &gt; Vy</math> ? High : Low.</p> <p>if <math>Vx \leq Vy</math> . . . : <math>V = \text{Low}</math><br/> if <math>Vx &gt; Vy</math> . . . : <math>V = \text{High}</math></p> <p>High and Low are logical levels. To see and set logical levels go to <b>Transient   Settings</b>, or <b>AC   Settings</b> then click <b>Advanced</b> button.</p> <p>When calculating DC operating point output is set to specified output voltage "IC". When calculating transient, output voltage is always delayed by one calculation step. This may affect stability of the schematic with closed loop.</p> |            |                                    |                       |

| F  |   | Function-2 |                                    | F/F_Function-2_LT.n15 |
|--|---|------------|------------------------------------|-----------------------|
| Model  | Parameter   | Units      | Description                        |                       |
| LT   | IC  | V          | Initial condition: output voltage. |                       |
|  | <p><b>Less than.</b> <math>V = Vx &lt; Vy</math> ? High : Low.</p> <p>if <math>Vx &lt; Vy</math> . . . . : <math>V = \text{High}</math><br/> if <math>Vx \geq Vy</math> . . . . : <math>V = \text{Low}</math></p> <p>High and Low are logical levels. To see and set logical levels go to <b>Transient   Settings</b>, or <b>AC   Settings</b> then click <b>Advanced</b> button.</p> <p>When calculating DC operating point output is set to specified output voltage "IC". When calculating transient, output voltage is always delayed by one calculation step. This may affect stability of the schematic with closed loop.</p> |            |                                    |                       |

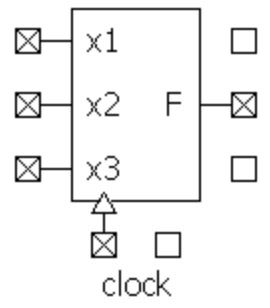
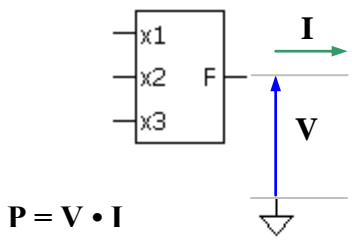
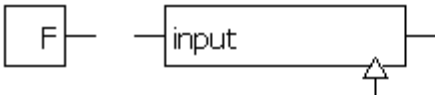
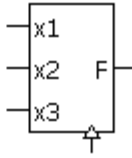
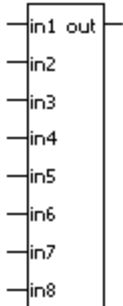
| F   |           | Function-2 |                                    | F/F_Function-2_Pwr.n15 |
|---|-----------|------------|------------------------------------|------------------------|
| Model   | Parameter | Units      | Description                        |                        |
| <b>Pwr</b><br>   | K         | V/V        | Gain.                              |                        |
|   | IC        | V          | Initial condition: output voltage. |                        |
| <p><b>“Signed” power function.</b> <math>V = K * \text{pwr}(Vx, Vy)</math>.</p> <p>The function is calculated as follows:</p> <p>if <math>Vy = 0</math>:</p> <ul style="list-style-type: none"> <li>if <math>Vx &lt; 0 \dots : V = -K</math></li> <li>if <math>Vx = 0 \dots : V = 0</math></li> <li>if <math>Vx &gt; 0 \dots : V = K</math></li> </ul> <p>if <math>Vy \neq 0</math>:</p> <ul style="list-style-type: none"> <li>if <math>Vx &lt; 0 \dots : V = -K * (-Vx)^{Vy}</math></li> <li>if <math>Vx = 0 \dots : V = 0</math></li> <li>if <math>Vx &gt; 0 \dots : V = K * Vx^{Vy}</math></li> </ul> <p>When calculating DC operating point output is set to specified output voltage “IC”. When calculating transient, output voltage is always delayed by one calculation step. This may affect stability of the schematic with closed loop.</p> |           |            |                                    |                        |

| F   |           | Function-2 |                                    | F/F_Function-2_Mag.n15 |
|---|-----------|------------|------------------------------------|------------------------|
| Model   | Parameter | Units      | Description                        |                        |
| <b>Mag</b><br>   | K         | V/V        | Gain.                              |                        |
|   | IC        | V          | Initial condition: output voltage. |                        |
| <p><b>Magnitude.</b> <math>V = K * \text{sqrt}(Vx^2 + Vy^2)</math>.</p> <p>When calculating DC operating point output is set to specified output voltage “IC”. When calculating transient, output voltage is always delayed by one calculation step. This may affect stability of the schematic with closed loop.</p> |           |            |                                    |                        |

| <b>F</b>   | Function-2 | F/F_Function-2_Phase.nl5 |                                    |
|--|------------|--------------------------|------------------------------------|
| Model  | Parameter  | Units                    | Description                        |
| <b>Phase</b><br>  | K          | V/V                      | Gain.                              |
|  | IC         | V                        | Initial condition: output voltage. |
| <p><b>Phase.</b> <math>V = K * \text{phase}(V_x, V_y)</math>.</p> <p>V in Volts is equal to phase of a vector <math>V_x + jV_y</math> in degrees.<br/>                 If <math>V_x = 0</math> and <math>V_y = 0</math>: <math>V = 0</math>.</p> <p>When calculating DC operating point output is set to specified output voltage "IC". When calculating transient, output voltage is always delayed by one calculation step. This may affect stability of the schematic with closed loop.</p> |            |                          |                                    |

| <b>F</b>  | Function-2 | F/F_Function-2_Table.nl5 |   |
|---|------------|--------------------------|---|
| Model   | Parameter  | Units                    | Description   |
| <b>Table</b><br>  | X          |                          | Comma-separated string, X (input values).           |
|   | Y          |                          | Comma-separated string, Y (input values).           |
|   | Table      |                          | Comma-separated string, Table of Z (output values). |
|   | IC         | V                        | Initial condition: output voltage.                  |
| <p><b>2D look-up table.</b> Function output is defined by "Table" parameter in the csv ("comma separated values") format, as follows:</p> <p>"Table" parameter of the model defines output of the component Z as a function of X and Y inputs of the component in the following format:</p> $Z_{11}, Z_{12}, \dots, Z_{1N}, Z_{21}, Z_{22}, \dots, Z_{2N}, \dots, Z_{M1}, Z_{M2}, \dots, Z_{MN}$ <p>where:</p> <ul style="list-style-type: none"> <li>• <math>Z_{ij}</math> defines output of the function for input values <math>X_i</math> and <math>Y_j</math>;</li> <li>• N is total number of X input values, defined by "X" parameter;</li> <li>• M is total number of Y input values, defined by "Y" parameter.</li> </ul> <p>Output value between specified X and Y points is linearly interpolated on both coordinates. Output value below <math>X_1</math> is linearly extrapolated using <math>X_1 \dots X_2</math> interval data, output value above <math>X_N</math> is linearly extrapolated using <math>X_{(N-1)} \dots X_N</math> interval data. The same rule is applied to Y coordinate</p> <p>See <i>Working with 2D Table model</i> chapter for more details.</p> <p>When calculating DC operating point output is set to specified output voltage "IC". When calculating transient, output voltage is always delayed by one calculation step. This may affect stability of the schematic with closed loop.</p> |            |                          |   |

| <b>F</b>      | Function-2  |       | F/F_Function-2_SubCir.n15                   |
|---------------|---|-------|---|
| Model         | Parameter   | Units | Description                                 |
| <b>SubCir</b> | File  |       | File name of subcircuit schematic.          |
|               | Pin1  |       | Name of subcircuit label connected to pin 1 |
|               | Pin2  |       | Name of subcircuit label connected to pin 2 |
|               | Pin3  |       | Name of subcircuit label connected to pin 3 |
|               | Cmd   |       | Subcircuit start-up command string          |
|               | IC  |       | Subcircuit Initial conditions string        |
|               | <b>Subcircuit.</b> See <i>Working with Subcircuits</i> chapter for details. |       |   |

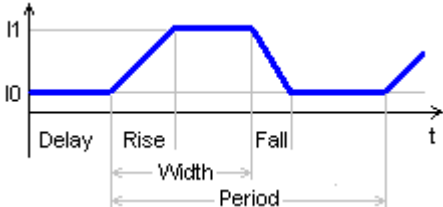
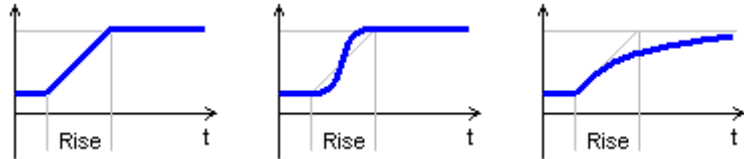
| F  | Custom function   | Function |   |
|--|---|----------|---|
|  |  |          |  |
|  | Models  |          | Traces  |
| <p>This is a customized component. A component can be edited in the <b>Edit Component</b> dialog box. See <i>Editing customized component</i> chapter for instructions on editing a component.</p> <p>This component may have:</p> <ul style="list-style-type: none"> <li>- arbitrary size up to 32(width) X 8(height),</li> <li>- up to 8 inputs on the left side,</li> <li>- one output on the right side,</li> <li>- one or no clock pins on the bottom side.</li> <li>- custom input and output names.</li> </ul> <p>Examples of Custom function component:</p> <div style="display: flex; justify-content: space-around; align-items: flex-start;"> <div style="text-align: center;">  </div> <div style="text-align: center;">  </div> <div style="text-align: center;">  </div> </div> |   |          |   |

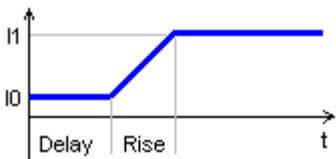



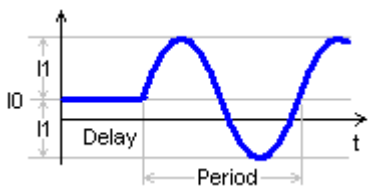
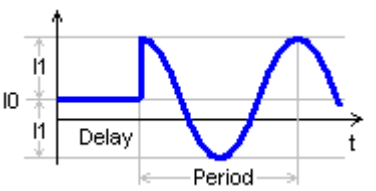
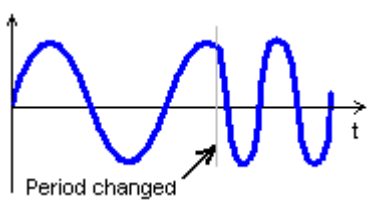
| <b>F</b>   | Custom function | F/F_Custom_Function_Ex1.nl5<br>F/F_Custom_Function_Ex2.nl5<br>F/F_Custom_Function_Ex3.nl5 |  |
|--|-----------------|---|--|
| Model  | Parameter       | Units   | Description                              |
| <b>Function</b>  | F(x)            | V   | Output as function of the inputs.        |
|  | F(s)            |   | AC transfer function in <i>s</i> domain. |
|  | IC              | V   | Initial condition: output voltage.       |
| <p><b>Arbitrary function.</b></p> <p><b>Transient analysis.</b> F(x) defines output voltage as a function of the following variables:</p> <p><b><i>pin_name</i></b> – input voltage on the input pin “pin_name”<br/> <b><i>t</i></b> - current time<br/> <b><i>V(name)</i></b> - voltage on the component <b><i>name</i></b><br/> <b><i>I(name)</i></b> - current through the component <b><i>name</i></b><br/> <b><i>P(name)</i></b> – power on the component <b><i>name</i></b></p> <p>where <b><i>name</i></b> is the name of any component in the schematic. If F(x) is blank, output is zero. F(s) is ignored.</p> <p>Example:<br/> <math display="block">F(x) = \max(x1,x2,x3)</math> <math display="block">F(x) = (in1+in2)*V(R1)</math></p> <p>If <b><i>clock</i></b> pin does not exist, the model operates in “continuous” mode: the function is calculated and applied to the output on every calculation step. Please note that input voltages and variables <b><i>V</i></b>, <b><i>I</i></b>, <b><i>P</i></b> are taken at previous calculation step. This may affect stability of the schematic with closed loop.</p> <p>If <b><i>clock</i></b> pin exists, the model operates in “synchronized” mode: the function is calculated and applied to the output only on rising edge of logical clock signal. As a result, “synchronized” mode provides faster simulation than “continuous” mode.</p> <p><b>AC analysis.</b> F(s) defines transfer function in <i>s</i> domain. Only operators and functions that support complex numbers can be used in this function. The following variables can be used in the function:</p> <p><b><i>f</i></b> – current AC frequency, Hz<br/> <b><i>w</i></b> – angular AC frequency, <math>w = 2\pi f</math>.<br/> <b><i>s</i></b> or <b><i>p</i></b> – Laplace parameter, <math>s = p = j*2\pi f</math>.</p> <p>Example:<br/> <math display="block">F(s) = 1/(1+s)</math> <math display="block">F(s) = \exp(-1mk*s)</math></p> <p>F(s) is calculated at each frequency. If F(s) is blank, it is assumed to be 1. Also, if F(x) is not blank, it is linearized at DC operating point, and F(s) is multiplied by linearized gain.</p> <p>When calculating DC operating point for transient or AC analysis, output is set to specified output voltage “IC”.</p> <p>If <b><i>clock</i></b> pin exists, F(s) is ignored, and transfer function of the model is zero.</p> |                 |   |  |

|   |                |  |   |
|---|----------------|--|---|
| <b>I</b>  | Current source | <p style="writing-mode: vertical-rl; transform: rotate(180deg);">Models</p> <p>I<br/>Pulse<br/>Step<br/>Sin<br/>PWL<br/>Function<br/>File<br/>SubCir</p> | <p style="writing-mode: vertical-rl; transform: rotate(180deg);">Traces</p>  <p><math>P = V \cdot I</math></p> |
|  |                |  |   |

|          |  |       |             |              |
|----------|--|-------|-------------|--------------|
| <b>I</b> | Current source                                 |       |             | I/I_CS_I.n15 |
| Model    | Parameter                                      | Units | Description |              |
| <b>I</b> | I  | A     | Current.    |              |
|          | <b>Constant current source.</b> Current = "I". |       |             |              |

| I  |           | Current source |                                  | I/I_CS_Pulse.n15 |
|--|-----------|----------------|----------------------------------|------------------|
| Model  | Parameter | Units          | Description                      |                  |
| Pulse  | I1        | A              | Pulse On current.                |                  |
|  | I0        | A              | Pulse Off current.               |                  |
|  | Period    | s              | Period.                          |                  |
|  | Width     | s              | Pulse width.                     |                  |
|  | Slope     |                | Slope type: Linear/Cos/Exp       |                  |
|  | Rise      | s              | Pulse rise length.               |                  |
|  | Fall      | s              | Pulse fall length.               |                  |
|  | Delay     | s              | Delay before first pulse starts. |                  |
| <p><b>Pulse current source.</b> Pulses start after “Delay” time. “Rise” time is included into “Width”, “Fall” time is not included into “Width”. Almost every parameter can be set to zero and infinity (“inf”), otherwise the error message will be displayed.</p>  <p>Slope type applies both to pulse rise and fall. The following slope types are available:</p>  <p style="text-align: center;"> <span>Linear</span>                      <span>Cos (cosine)</span>                      <span>Exp (exponential)</span> </p> |           |                |                                  |                  |

| I  | Current source | I/I_CS_Step.n15 |                            |
|--|----------------|-----------------|----------------------------|
| Model  | Parameter      | Units           | Description                |
| <b>Step</b>  | I1             | A               | Step On current.           |
|  | I0             | A               | Step Off current.          |
|  | Slope          |                 | Slope type: Linear/Cos/Exp |
|  | Rise           | s               | Step rise length.          |
|  | Delay          | s               | Delay before step starts.  |
| <p><b>Step current source.</b> Step starts after “Delay” time.</p>  <p>The following slope types are available:</p>  <p style="text-align: center;"> <span data-bbox="438 1008 519 1039">Linear</span> <span data-bbox="673 1008 828 1039">Cos (cosine)</span> <span data-bbox="925 1008 1128 1039">Exp (exponential)</span> </p> |                |                 |                            |

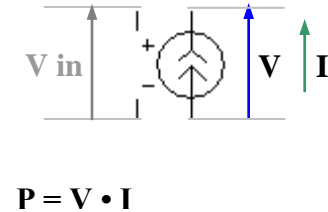
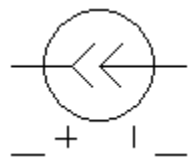
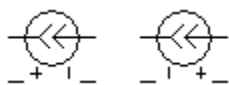
| I   |           | Current source | I/I_CS_Sin.nl5                   |
|---|-----------|----------------|----------------------------------|
| Model   | Parameter | Units          | Description                      |
| Sin   | I1        | A              | Current amplitude.               |
|   | I0        | A              | Current baseline.                |
|   | Period    | s              | Period.                          |
|   | Phase     | deg            | Phase.                           |
|   | Delay     | s              | Delay before sine signal starts. |
| <p><b>Sine current source.</b> Sine signal starts after “Delay” time. “Phase” is sine phase in degrees at the moment when signal starts:</p> <div style="display: flex; justify-content: space-around; align-items: flex-end;"> <div style="text-align: center;">  <p><b>Phase = 0</b></p> </div> <div style="text-align: center;">  <p><b>Phase = 90</b></p> </div> </div> <p>If transient is paused, sine period changed, then transient is continued, the phase of the signal remains continuous, providing smooth sine signal of variable frequency:</p> <div style="text-align: center;">  </div> |           |                |                                  |

|   |                |       |                           |                |
|---|----------------|-------|---------------------------|----------------|
| <b>I</b>  | Current source |       |                           | I/I_CS_PWL.n15 |
| Model   | Parameter      | Units | Description               |                |
| PWL   | pwl            |       | Comma-separated string.   |                |
|   | Cycle          |       | Cycling (repeat): No/Yes. |                |
|   | Delay          | s     | Delay.                    |                |
| <p><b>Piece-wise linear current source.</b> Signal is defined by “pwl” parameter in the csv (“comma-separated values”) format, as follows:</p> $t_0, I_0, t_1, I_1, \dots, t_n, I_n$ <p>where all <math>t</math> and <math>I</math> can be numerical values or expressions. If <math>t &lt; t_0</math>, signal is <math>I_0</math>. If <math>t_0 &lt; t &lt; t_1</math>, signal value is linearly interpolated between <math>I_0</math> and <math>I_1</math>, etc. If <math>t &gt; t_n</math>, then signal value is <math>I_n</math> if “Cycle” parameter is set to “No”, otherwise signal defined in <math>t_0 \dots t_n</math> interval is repeated continuously. In addition, the whole signal is delayed by “Delay” time.</p> <p>Example:<br/> <math>pwl = 0, 0, 1, 2, 4, 3, 5, 0, 8, 0</math></p> <p>If “Cycle” = Yes, “Delay” = 0, the following current will be generated:</p> <p>See <i>Working with PWL source</i> chapter for more details.</p> |                |       |                           |                |

|   |                |       |             |                     |
|---|----------------|-------|-------------|---------------------|
| <b>I</b>  | Current source |       |             | I/I_CS_Function.n15 |
| Model   | Parameter      | Units | Description |                     |
| Function  | F(t)           | A     | Function    |                     |
| <p><b>Arbitrary function.</b> F(t) defines current as a function of the following variables:</p> <ul style="list-style-type: none"> <li><math>t</math> - current time</li> <li><math>V(name)</math> - voltage on the component <i>name</i></li> <li><math>I(name)</math> - current through the component <i>name</i></li> <li><math>P(name)</math> – power on the component <i>name</i></li> </ul> <p>where <i>name</i> is the name of any component in the schematic. If F(t) is blank, current is zero.</p> <p>Example:<br/> <math>F(t) = \sin(t) * (1 + \cos(t * .01))</math><br/> <math>F(t) = V(R1) * I(R1)</math></p> <p>Please note that <b>V</b>, <b>I</b>, and <b>P</b> variables are taken at previous calculation step. This may affect stability of the schematic with closed loop.</p> |                |       |             |                     |

|             |   |       |                           |                 |
|-------------|---|-------|---------------------------|-----------------|
| <b>I</b>    | Current source  |       |                           | I/I_CS_File.n15 |
| Model       | Parameter   | Units | Description               |                 |
| <b>File</b> | File  |       | File name.                |                 |
|             | Cycle   |       | Cycling (repeat): No/Yes. |                 |
|             | Delay   | s     | Delay.                    |                 |
|             | <p><b>Current source defined in the text file.</b> “File” parameter is a file name, with full path to the file. If the file is located in the same directory as schematic file, the path can be omitted. Signal is defined in the following format:</p> <pre>                     &lt;if first line does not start with a number, it is ignored&gt;                     t0,I0                     t1,I1                     .....                     tn,In                 </pre> <p>where all t and I can be numerical values or expressions. If <math>t &lt; t_0</math>, signal is <math>v_0</math>. If <math>t_0 &lt; t &lt; t_1</math>, signal value is linearly interpolated between <math>I_0</math> and <math>I_1</math>, etc. If <math>t &gt; t_n</math>, then signal value is <math>I_n</math> if “Cycle” parameter is set to “No”, otherwise signal defined in <math>t_0 \dots t_n</math> interval is repeated continuously. In addition, the whole signal is delayed by “Delay” time.</p> <p>Example:</p> <pre>                     0,0                     1,2                     4,3                     5,0                     8,0                 </pre> <p>If “Cycle” = Yes, “Delay” = 0, the following current will be generated:</p> |       |                           |                 |

|  |                |       |   |                   |
|--|----------------|-------|---|-------------------|
| <b>I</b>   | Current source |       |   | I/I_CS_SubCir.n15 |
| Model  | Parameter      | Units | Description                                 |                   |
| <b>SubCir</b>  | File           |       | File name of subcircuit schematic.          |                   |
|  | Pin1           |       | Name of subcircuit label connected to pin 1 |                   |
|  | Pin2           |       | Name of subcircuit label connected to pin 2 |                   |
|  | Cmd            |       | Subcircuit start-up command string          |                   |
|  | IC             |       | Subcircuit Initial conditions string        |                   |
| <p><b>Subcircuit.</b> See <i>Working with Subcircuits</i> chapter for details.</p> |                |       |   |                   |

|   |   |  |  |
|---|---|--|--|
| <b>I</b>  | Voltage controlled current source   | Models<br>Linear<br>Function<br>PWL<br>VCO<br>Pulse<br>PWM<br>SubCir | Traces<br><br>$P = V \cdot I$ |
|  |   |  |  |
| Views   |  |  |  |

|          |                                   |       |   |                     |
|----------|-----------------------------------|-------|---|---------------------|
| <b>I</b> | Voltage controlled current source |       |   | I/I_VCCS_Linear.nl5 |
| Model    | Parameter                         | Units | Description   |                     |
| Linear   | K                                 | A/V   | Gain<br><br><b>Linear voltage controlled current source. <math>I = K * V_{in}</math>.</b> |                     |



| <b>I</b>        | Voltage controlled current source   |       | I/I_VCCS_Function.nl5                    |
|-----------------|---|-------|--|
| Model           | Parameter   | Units | Description                              |
| <b>Function</b> | F(x)  | A     | Output as function of the input.         |
|                 | F(s)  |       | AC transfer function in <b>s</b> domain. |
|                 | IC  | A     | Initial condition: output current.       |
|                 | <p><b>Arbitrary function.</b></p> <p><b>Transient analysis.</b> F(x) defines output current as a function of the following variables:</p> <p><b>x</b> – input voltage Vin<br/> <b>t</b> - current time<br/> <b>V(name)</b> - voltage on the component <b>name</b><br/> <b>I(name)</b> - current through the component <b>name</b><br/> <b>P(name)</b> – power on the component <b>name</b></p> <p>where <b>name</b> is the name of any component in the schematic. If F(x) is blank, output is zero. F(s) is ignored.</p> <p>Example:<br/> <math>F(x) = x^3</math><br/> <math>F(x) = x * \sin(t)</math><br/> <math>F(x) = P(r1)+P(r2)</math></p> <p>Please note that input voltage <b>x</b> and variables <b>V</b>, <b>I</b>, and <b>P</b> are taken at previous calculation step. This may affect stability of the schematic with closed loop.</p> <p><b>AC analysis.</b> F(s) defines transfer function in <b>s</b> domain. Only operators and functions that support complex numbers can be used in this function. The following variables can be used in the function:</p> <p><b>f</b> – current AC frequency, Hz<br/> <b>w</b> – angular AC frequency, <math>w = 2\pi f</math>.<br/> <b>s</b> or <b>p</b> – Laplace parameter, <math>s = p = j*2\pi f</math>.</p> <p>Example:<br/> <math>F(s) = 1/(1+s)</math><br/> <math>F(s) = \exp(-1mk*s)</math></p> <p>F(s) is calculated at each frequency. If F(s) is blank, it is assumed to be 1. Also, if F(x) is not blank, it is linearized at DC operating point, and F(s) is multiplied by linearized gain.</p> <p>When calculating DC operating point for transient or AC analysis, output is set to specified output current “IC”.</p> |       |  |

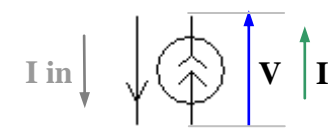
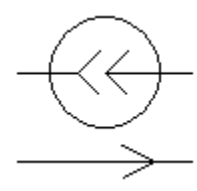
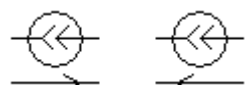
| I     | Voltage controlled current source  |       |                                | I/I_VCCS_PWL.n15 |
|-------|--|-------|--------------------------------|------------------|
| Model | Parameter  | Units | Description                    |                  |
| PWL   | pwl  |       | Comma-separated string, K(Vin) |                  |
|       | <p><b>Piece-wise linear voltage controlled current source.</b> Source gain K is defined by “pwl” string as a function of input voltage Vin. See <i>Working with PWL model</i> chapter for details.</p> |       |                                |                  |

| I  | Voltage controlled current source |       |   | I/I_VCCS_VCO.n15 |
|--|-----------------------------------|-------|---|------------------|
| Model  | Parameter                         | Units | Description   |                  |
| VCO  | I1                                | A     | Current amplitude (Sin), or Pulse On current (Pulse). |                  |
|  | I0                                | A     | Current baseline (Sin), or Pulse Off current (Pulse). |                  |
|  | dFdV                              | Hz/V  | Gain.   |                  |
|  | Phase                             | deg   | Phase.  |                  |
|  | Type                              |       | Signal type: Sin/Square/Triangle/Sawtooth.            |                  |
| <p><b>Voltage controlled oscillator.</b> Output current is a signal with frequency equal to:<br/> <math>f(\text{Hz}) = \text{dFdV} * \text{Vin}</math>.</p> <p>For Sine signal, “I0” is baseline, and “I1” is amplitude. For Square, Triangle, and Sawtooth signals, “I0” is Off level, “I1” is On level. “Phase” is additional phase of the signal, in degrees.</p> |                                   |       |   |                  |

| I   | Voltage controlled current source |       |                    | I/I_VCCS_Pulse.n15 |
|---|-----------------------------------|-------|--------------------|--------------------|
| Model   | Parameter                         | Units | Description        |                    |
| Pulse   | Width                             | s     | Pulse width.       |                    |
|   | Threshold                         | V     | Voltage threshold. |                    |
|   | I1                                | A     | Pulse On current.  |                    |
|   | I0                                | A     | Pulse Off current. |                    |
| <p><b>One-shot pulse generator.</b> When increasing input voltage Vin crosses “Threshold” value, current pulse of “Width” duration is generated. “I0” is pulse Off level, “I1” is pulse On level. If increasing Vin crosses “Threshold” value while pulse is generated, the pulse is restarted.</p> |                                   |       |                    |                    |

| I  |           | Voltage controlled current source |   | I/I_VCCS_PWM.n15 |
|--|-----------|-----------------------------------|---|------------------|
| Model  | Parameter | Units                             | Description                               |                  |
| PWM  | I1        | A                                 | Pulse On current.                         |                  |
|  | I0        | A                                 | Pulse Off current.                        |                  |
|  | F         | Hz                                | Frequency.                                |                  |
|  | Phase     | deg                               | Phase.                                    |                  |
|  | Vmax      | V                                 | Input voltage corresponding to 100% duty. |                  |
| <p><b>Voltage controlled Pulse-Width Modulator.</b> Output current is a pulse signal of frequency “F” shifted by “Phase”. Input voltage <math>V_{in}</math> is sampled at the beginning of each cycle of the signal, and width of the output pulse during this cycle is calculated according to the equation:</p> $\text{width} = 1/F * (V_{in} / V_{max})$ <p>or</p> $\text{duty} = 100\% * (V_{in} / V_{max});$ <p>If the width is equal or less than zero, a short “On” pulse with the width equal to the minimum calculation step at that moment will be generated. If the width is equal or greater than period of frequency “F”, a short “Off” pulse at the end of the period will be generated. Due to that, the frequency of the output signal is always “F”. Such a signal can be, for instance, divided by D-trigger to create a signal with duty cycle less than 50%.</p> |           |                                   |   |                  |

| I  |           | Voltage controlled current source |   | I/I_VCCS_SubCir.n15 |
|--|-----------|-----------------------------------|---|---------------------|
| Model  | Parameter | Units                             | Description                                 |                     |
| SubCir   | File      |                                   | File name of subcircuit schematic.          |                     |
|  | Pin1      |                                   | Name of subcircuit label connected to pin 1 |                     |
|  | Pin2      |                                   | Name of subcircuit label connected to pin 2 |                     |
|  | Pin3      |                                   | Name of subcircuit label connected to pin 3 |                     |
|  | Pin4      |                                   | Name of subcircuit label connected to pin 4 |                     |
|  | Cmd       |                                   | Subcircuit start-up command string          |                     |
|  | IC        |                                   | Subcircuit Initial conditions string        |                     |
| <p><b>Subcircuit.</b> See <i>Working with Subcircuits</i> chapter for details.</p> |           |                                   |   |                     |

|          |   |        |  |        |   |
|----------|---|--------|--|--------|---|
| <b>I</b> | Current controlled current source   | Models | Linear<br>Function<br>PWL<br>CCO<br>Pulse<br>PWM<br>SubCir | Traces |  |
|          |  |        |  |        | $P = V \cdot I$   |
| Views    |  |        |  |        |   |

|   |                                   |       |             |                     |
|---|-----------------------------------|-------|-------------|---------------------|
| <b>I</b>  | Current controlled current source |       |             | I/I_CCCS_Linear.n15 |
| Model   | Parameter                         | Units | Description |                     |
| Linear  | K                                 | A/A   | Gain        |                     |
| <b>Linear current controlled current source. <math>I = K \cdot I_{in}</math>.</b> |                                   |       |             |                     |

| I        | Current controlled current source  |       | I/I_CCCS_Function.nl5                    |
|----------|--|-------|--|
| Model    | Parameter  | Units | Description                              |
| Function | F(x)   | A     | Output as function of the input.         |
|          | F(s)   |       | AC transfer function in <b>s</b> domain. |
|          | IC   | A     | Initial condition: output current.       |
|          | <p><b>Arbitrary function.</b></p> <p><b>Transient analysis.</b> F(x) defines output current as a function of the following variables:</p> <p><b>x</b> – input current lin<br/> <b>t</b> - current time<br/> <b>V(name)</b> - voltage on the component <b>name</b><br/> <b>I(name)</b> - current through the component <b>name</b><br/> <b>P(name)</b> – power on the component <b>name</b></p> <p>where <b>name</b> is the name of any component in the schematic. If F(x) is blank, output is zero. F(s) is ignored.</p> <p>Example:<br/> <math>F(x) = x^3</math><br/> <math>F(x) = x * \sin(t)</math><br/> <math>F(x) = P(r1)+P(r2)</math></p> <p>Please note that input current <b>x</b>, and variables <b>V</b>, <b>I</b>, and <b>P</b> are taken at previous calculation step. This may affect stability of the schematic with closed loop.</p> <p><b>AC analysis.</b> F(s) defines transfer function in <b>s</b> domain. Only operators and functions that support complex numbers can be used in this function. The following variables can be used in the function:<br/> <b>f</b> – current AC frequency, Hz<br/> <b>w</b> – angular AC frequency, <math>w = 2\pi f</math>.<br/> <b>s</b> or <b>p</b> – Laplace parameter, <math>s = p = j*2\pi f</math>.</p> <p>Example:<br/> <math>F(s) = 1/(1+s)</math><br/> <math>F(s) = \exp(-1mk*s)</math></p> <p>F(s) is calculated at each frequency. If F(s) is blank, it is assumed to be 1. Also, if F(x) is not blank, it is linearized at DC operating point, and F(s) is multiplied by linearized gain.</p> <p>When calculating DC operating point for transient or AC analysis, output is set to specified output current "IC".</p> |       |  |

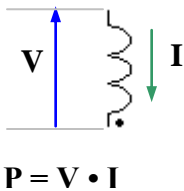

| I     | Current controlled current source   |       |                                | I/I_CCCS_PWL.n15 |
|-------|---|-------|--------------------------------|------------------|
| Model | Parameter   | Units | Description                    |                  |
| PWL   | pwl   |       | Comma-separated string, K(lin) |                  |
|       | <p><b>Piece-wise linear current controlled current source.</b> Source gain K is defined by “pwl” string as a function of input current lin. See <i>Working with PWLmodel</i> chapter for details.</p> |       |                                |                  |

| I  | Current controlled current source |       |   | I/I_CCCS_CCO.n15 |
|--|-----------------------------------|-------|---|------------------|
| Model  | Parameter                         | Units | Description   |                  |
| CCO  | I1                                | A     | Current amplitude (Sin), or Pulse On current (Pulse). |                  |
|  | I0                                | A     | Current baseline (Sin), or Pulse Off current (Pulse). |                  |
|  | dFdl                              | Hz/A  | Gain.   |                  |
|  | Phase                             | deg   | Phase.  |                  |
|  | Type                              |       | Signal type: Sin/Square/Triangle/Sawtooth.            |                  |
| <p><b>Current controlled oscillator.</b> Output current is a signal with frequency equal to:<br/> <math>f(\text{Hz}) = \text{dFdl} * \text{lin}.</math></p> <p>For Sine signal, “I0” is baseline, and “I1” is amplitude. For Square, Triangle, and Sawtooth signals, “I0” is Off level, “I1” is On level. “Phase” is additional phase of the signal, in degrees.</p> |                                   |       |   |                  |

| I   | Current controlled current source |       |                     | I/I_CCCS_Pulse.n15 |
|---|-----------------------------------|-------|---------------------|--------------------|
| Model   | Parameter                         | Units | Description         |                    |
| Pulse   | Width                             | s     | Pulse width.        |                    |
|   | Threshold                         | A     | Current threshold.  |                    |
|   | I1                                | A     | Pulse Off current.. |                    |
|   | I0                                | A     | Phase.              |                    |
| <p><b>One-shot pulse generator.</b> When increasing input current lin crosses “Threshold” value, current pulse of “Width” duration is generated. “I0” is pulse Off level, “I1” is pulse On level. If increasing lin crosses “Threshold” value while pulse is generated, the pulse is restarted.</p> |                                   |       |                     |                    |

| I   | Current controlled current source |       | I/I_CCCS_PWM.n15                          |
|---|-----------------------------------|-------|---|
| Model   | Parameter                         | Units | Description                               |
| PWM   | I1                                | A     | Pulse On current.                         |
|   | I0                                | A     | Pulse Off current.                        |
|   | F                                 | Hz    | Frequency.                                |
|   | Phase                             | deg   | Phase.                                    |
|   | I <sub>max</sub>                  | A     | Input current corresponding to 100% duty. |
| <p><b>Current controlled Pulse-Width Modulator.</b> Output current is a pulse signal of frequency “F” shifted by “Phase”. Input current I<sub>in</sub> is sampled at the beginning of each cycle of the signal, and width of the output pulse during this cycle is calculated according to the equation:</p> $\text{width} = 1/F * (I_{in} / I_{max})$ <p>or</p> $\text{duty} = 100\% * (I_{in} / I_{max});$ <p>If the width is equal or less than zero, a short “On” pulse with the width equal to the minimum calculation step at that moment will be generated. If the width is equal or greater than period of frequency “F”, a short “Off” pulse at the end of the period will be generated. Due to that, the frequency of the output signal is always “F”. Such a signal can be, for instance, divided by D-trigger to create a signal with duty cycle less than 50%.</p> |                                   |       |   |

| I  | Current controlled current source |       | I/I_CCCS_SubCir.n15                         |
|--|-----------------------------------|-------|---|
| Model  | Parameter                         | Units | Description                                 |
| SubCir   | File                              |       | File name of subcircuit schematic.          |
|  | Pin1                              |       | Name of subcircuit label connected to pin 1 |
|  | Pin2                              |       | Name of subcircuit label connected to pin 2 |
|  | Pin3                              |       | Name of subcircuit label connected to pin 3 |
|  | Pin4                              |       | Name of subcircuit label connected to pin 4 |
|  | Cmd                               |       | Subcircuit start-up command string          |
|  | IC                                |       | Subcircuit Initial conditions string        |
| <p><b>Subcircuit.</b> See <i>Working with Subcircuits</i> chapter for details.</p> |                                   |       |   |

|   |                 |        |                                |        |   |
|---|-----------------|--------|--------------------------------|--------|---|
| <b>L</b>  | <b>Inductor</b> | Models | L<br>PWL<br>Function<br>SubCir | Traces |  |
|  |                 |        |                                |        |   |

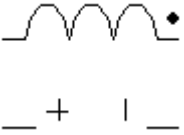

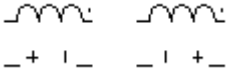
| <b>L</b>  | Inductor  | L/L_Inductor_L.n15 |  |  |
|---|-----------|--------------------|--|--|
| Model   | Parameter | Units              | Description  |  |
| <b>L</b>  | L         | H                  | Inductance   |  |
|   | IC        | A                  | Initial condition: current. Leave blank if IC not defined. |  |
| <p><b>Linear inductor.</b> <math>V = L \cdot di/dt</math>.</p> <p>When calculating DC operating point, if "IC" is defined, inductor is replaced with current source equal to IC. If "IC" is not defined (blank), inductor is temporarily replaced by short circuit, DC operating point calculated, and then the current through short circuit is assigned to the inductor as its initial current.</p> |           |                    |  |  |

| <b>L</b>   | Inductor  | L/L_Inductor_PWL.n15 |  |  |
|--|-----------|----------------------|--|--|
| Model  | Parameter | Units                | Description  |  |
| <b>PWL</b>   | pwl       |                      | Comma-separated string, L(I)                               |  |
|  | IC        | A                    | Initial condition: current. Leave blank if IC not defined. |  |
| <p><b>Piece-wise linear capacitor.</b> "pwl" string defines inductance as a function of current. See <i>Working with PWL model</i> chapter for details.</p> <p>When calculating DC operating point, if "IC" is defined, inductor is replaced with current source equal to IC. If "IC" is not defined (blank), inductor is temporarily replaced by short circuit, DC operating point calculated, and then the current through short circuit is assigned to the inductor as its initial current.</p> |           |                      |  |  |

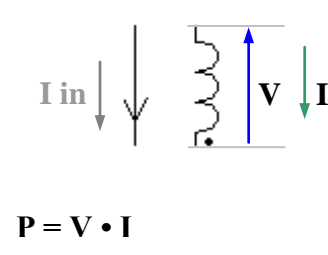
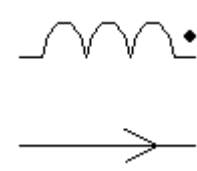
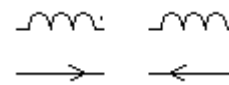


| L        |   | Inductor |  | L/L_Inductor_Function.n15 |
|----------|---|----------|--|---------------------------|
| Model    | Parameter   | Units    | Description                                    |                           |
| Function | Z(s)  | Ohm      | Impedance as a function of <b>s</b> parameter. |                           |
|          | <p><b>Impedance function in s domain.</b> For transient, constant impedance Z(0) is used. For linearized AC analysis, complex impedance Z(s) is used. The following variables can be used in the function:</p> <p><b>f</b> – current AC frequency, Hz<br/> <b>w</b> – angular AC frequency, <math>w = 2\pi f</math>.<br/> <b>s</b> or <b>p</b> – Laplace parameter, <math>s = p = j*2\pi f</math>.</p> <p>Example:</p> <p><math>Z(s) = 3n*s + 0.5</math>                    - 3 nH inductor in series with 0.5 Ohm resistor.<br/> <math>Z(s) = (1u*s)   (1.0/10p/s)</math>   - 1 mkH inductor in parallel with 10 pF capacitor.</p> |          |  |                           |

| L  |           | Inductor |   | L/L_Inductor_SubCir.n15 |
|--|-----------|----------|---|-------------------------|
| Model  | Parameter | Units    | Description                                 |                         |
| SubCir   | File      |          | File name of subcircuit schematic.          |                         |
|  | Pin1      |          | Name of subcircuit label connected to pin 1 |                         |
|  | Pin2      |          | Name of subcircuit label connected to pin 2 |                         |
|  | Cmd       |          | Subcircuit start-up command string          |                         |
|  | IC        |          | Subcircuit Initial conditions string        |                         |
| <p><b>Subcircuit.</b> See <i>Working with Subcircuits</i> chapter for details.</p> |           |          |   |                         |

|          |   |        |     |        |   |
|----------|---|--------|-----|--------|---|
| <b>L</b> | Voltage controlled inductor   | Models | PWL | Traces |  |
|          |  |        |     |        |   |
| Views    |  |        |     |        |   |

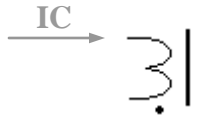

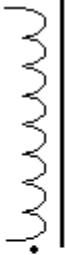
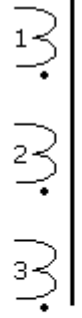
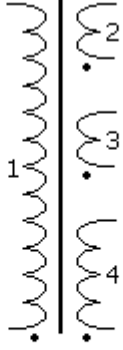
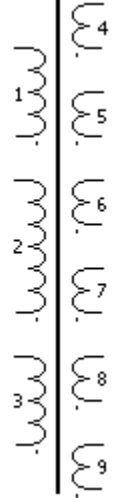
|   |                             |       |  |                 |
|---|-----------------------------|-------|--|-----------------|
| <b>L</b>  | Voltage controlled inductor |       |  | L/L_VCL_PWL.n15 |
| Model   | Parameter                   | Units | Description  |                 |
| <b>PWL</b>  | pwl                         |       | Comma-separated string, L(Vin)                             |                 |
|   | IC                          | A     | Initial condition: current. Leave blank if IC not defined. |                 |
| <p><b>Piece-wise linear voltage controlled inductor.</b> At any moment, inductor is a <b>linear</b> inductor. Its inductance L is a function of control voltage:</p> $V = L(Vin) \cdot di/dt.$ <p>“pwl” string defines inductance as a function of control voltage Vin. See <i>Working with PWL model</i> chapter for details.</p> <p>When calculating DC operating point, if “IC” is defined, inductor is replaced with current source equal to IC. If “IC” is not defined (blank), inductor is temporarily replaced by short circuit, DC operating point calculated, and then the current through short circuit is assigned to the inductor as its initial current.</p> |                             |       |  |                 |

|          |   |        |     |        |   |
|----------|---|--------|-----|--------|---|
| <b>L</b> | Current controlled inductor   | Models | PWL | Traces |  <p><math>P = V \cdot I</math></p> |
|          |  |        |     |        |   |
| Views    |  |        |     |        |   |

|  |                             |       |  |                 |
|--|-----------------------------|-------|--|-----------------|
| <b>L</b>   | Current controlled inductor |       |  | L/L_CCL_PWL.n15 |
| Model  | Parameter                   | Units | Description  |                 |
| <b>PWL</b>   | pwl                         |       | Comma-separated string, L( <i>Iin</i> )                    |                 |
|  | IC                          | A     | Initial condition: current. Leave blank if IC not defined. |                 |
| <p><b>Piece-wise linear current controlled inductor.</b> At any moment, inductor is a <b>linear</b> inductor. Its inductance L is a function of control current:</p> $V = L(Iin) \cdot dI/dt.$ <p>“pwl” string defines inductance as a function of control current <i>Iin</i>. See <i>Working with PWL model</i> chapter for details.</p> <p>When calculating DC operating point, if “IC” is defined, inductor is replaced with current source equal to IC. If “IC” is not defined (blank), inductor is temporarily replaced by short circuit, DC operating point calculated, and then the current through short circuit is assigned to the inductor as its initial current.</p> |                             |       |  |                 |

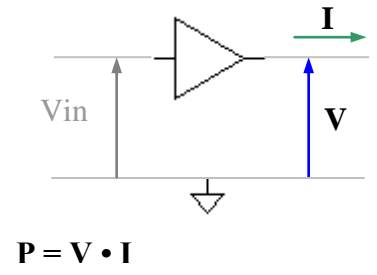
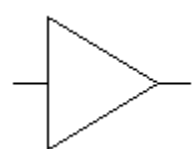
|          |                          |        |          |        |  |
|----------|--------------------------|--------|----------|--------|--|
| <b>L</b> | <b>Coupled inductors</b> | Models | <b>L</b> | Traces |  |
|          |                          |        |          |        |  |

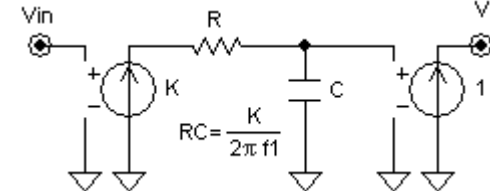
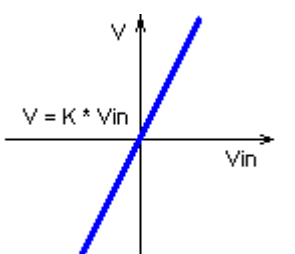
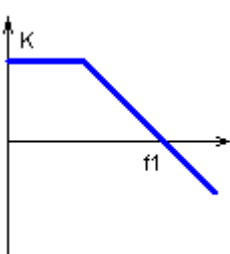
|  |                          |                            |  |  |
|--|--------------------------|----------------------------|--|--|
| <b>L</b>   | <b>Coupled inductors</b> | L/L_CoupledInductors_L.n15 |  |  |
| Model  | Parameter                | Units                      | Description  |  |
| <b>L</b>   | L1                       | H                          | L1 inductance  |  |
|  | L2                       | H                          | L2 inductance  |  |
|  | K                        |                            | Coupling coefficient (-1...1)                                  |  |
|  | IC1                      | A                          | L1 initial condition: current. Leave blank if IC1 not defined. |  |
|  | IC2                      | A                          | L2 initial condition: current. Leave blank if IC2 not defined. |  |
| <p><b>Coupled linear inductors.</b></p> $V1 = L1 \cdot dI1/dt + M \cdot dI2/dt$ $V2 = M \cdot dI1/dt + L2 \cdot dI2/dt$ <p>Where <math>M = K \cdot \sqrt{L1 \cdot L2}</math> is mutual inductance.</p> <p>When calculating DC operating point, initial conditions IC1 and IC2 are independently applied to corresponding inductors L1 and L2, similar to how it is done for the component <b>L</b> (inductor).</p> |                          |                            |  |  |

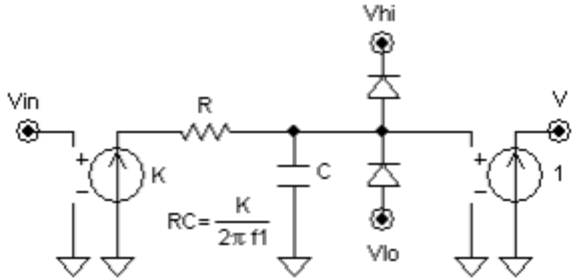
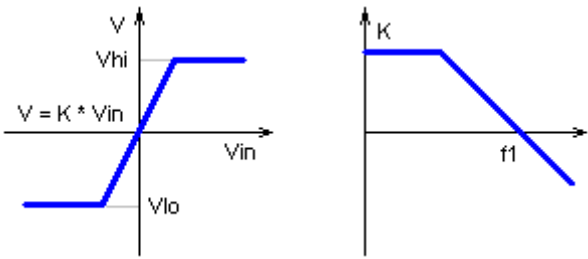
|   |  |        |                    |        |   |
|---|--|--------|--------------------|--------|---|
| <b>L</b>  | <b>Custom coupled inductors</b>  | Models | <b>L</b><br>SubCir | Traces |  |
|  |  |        |                    |        |   |
| Views   | <p>This is a customized component. A component can be edited in the <b>Edit Component</b> dialog box. See <i>Editing customized component</i> chapter for instructions on editing a component.</p> <p>This component may have:</p> <ul style="list-style-type: none"> <li>- height from 2 to 32,</li> <li>- up to 9 windings (total) on both sides,</li> <li>- arbitrary length of a winding.</li> </ul> <p>Examples of Custom coupled inductors component:</p> <div style="display: flex; justify-content: space-around; align-items: center;"> <div style="text-align: center;">  </div> <div style="text-align: center;">  </div> <div style="text-align: center;">  </div> <div style="text-align: center;">  </div> </div> |        |                    |        |   |

| L  |           | Custom coupled inductors |  | L/L_CustomCoupledInductors_L.n15 |
|--|-----------|--------------------------|--|----------------------------------|
| Model  | Parameter | Units                    | Description  |                                  |
| L  | L1        | H                        | L1 inductance  |                                  |
|  | ...       | H                        | ...  |                                  |
|  | LN        | H                        | LN inductance  |                                  |
|  | K12       |                          | L1-L2 coupling coefficient (-1...1)                            |                                  |
|  | ...       |                          | ...  |                                  |
|  | K(N-1)N   |                          | L(N-1)-LN coupling coefficient (-1...1)                        |                                  |
|  | IC1       | A                        | L1 initial condition: current. Leave blank if IC1 not defined. |                                  |
|  | ...       | A                        | ...  |                                  |
|  | ICN       | A                        | LN initial condition: current. Leave blank if ICN not defined. |                                  |
| <p><b>Custom coupled inductors.</b></p> $V1 = L1 \cdot dI1/dt + M12 \cdot dI2/dt + \dots + M1N \cdot dIN/dt$ $V2 = M12 \cdot dI1/dt + L2 \cdot dI2/dt + \dots + M2N \cdot dIN/dt$ $\dots$ $VN = M1N \cdot dI1/dt + M2N \cdot dI2/dt + \dots + LN \cdot dIN/dt$ <p>Where <math>M_{ij} = K_{ij} \cdot \sqrt{L_i \cdot L_j}</math> is mutual inductance, <math>M_{ij} = M_{ji}</math>.</p> <p>When calculating DC operating point, initial conditions ICN are independently applied to corresponding inductors LN, similar to how it is done for the component L (inductor).</p> <p>If only one winding is defined, a component behaves exactly as a linear inductor L.</p> <p>Please be aware that coupling coefficients <math>K_{ij}</math> should be properly specified within allowable range (-1...1) in order to represent a “physically-realizable” system. See NL5 website (<a href="http://nl5.sidelinesoft.com">http://nl5.sidelinesoft.com</a>) and other public resources for more details.</p> <p>If all coupling coefficients are equal to 1 (or -1), using Winding components W with one magnetizing inductor will give better performance and more stable solution.</p> |           |                          |  |                                  |


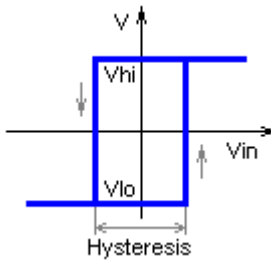
| L      |  | Custom coupled inductors |   | L/L_CustomCoupledInductors_SubCir.n15 |
|--------|--|--------------------------|---|---------------------------------------|
| Model  | Parameter  | Units                    | Description                                 |                                       |
| SubCir | File   |                          | File name of subcircuit schematic.          |                                       |
|        | Pin1   |                          | Name of subcircuit label connected to pin 1 |                                       |
|        | ...  |                          | ...   |                                       |
|        | PinN   |                          | Name of subcircuit label connected to pin N |                                       |
|        | Cmd  |                          | Subcircuit start-up command string          |                                       |
|        | IC   |                          | Subcircuit Initial conditions string        |                                       |
|        | <p><b>Subcircuit.</b> See <i>Working with Subcircuits</i> chapter for details.</p> |                          |   |                                       |


|   |           |  |   |
|---|-----------|--|---|
| O   | Amplifier | Models<br>Linear<br>OpAmp<br>Comparator<br>Function<br>PWL<br>SubCir | Traces<br> <p><math>P = V \cdot I</math></p> |
|  |           |  |   |

|   |           |   |                                    |  |
|---|-----------|---|------------------------------------|--|
| O   | Amplifier | O/O_Amplifier_Linear.n15  |                                    |  |
| Model   | Parameter | Units   | Description                        |  |
| Linear  | K         | V/V   | Gain                               |  |
|   | f1        | Hz  | Unit gain frequency.               |  |
|   | IC        | V   | Initial condition: output voltage. |  |
| <p><b>Linear amplifier.</b> “K” is open loop gain. Frequency response consists of one pole, “f1” is unit gain frequency. “K” and “f1” can be set to infinity (“inf”).</p>                           |           |   |                                    |  |
| <p>When calculating DC operating point, if “f1” is not infinity and “IC” is defined, amplifier output is set to specified output voltage “IC”. If “IC” is blank, static characteristic is used.</p> |           |   |                                    |  |
|  <p style="text-align: center;"><math>RC = \frac{K}{2\pi f1}</math></p>  |           |   |                                    |  |
| <p><b>Equivalent schematic</b></p>  |           |   |                                    |  |
|  <p style="text-align: center;"><math>V = K \cdot V_{in}</math></p>  |           |  |                                    |  |
| <p><b>Static characteristic</b></p>   |           | <p><b>AC response</b></p>   |                                    |  |

| O Amplifier   |           | O/O_Amplifier_OpAmp.nl5 |                                    |
|---|-----------|-------------------------|------------------------------------|
| Model   | Parameter | Units                   | Description                        |
| OpAmp   | K         | V/V                     | Gain                               |
|   | f1        | Hz                      | Unit gain frequency.               |
|   | Vhi       | V                       | Max output voltage.                |
|   | Vlo       | V                       | Min output voltage.                |
|   | IC        | V                       | Initial condition: output voltage. |
| <p><b>Linear amplifier with output limiter.</b> “K” is open loop gain. Frequency response consists of one pole, “f1” is unit gain frequency. “K” and “f1” can be set to infinity (“inf”). Output voltage is limiting between “Vlo” and “Vhi”.</p> <p>When calculating DC operating point, if “f1” is not infinity and “IC” is defined, amplifier output is set to specified output voltage “IC”. If “IC” is blank, static characteristic is used.</p> <p><b>Please note:</b> if both “K” and “f1” are set to infinity, the model may experience convergence problem. Use <b>Comparator</b> model instead.</p> |           |                         |                                    |
|  <p style="text-align: center;"><b>Equivalent schematic</b></p>   |           |                         |                                    |
|  <div style="display: flex; justify-content: space-around;"> <div style="text-align: center;"> <p><b>Static characteristic</b></p> </div> <div style="text-align: center;"> <p><b>AC response</b></p> </div> </div>  |           |                         |                                    |

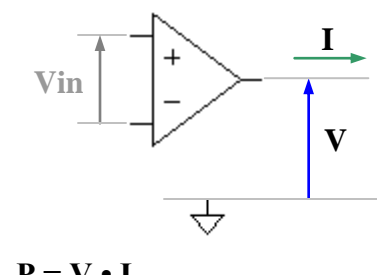
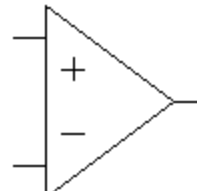


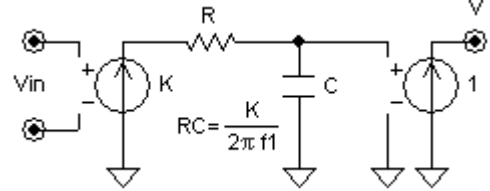
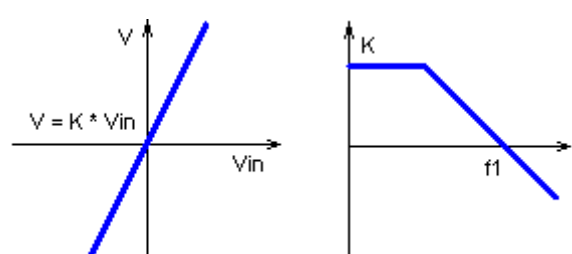
|   |            |                                     |                              |
|---|------------|-------------------------------------|------------------------------|
| <p><b>O</b> Amplifier</p>   |            | <p>O/O_Amplifier_Comparator.nl5</p> |                              |
| Model   | Parameter  | Units                               | Description                  |
| <p><b>Comparator</b></p>   | Hysteresis | V                                   | Hysteresis                   |
|   | Vhi        | V                                   | Max output voltage.          |
|   | Vlo        | V                                   | Min output voltage.          |
|   | Delay      | s                                   | Output delay.                |
|   | IC         |                                     | Initial condition: Low/High. |
| <p><b>Comparator with hysteresis.</b> Comparator output is set to “Vhi” or “Vlo” using following rules:</p> <p style="margin-left: 40px;"> <math>V_{in} &gt; \text{Hysteresis}/2 \dots : V = V_{hi}</math><br/> <math>V_{in} &lt; -\text{Hysteresis}/2 \dots : V = V_{lo}</math><br/>                     Otherwise . . . . . : <math>V = \text{previous state}</math> </p> <p>The output is delayed by “Delay” time. Input pulses shorter than “Delay” will not pass through and will not affect output.</p> <p>When calculating DC operating point comparator output is set to “Vlo” or to “Vhi”, according to selected “IC”.</p> <div style="text-align: center;">  </div> <p><b>Static characteristic</b></p> |            |                                     |                              |

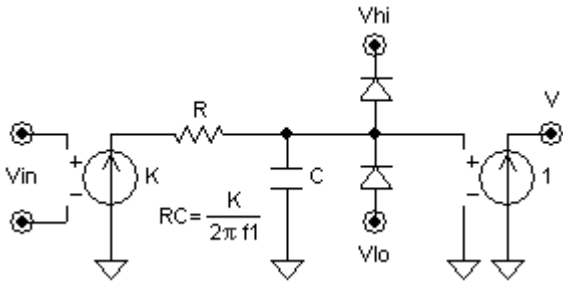
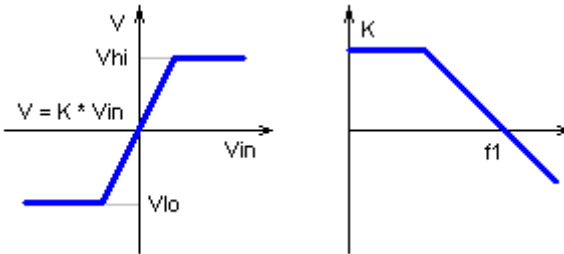
| O Amplifier  |           | O/O_Amplifier_Function.nl5 |  |
|--|-----------|----------------------------|--|
| Model  | Parameter | Units                      | Description                              |
| <b>Function</b><br><br>   | F(x)      | V                          | Output as function of the input.         |
|  | F(s)      |                            | AC transfer function in <b>s</b> domain. |
|  | IC        | V                          | Initial condition: output voltage.       |
| <p><b>Arbitrary function.</b></p> <p><b>Transient analysis.</b> F(x) defines output voltage as a function of the following variables:</p> <ul style="list-style-type: none"> <li><b>x</b> – input voltage Vin</li> <li><b>t</b> - current time</li> <li><b>V(name)</b> - voltage on the component <b>name</b></li> <li><b>I(name)</b> - current through the component <b>name</b></li> <li><b>P(name)</b> – power on the component <b>name</b></li> </ul> <p>where <b>name</b> is the name of any component in the schematic. If F(x) is blank, output is zero. F(s) is ignored.</p> <p>Example:</p> <ul style="list-style-type: none"> <li>F(x) = x^3</li> <li>F(x) = x * sin(t)</li> <li>F(x) = P(r1)+P(r2)</li> </ul> <p>Please note that input voltage <b>x</b>, and variables <b>V</b>, <b>I</b>, and <b>P</b> are taken at previous calculation step. This may affect stability of the schematic with closed loop.</p> <p><b>AC analysis.</b> F(s) defines transfer function in <b>s</b> domain. Only operators and functions that support complex numbers can be used in this function. The following variables can be used in the function:</p> <ul style="list-style-type: none"> <li><b>f</b> – current AC frequency, Hz</li> <li><b>w</b> – angular AC frequency, <b>w = 2πf</b> .</li> <li><b>s</b> or <b>p</b> – Laplace parameter, <b>s = p = j*2πf</b>.</li> </ul> <p>Example:</p> <ul style="list-style-type: none"> <li>F(s) = 1/(1+s)</li> <li>F(s) = exp(-1mk*s)</li> </ul> <p>F(s) is calculated at each frequency. If F(s) is blank, it is assumed to be 1. Also, if F(x) is not blank, it is linearized at DC operating point, and F(s) is multiplied by linearized gain.</p> <p>When calculating DC operating point for transient or AC analysis, output is set to specified output voltage “IC”.</p> |           |                            |  |

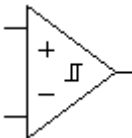
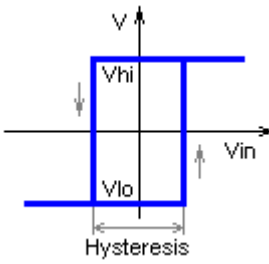
|          |  |       |                                |                       |
|----------|--|-------|--------------------------------|-----------------------|
| <b>O</b> | Amplifier  |       |                                | O/O_Amplifier_PWL.n15 |
| Model    | Parameter  | Units | Description                    |                       |
| PWL      | pwl  |       | Comma-separated string, K(Vin) |                       |
|          | <b>Piece-wise linear amplifier.</b> Amplifier gain K is defined by “pwl” string as a function of input voltage Vin. See <i>Working with PWL model</i> chapter for details. |       |                                |                       |

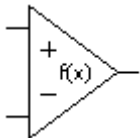
|   |           |       |   |                          |
|---|-----------|-------|---|--------------------------|
| <b>O</b>  | Amplifier |       |   | O/O_Amplifier_SubCir.n15 |
| Model   | Parameter | Units | Description                                 |                          |
| SubCir  | File      |       | File name of subcircuit schematic.          |                          |
|   | Pin1      |       | Name of subcircuit label connected to pin 1 |                          |
|   | Pin2      |       | Name of subcircuit label connected to pin 2 |                          |
|   | Cmd       |       | Subcircuit start-up command string          |                          |
|   | IC        |       | Subcircuit Initial conditions string        |                          |
| <b>Subcircuit.</b> See <i>Working with Subcircuits</i> chapter for details. |           |       |   |                          |

|   |                        |  |   |
|---|------------------------|--|---|
| O   | Differential amplifier | Models<br>Linear<br>OpAmp<br>Comparator<br>Function<br>PWL<br>SubCir | Traces<br> <p><math>P = V \cdot I</math></p> |
|  |                        |  |   |

|   |                        |                        |                                    |  |
|---|------------------------|------------------------|------------------------------------|--|
| O   | Differential amplifier | O/O_DiffAmp_Linear.n15 |                                    |  |
| Model   | Parameter              | Units                  | Description                        |  |
| Linear  | K                      | V/V                    | Gain                               |  |
|   | f1                     | Hz                     | Unit gain frequency.               |  |
|   | IC                     | V                      | Initial condition: output voltage. |  |
| <p><b>Linear differential amplifier.</b> “K” is open loop gain. Frequency response consists of one pole, “f1” is unit gain frequency. “K” and “f1” can be set to infinity (“inf”).</p>              |                        |                        |                                    |  |
| <p>When calculating DC operating point, if “f1” is not infinity and “IC” is defined, amplifier output is set to specified output voltage “IC”. If “IC” is blank, static characteristic is used.</p> |                        |                        |                                    |  |
|  <p style="text-align: center;"><math>RC = \frac{K}{2\pi f1}</math></p>  |                        |                        |                                    |  |
| Equivalent schematic  |                        |                        |                                    |  |
|    |                        |                        |                                    |  |
| Static characteristic   |                        | AC response            |                                    |  |

| O Differential amplifier  |           | O/O_DiffAmp_OpAmp.n15 |                                    |
|---|-----------|-----------------------|------------------------------------|
| Model   | Parameter | Units                 | Description                        |
| OpAmp   | K         | V/V                   | Gain                               |
|   | f1        | Hz                    | Unit gain frequency.               |
|   | Vhi       | V                     | Max output voltage.                |
|   | Vlo       | V                     | Min output voltage.                |
|   | IC        | V                     | Initial condition: output voltage. |
| <p><b>Linear amplifier with output limiter.</b> “K” is open loop gain. Frequency response consists of one pole, “f1” is unit gain frequency. “K” and “f1” can be set to infinity (“inf”). Output voltage is limiting between “Vlo” and “Vhi”.</p> <p>When calculating DC operating point, if “f1” is not infinity and “IC” is defined, amplifier output is set to specified output voltage “IC”. If “IC” is blank, static characteristic is used.</p> <p><b>Please note:</b> if both “K” and “f1” are set to infinity, the model may experience convergence problem. Use <b>Comparator</b> model instead.</p> |           |                       |                                    |
|  <p style="text-align: center;"><b>Equivalent schematic</b></p>   |           |                       |                                    |
|  <div style="display: flex; justify-content: space-around;"> <div style="text-align: center;"> <p><b>Static characteristic</b></p> </div> <div style="text-align: center;"> <p><b>AC response</b></p> </div> </div>  |           |                       |                                    |

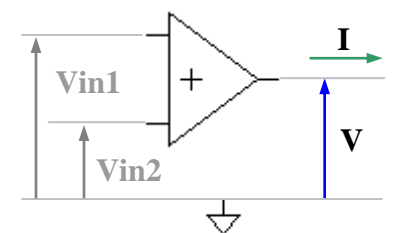
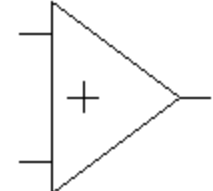
|   |                        |                            |                              |
|---|------------------------|----------------------------|------------------------------|
| <b>O</b>  | Differential amplifier | O/O_DiffAmp_Comparator.nl5 |                              |
| Model   | Parameter              | Units                      | Description                  |
| <b>Comparator</b><br>  | Hysteresis             | V                          | Hysteresis                   |
|   | Vhi                    | V                          | Max output voltage.          |
|   | Vlo                    | V                          | Min output voltage.          |
|   | Delay                  | s                          | Output delay.                |
|   | IC                     |                            | Initial condition: Low/High. |
| <p><b>Comparator with hysteresis.</b> Comparator output is set to “Vhi” or “Vlo” using following rules:</p> <p style="margin-left: 40px;"> <math>V_{in} &gt; \text{Hysteresis}/2 \dots : V = V_{hi}</math><br/> <math>V_{in} &lt; -\text{Hysteresis}/2 \dots : V = V_{lo}</math><br/>                 Otherwise . . . . . : <math>V = \text{previous state}</math> </p> <p>The output is delayed by “Delay” time. Input pulses shorter than “Delay” will not pass through and will not affect output.</p> <p>When calculating DC operating point comparator output is set to “Vlo” or to “Vhi”, according to selected “IC”.</p> <div style="text-align: center;">  </div> <p><b>Static characteristic</b></p> |                        |                            |                              |

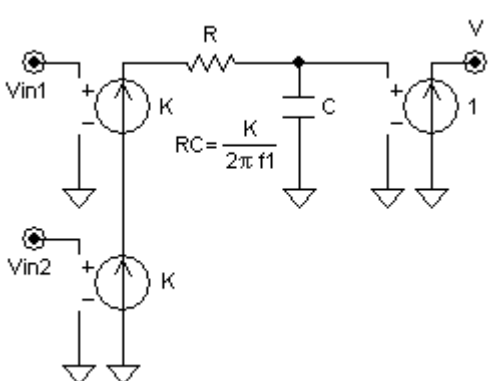
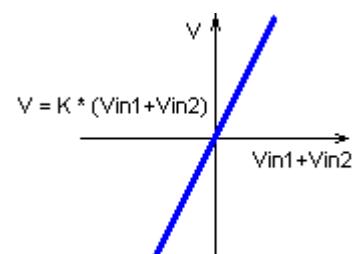
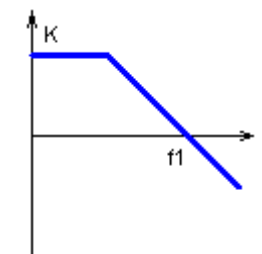
| O Differential amplifier  |           | O/O_DiffAmp_Function.nl5 |  |
|---|-----------|--------------------------|--|
| Model   | Parameter | Units                    | Description                              |
| <b>Function</b><br>  | F(x)      | V                        | Output as function of the input.         |
|   | F(s)      |                          | AC transfer function in <i>s</i> domain. |
|   | IC        | V                        | Initial condition: output voltage.       |
| <p><b>Arbitrary function.</b></p> <p><b>Transient analysis.</b> F(x) defines output voltage as a function of the following variables:</p> <ul style="list-style-type: none"> <li><b>x</b> – input voltage <i>V<sub>in</sub></i></li> <li><b>t</b> - current time</li> <li><b>V(name)</b> - voltage on the component <i>name</i></li> <li><b>I(name)</b> - current through the component <i>name</i></li> <li><b>P(name)</b> – power on the component <i>name</i></li> </ul> <p>where <i>name</i> is the name of any component in the schematic. If F(x) is blank, output is zero. F(s) is ignored.</p> <p>Example:</p> <ul style="list-style-type: none"> <li>F(x) = x<sup>3</sup></li> <li>F(x) = x * sin(t)</li> <li>F(x) = P(r1)+P(r2)</li> </ul> <p>Please note that input voltage <b>x</b>, and variables <b>V</b>, <b>I</b>, and <b>P</b> are taken at previous calculation step. This may affect stability of the schematic with closed loop.</p> <p><b>AC analysis.</b> F(s) defines transfer function in <i>s</i> domain. Only operators and functions that support complex numbers can be used in this function. The following variables can be used in the function:</p> <ul style="list-style-type: none"> <li><b>f</b> – current AC frequency, Hz</li> <li><b>w</b> – angular AC frequency, <b>w = 2πf</b>.</li> <li><b>s</b> or <b>p</b> – Laplace parameter, <b>s = p = j*2πf</b>.</li> </ul> <p>Example:</p> <ul style="list-style-type: none"> <li>F(s) = 1/(1+s)</li> <li>F(s) = exp(-1mk*s)</li> </ul> <p>F(s) is calculated at each frequency. If F(s) is blank, it is assumed to be 1. Also, if F(x) is not blank, it is linearized at DC operating point, and F(s) is multiplied by linearized gain.</p> <p>When calculating DC operating point for transient or AC analysis, output is set to specified output voltage “IC”.</p> |           |                          |  |

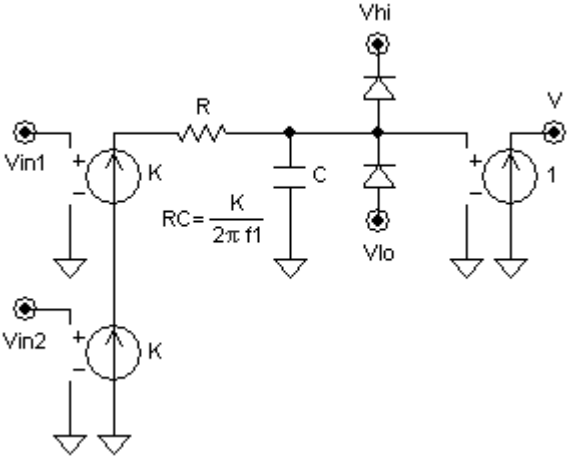
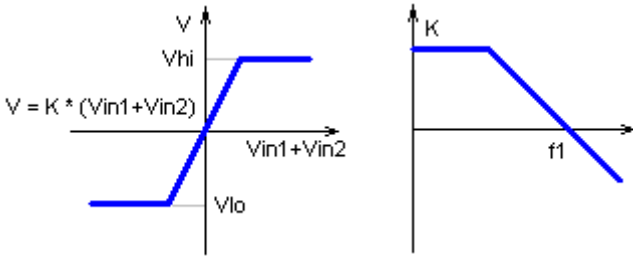
|          |  |       |                                |                     |
|----------|--|-------|--------------------------------|---------------------|
| <b>O</b> | Differential amplifier   |       |                                | O/O_DiffAmp_PWL.n15 |
| Model    | Parameter  | Units | Description                    |                     |
| PWL      | pwl  |       | Comma-separated string, K(Vin) |                     |
|          | <b>Piece-wise linear amplifier.</b> Amplifier gain K is defined by “pwl” string as a function of input voltage Vin. See <i>Working with PWL model</i> chapter for details. |       |                                |                     |

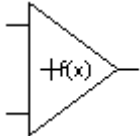
|          |   |       |   |                        |
|----------|---|-------|---|------------------------|
| <b>O</b> | Differential amplifier  |       |   | O/O_DiffAmp_SubCir.n15 |
| Model    | Parameter   | Units | Description                                 |                        |
| SubCir   | File  |       | File name of subcircuit schematic.          |                        |
|          | Pin1  |       | Name of subcircuit label connected to pin 1 |                        |
|          | Pin2  |       | Name of subcircuit label connected to pin 2 |                        |
|          | Pin3  |       | Name of subcircuit label connected to pin 3 |                        |
|          | Cmd   |       | Subcircuit start-up command string          |                        |
|          | IC  |       | Subcircuit Initial conditions string        |                        |
|          | <b>Subcircuit.</b> See <i>Working with Subcircuits</i> chapter for details. |       |   |                        |



|   |                   |  |   |
|---|-------------------|--|---|
| O   | Summing amplifier | Models<br>Linear<br>OpAmp<br>Function<br>PWL<br>SubCir | Traces<br> <p><math>P = V \cdot I</math></p> |
|  |                   |  |   |

|  |                   |                       |                                    |  |
|--|-------------------|-----------------------|------------------------------------|--|
| O  | Summing amplifier | O/O_SumAmp_Linear.n15 |                                    |  |
| Model  | Parameter         | Units                 | Description                        |  |
| Linear   | K                 | V/V                   | Gain                               |  |
|  | f1                | Hz                    | Unit gain frequency.               |  |
|  | IC                | V                     | Initial condition: output voltage. |  |
| <p><b>Linear summing amplifier.</b> “K” is open loop gain. Frequency response consists of one pole, “f1” is unit gain frequency. “K” and “f1” can be set to infinity (“inf”).</p> <p>When calculating DC operating point, if “f1” is not infinity and “IC” is defined, amplifier output is set to specified output voltage “IC”. If “IC” is blank, static characteristic is used.</p> <div style="text-align: center;">  <p><b>Equivalent schematic</b></p> </div> <div style="display: flex; justify-content: space-around;"> <div style="text-align: center;">  <p><b>Static characteristic</b></p> </div> <div style="text-align: center;">  <p><b>AC response</b></p> </div> </div> |                   |                       |                                    |  |

| O Summing amplifier   |           | O/O_SumAmp_OpAmp.nl5 |                                    |
|---|-----------|----------------------|------------------------------------|
| Model   | Parameter | Units                | Description                        |
| OpAmp   | K         | V/V                  | Gain                               |
|   | f1        | Hz                   | Unit gain frequency.               |
|   | Vhi       | V                    | Max output voltage.                |
|   | Vlo       | V                    | Min output voltage.                |
|   | IC        | V                    | Initial condition: output voltage. |
| <p><b>Linear amplifier with output limiter.</b> “K” is open loop gain. Frequency response consists of one pole, “f1” is unit gain frequency. “K” and “f1” can be set to infinity (“inf”). Output voltage is limiting between “Vlo” and “Vhi”.</p> <p>When calculating DC operating point, if “f1” is not infinity and “IC” is defined, amplifier output is set to specified output voltage “IC”. If “IC” is blank, static characteristic is used.</p>  <p style="text-align: center;"><b>Equivalent schematic</b></p>  <p style="display: flex; justify-content: space-around;"><span><b>Static characteristic</b></span> <span><b>AC response</b></span></p> |           |                      |                                    |

|  |                   |       |  |
|--|-------------------|-------|--|
| <b>O</b>   | Summing amplifier |       | O/O_SumAmp_Function.nl5                  |
| Model  | Parameter         | Units | Description                              |
| <b>Function</b><br><br>   | F(x)              | V     | Output as function of the input.         |
|  | F(s)              |       | AC transfer function in <b>s</b> domain. |
|  | IC                | V     | Initial condition: output voltage.       |
| <p><b>Arbitrary function.</b></p> <p><b>Transient analysis.</b> F(x) defines output voltage as a function of the following variables:</p> <ul style="list-style-type: none"> <li><b>x</b> – voltage Vin1+Vin2</li> <li><b>t</b> - current time</li> <li><b>V(name)</b> - voltage on the component <b>name</b></li> <li><b>I(name)</b> - current through the component <b>name</b></li> <li><b>P(name)</b> – power on the component <b>name</b></li> </ul> <p>where <b>name</b> is the name of any component in the schematic. If F(x) is blank, output is zero. F(s) is ignored.</p> <p>Example:</p> <ul style="list-style-type: none"> <li>F(x) = x^3</li> <li>F(x) = x * sin(t)</li> <li>F(x) = P(r1)+P(r2)</li> </ul> <p>Please note that input voltage <b>x</b>, and variables <b>V</b>, <b>I</b>, and <b>P</b> are taken at previous calculation step. This may affect stability of the schematic with closed loop.</p> <p><b>AC analysis.</b> F(s) defines transfer function in <b>s</b> domain. Only operators and functions that support complex numbers can be used in this function. The following variables can be used in the function:</p> <ul style="list-style-type: none"> <li><b>f</b> – current AC frequency, Hz</li> <li><b>w</b> – angular AC frequency, <b>w = 2πf</b> .</li> <li><b>s</b> or <b>p</b> – Laplace parameter, <b>s = p = j*2πf</b>.</li> </ul> <p>Example:</p> <ul style="list-style-type: none"> <li>F(s) = 1/(1+s)</li> <li>F(s) = exp(-1mk*s)</li> </ul> <p>F(s) is calculated at each frequency. If F(s) is blank, it is assumed to be 1. Also, if F(x) is not blank, it is linearized at DC operating point, and F(s) is multiplied by linearized gain.</p> <p>When calculating DC operating point for transient or AC analysis, output is set to specified output voltage “IC”.</p> |                   |       |  |

|          |   |       |  |                    |
|----------|---|-------|--|--------------------|
| <b>O</b> | Summing amplifier   |       |  | O/O_SumAmp_PWL.n15 |
| Model    | Parameter   | Units | Description                                  |                    |
| PWL      | pwl   |       | Comma-separated string, $K(V_{in1}+V_{in2})$ |                    |
|          | <b>Piece-wise linear amplifier.</b> Amplifier gain $K$ is defined by “pwl” string as a function of sum of input voltages $V_{in1}+V_{in2}$ . See <i>Working with PWL model</i> chapter for details. |       |  |                    |

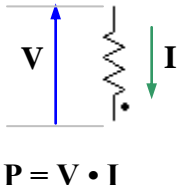

|   |                   |       |   |                       |
|---|-------------------|-------|---|-----------------------|
| <b>O</b>  | Summing amplifier |       |   | O/O_SumAmp_SubCir.n15 |
| Model   | Parameter         | Units | Description                                 |                       |
| SubCir  | File              |       | File name of subcircuit schematic.          |                       |
|   | Pin1              |       | Name of subcircuit label connected to pin 1 |                       |
|   | Pin2              |       | Name of subcircuit label connected to pin 2 |                       |
|   | Pin3              |       | Name of subcircuit label connected to pin 3 |                       |
|   | Cmd               |       | Subcircuit start-up command string          |                       |
|   | IC                |       | Subcircuit Initial conditions string        |                       |
| <b>Subcircuit.</b> See <i>Working with Subcircuits</i> chapter for details. |                   |       |   |                       |

|       |   |        |   |
|-------|---|--------|---|
| O     | Voltage controlled amplifier  | PWL    |   |
|       |  | Models |  <p><math>P = V \cdot I</math></p> |
| Views |  |        |   |

|       |                              |                 |  |  |
|-------|------------------------------|-----------------|--|--|
| O     | Voltage controlled amplifier | O/O_VCA_PWL.n15 |  |  |
| Model | Parameter                    | Units           | Description  |  |
| PWL   | pwl                          |                 | <p>Comma-separated string, <math>K(V_c)</math></p> <p><b>Piece-wise linear voltage controlled amplifier.</b> At any moment, the amplifier is a <b>linear</b> amplifier. Its gain <math>K</math> is a function of control voltage:</p> $V = K(V_c) \cdot V_{in}.$ <p>“pwl” string defines gain <math>K</math> as a function of control voltage <math>V_c</math>. See <i>Working with PWL model</i> chapter for details.</p> |  |

|       |   |        |   |
|-------|---|--------|---|
| O     | Current controlled amplifier  | PWL    |  |
|       |  | Models | Traces  |
| Views |  |        |   |

|       |                              |       |   |                 |
|-------|------------------------------|-------|---|-----------------|
| O     | Current controlled amplifier |       |   | O/O_CCA_PWL.n15 |
| Model | Parameter                    | Units | Description   |                 |
| PWL   | pwl                          |       | Comma-separated string, K(Ic)<br><br><b>Piece-wise linear current controlled amplifier.</b> At any moment, the amplifier is a <b>linear</b> amplifier. Its gain K is a function of control current:<br><br>$V = K(Ic) * Vin.$<br><br>"pwl" string defines gain K as a function of control current Ic. See <i>Working with PWL model</i> chapter for more details. |                 |

|   |          |        |   |        |   |
|---|----------|--------|---|--------|---|
| <b>R</b>  | Resistor | Models | R<br>PWL<br>PWL-I<br>Function<br>SubCir | Traces |  |
|  |          |        |   |        |   |

|  |           |       |             |  |                    |
|--|-----------|-------|-------------|--|--------------------|
| <b>R</b>   | Resistor  |       |             |  | R/R_Resistor_R.n15 |
| Model  | Parameter | Units | Description |  |                    |
| R  | R         | Ohm   | Resistance  |  |                    |
| <p><b>Linear resistor.</b> <math>V = R \cdot I</math>.</p> |           |       |             |  |                    |

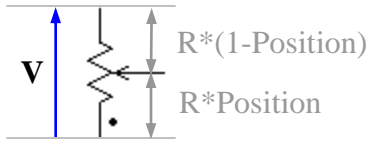
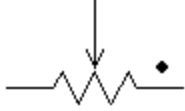
|   |           |       |                              |  |                      |
|---|-----------|-------|------------------------------|--|----------------------|
| <b>R</b>  | Resistor  |       |                              |  | R/R_Resistor_PWL.n15 |
| Model   | Parameter | Units | Description                  |  |                      |
| PWL   | pwl       |       | Comma-separated string, R(V) |  |                      |
| <p><b>Piece-wise linear resistor.</b> “pwl” string defines resistance as a function of voltage across the resistor R(V). See <i>Working with PWL model</i> chapter for details.</p> |           |       |                              |  |                      |

|  |           |       |                              |  |                        |
|--|-----------|-------|------------------------------|--|------------------------|
| <b>R</b>   | Resistor  |       |                              |  | R/R_Resistor_PWL-I.n15 |
| Model  | Parameter | Units | Description                  |  |                        |
| PWL-I  | pwl       |       | Comma-separated string, R(I) |  |                        |
| <p><b>Piece-wise linear resistor.</b> “pwl” string defines resistance as a function of current through the resistor R(I). See <i>Working with PWL model</i> chapter for details.</p> |           |       |                              |  |                        |

| <b>R</b> Resistor |  | R/R_Resistor_Function.n15 |  |
|-------------------|--|---------------------------|--|
| Model             | Parameter  | Units                     | Description                                    |
| Function          | Z(s)   | Ohm                       | Impedance as a function of <b>s</b> parameter. |
|                   | <p><b>Impedance function in s domain.</b> For transient, constant impedance <math>Z(0)</math> is used. For linearized AC analysis, complex impedance <math>Z(s)</math> is used. The following variables can be used in the function:</p> <p><b>f</b> – current AC frequency, Hz<br/> <b>w</b> – angular AC frequency, <math>w = 2\pi f</math>.<br/> <b>s</b> or <b>p</b> – Laplace parameter, <math>s = p = j*2\pi f</math>.</p> <p>Example:<br/> <math>Z(s) = 10+3n*s</math>      - 10 Ohm resistor in series with 3 nH inductor.<br/> <math>Z(s) = 10  {(1.0/3p/s)}</math>      - 10 Ohm resistor in parallel with 3 pF capacitor.</p> |                           |  |

| <b>R</b> Resistor  |           | R/R_Resistor_SubCir.n15 |   |
|--|-----------|-------------------------|---|
| Model  | Parameter | Units                   | Description                                 |
| SubCir   | File      |                         | File name of subcircuit schematic.          |
|  | Pin1      |                         | Name of subcircuit label connected to pin 1 |
|  | Pin2      |                         | Name of subcircuit label connected to pin 2 |
|  | Cmd       |                         | Subcircuit start-up command string          |
|  | IC        |                         | Subcircuit Initial conditions string        |
| <p><b>Subcircuit.</b> See <i>Working with Subcircuits</i> chapter for details.</p> |           |                         |   |

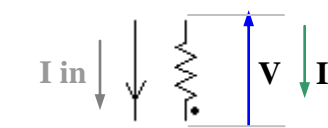
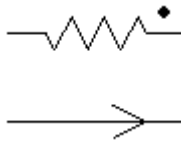
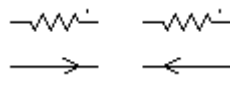


|   |               |        |               |        |   |
|---|---------------|--------|---------------|--------|---|
| <b>R</b>  | Potentiometer | Models | Potentiometer | Traces |  |
|  |               |        |               |        |   |

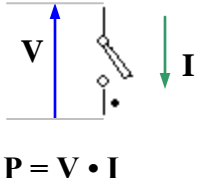
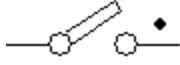
|  |               |                                     |                               |  |
|--|---------------|-------------------------------------|-------------------------------|--|
| <b>R</b>   | Potentiometer | R/R_Potentiometer_Potentiometer.n15 |                               |  |
| Model  | Parameter     | Units                               | Description                   |  |
| <b>Potentiometer</b>   | R             | Ohm                                 | Resistance                    |  |
|  | Position      |                                     | Position of the wiper (0...1) |  |
| <p><b>Potentiometer.</b> Position of the wiper is referenced to the terminal with dot:</p> <p>0 – wiper is connected to the terminal with dot<br/>                 1 – wiper is connected to another terminal.</p> |               |                                     |                               |  |

|          |                             |        |     |        |  |
|----------|-----------------------------|--------|-----|--------|--|
| <b>R</b> | Voltage controlled resistor | Models | PWL | Traces |  |
|          |                             |        |     |        |  |
| Views    |                             |        |     |        |  |

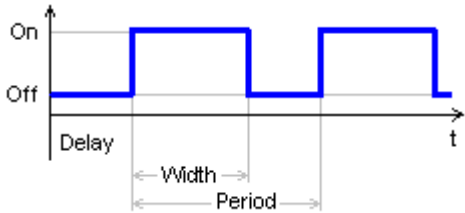
|   |                             |       |                                |                 |
|---|-----------------------------|-------|--------------------------------|-----------------|
| <b>R</b>  | Voltage controlled resistor |       |                                | R/R_VCR_PWL.n15 |
| Model   | Parameter                   | Units | Description                    |                 |
| PWL   | pwl                         |       | Comma-separated string, R(Vin) |                 |
| <p><b>Piece-wise linear voltage controlled resistor.</b> At any moment, resistor is a <b>linear</b> resistor. Its resistance R is a function of control voltage:</p> $V = R(Vin) \cdot I.$ <p>“pwl” string defines resistance as a function of control voltage <i>Vin</i>. See <i>Working with PWL model</i> chapter for details.</p> |                             |       |                                |                 |

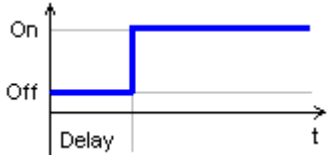
|          |   |        |     |        |   |
|----------|---|--------|-----|--------|---|
| <b>R</b> | Current controlled resistor   | Models | PWL | Traces |  <p><math>P = V \cdot I</math></p> |
|          |  |        |     |        |   |
| Views    |  |        |     |        |   |


|   |                             |       |   |                 |
|---|-----------------------------|-------|---|-----------------|
| <b>R</b>  | Current controlled resistor |       |   | R/R_CCR_PWL.n15 |
| Model   | Parameter                   | Units | Description                             |                 |
| PWL   | pwl                         |       | Comma-separated string, R( <i>lin</i> ) |                 |
| <p><b>Piece-wise linear current controlled resistor.</b> At any moment, resistor is a <b>linear</b> resistor. Its resistance R is a function of control current:</p> $V = R(lin) \cdot I.$ <p>“pwl” string defines resistance as a function of control current <i>lin</i>. See <i>Working with PWL model</i> chapter for details.</p> |                             |       |   |                 |

|          |   |        |   |        |   |
|----------|---|--------|---|--------|---|
| <b>S</b> | <b>Switch</b>   | Models | Switch<br>Pulse<br>Step<br>List<br>File<br>SubCir | Traces |  |
|          |  |        |   |        |   |

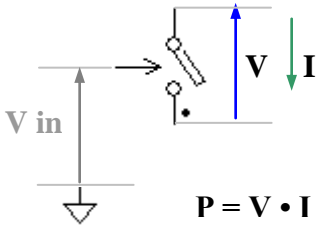
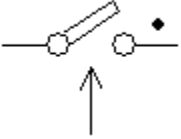
|               |  |                       |                       |  |
|---------------|--|-----------------------|-----------------------|--|
| <b>S</b>      | Switch   | S/S_Switch_Switch.n15 |                       |  |
| Model         | Parameter  | Units                 | Description           |  |
| <b>Switch</b> | Switch   |                       | Switch state: Off/On. |  |
|               | <p><b>Switch.</b> Off – open switch, infinite resistance. On – closed switch, zero resistance.</p> |                       |                       |  |

|  |           |                      |                                  |  |
|--|-----------|----------------------|----------------------------------|--|
| <b>S</b>   | Switch    | S/S_Switch_Pulse.n15 |                                  |  |
| Model  | Parameter | Units                | Description                      |  |
| <b>Pulse</b>   | Period    | s                    | Period.                          |  |
|  | Width     | s                    | Pulse width.                     |  |
|  | Delay     | s                    | Delay before first pulse starts. |  |
|  | Active    |                      | Active switch state: Off/On.     |  |
| <p><b>Pulse switch.</b> Switching starts after “Delay” time. Switch is in active state during “Width” time.</p> <p>The following switching diagram is shown for “Active” = On:</p> |           |                      |                                  |  |
|   |           |                      |                                  |  |

| <b>S</b>  | Switch    | S/S_Switch_Step.n15 |                              |
|---|-----------|---------------------|------------------------------|
| Model   | Parameter | Units               | Description                  |
| <b>Step</b>   | Delay     | s                   | Delay before active state.   |
|   | Active    |                     | Active switch state: Off/On. |
| <p><b>Step switch.</b> Switch is in active state after “Delay” time.</p> <p>The following switching diagram is shown for “Active” = On:</p>  |           |                     |                              |

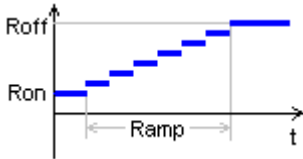
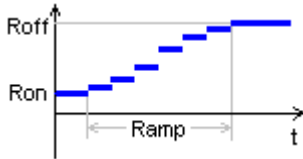
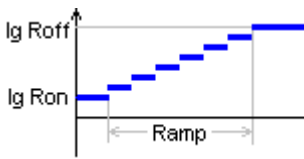
| <b>S</b>  | Switch    | V/S_Switch_List.n15 |                           |
|---|-----------|---------------------|---------------------------|
| Model   | Parameter | Units               | Description               |
| <b>List</b>   | List      |                     | Comma-separated string.   |
|   | Cycle     |                     | Cycling (repeat): No/Yes. |
|   | Delay     | s                   | Delay.                    |
| <p><b>List switch.</b> Switching sequence is defined in the “List” parameter in the csv (“comma-separated values”) format, as follows:</p> $t_0, s_0, t_1, s_1, \dots, t_n, s_n$ <p><math>s_0 \dots s_n</math> defines switch state: positive number corresponds to On state, zero or negative number - Off state. If <math>t &lt; t_0</math>, switch is in <math>s_0</math> state. At <math>t_0</math> switch is set to <math>s_0</math> state. At <math>t_1</math> switch is set to <math>s_1</math> state, and so on. At <math>t &gt; t_n</math>, switch remains in <math>s_n</math> state if “Cycle” parameter is set to “No”, otherwise states sequence defined in <math>t_0 \dots t_n</math> interval is repeated continuously. In addition, the whole signal is delayed by “Delay” time.</p> <p>Example:<br/> <code>List = 0,0,3,1,4,0,5,1,8,0</code></p> <p>The following switching diagram is shown for “Cycle” = Yes, “Delay” = 0:</p>  <p>See <i>Working with List model</i> chapter for more details.</p> |           |                     |                           |



|          |   |        |                                    |        |   |
|----------|---|--------|------------------------------------|--------|---|
| <b>S</b> | Logic controlled switch   | Models | Switch<br>Pulse<br>Steps<br>SubCir | Traces |  |
|          |  |        |                                    |        |   |

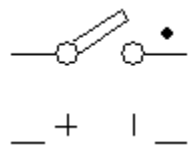
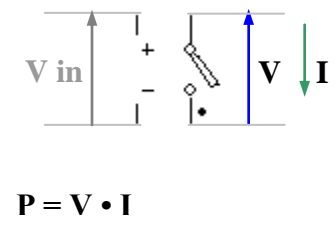
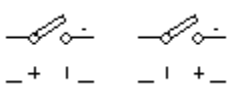
|  |                         |       |                            |                    |
|--|-------------------------|-------|----------------------------|--------------------|
| <b>S</b>   | Logic controlled switch |       |                            | S/S_LCS_Switch.n15 |
| Model  | Parameter               | Units | Description                |                    |
| <b>Switch</b>  | Active                  |       | Active state: Off/On.      |                    |
|  | IC                      |       | Initial condition: Off/On. |                    |
| <p><b>Logic controlled switch.</b> Switch is set to active or non-active state using following rules:</p> <p style="padding-left: 40px;"> <math>V_{in} &gt; \text{logical threshold} \dots : \text{active}</math><br/> <math>V_{in} &lt; \text{logical threshold} \dots : \text{non-active}</math> </p> <p>To see and set logical levels and threshold go to <b>Transient   Settings</b>, or <b>AC   Settings</b>, then click <b>Advanced</b> button.</p> <p>When calculating DC operating point switch is set to the state defined in "IC".</p> |                         |       |                            |                    |

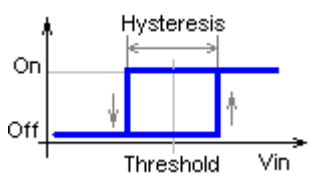
|  |                         |       |                       |                   |
|--|-------------------------|-------|-----------------------|-------------------|
| <b>S</b>   | Logic controlled switch |       |                       | S/S_LCS_Pulse.n15 |
| Model  | Parameter               | Units | Description           |                   |
| <b>Pulse</b>   | Width                   | s     | Pulse width.          |                   |
|  | Active                  |       | Active state: Off/On. |                   |
| <p><b>One-shot pulse generator.</b> When increasing input voltage <math>V_{in}</math> crosses logical threshold, switch is set to active state for "Width" time interval. If increasing <math>V_{in}</math> crosses logical threshold value while switch is in active state, the pulse is restarted.</p> <p>To see and set logical levels and threshold go to <b>Transient   Settings</b>, or <b>AC   Settings</b>, then click <b>Advanced</b> button.</p> |                         |       |                       |                   |

| <b>S</b>     | Logic controlled switch   |       | S/S_LCS_Steps.n15                          |
|--------------|---|-------|--|
| Model        | Parameter   | Units | Description                                |
| <b>Steps</b> | Roff  | Ohm   | Off state resistance.                      |
|              | Ron   | Ohm   | On state resistance.                       |
|              | Slope   |       | Type of resistance change: Linear/Cos/Log. |
|              | Ramp  | s     | Resistance ramp time.                      |
|              | Steps   |       | Number of resistance steps in the ramp.    |
|              | IC  |       | Initial condition: Off/On.                 |
|              | <p><b>Switch with resistance ramping.</b> When increasing input voltage <math>V_{in}</math> crosses logical threshold, switch resistance starts ramping from “Roff” to “Ron”. When decreasing input voltage <math>V_{in}</math> crosses logical threshold, switch resistance starts ramping from “Ron” to “Roff”.</p> <p>Resistance is changing during “Ramp” time interval, with number of steps specified by “Steps” parameter. If “Steps” = 0, resistance is changed instantly.</p> <p>“Slope” parameter specifies how resistance is changing during the ramp. The following slope types are available (“Steps” = 6):</p> <div style="display: flex; justify-content: space-around; align-items: flex-end;"> <div style="text-align: center;">  <p><b>Linear</b></p> </div> <div style="text-align: center;">  <p><b>Cos</b></p> </div> <div style="text-align: center;">  <p><b>Log</b></p> </div> </div> <p>To see and set logical levels and threshold go to <b>Transient   Settings</b>, or <b>AC   Settings</b>, then click <b>Advanced</b> button.</p> <p>When calculating DC operating point switch is set to the state specified in “IC”.</p> |       |  |

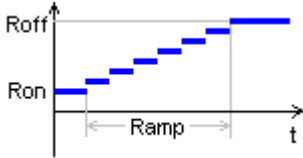
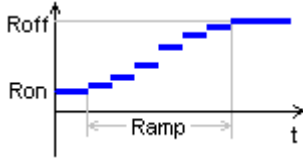
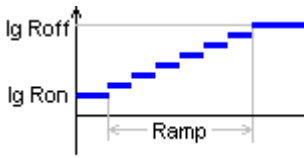
| <b>S</b>   | Logic controlled switch |       | S/S_LCS_SubCir.n15                          |
|--|-------------------------|-------|---|
| Model  | Parameter               | Units | Description                                 |
| <b>SubCir</b>  | File                    |       | File name of subcircuit schematic.          |
|  | Pin1                    |       | Name of subcircuit label connected to pin 1 |
|  | Pin2                    |       | Name of subcircuit label connected to pin 2 |
|  | Pin3                    |       | Name of subcircuit label connected to pin 3 |
|  | Pin4                    |       | Name of subcircuit label connected to pin 4 |
|  | Cmd                     |       | Subcircuit start-up command string          |
|  | IC                      |       | Subcircuit Initial conditions string        |
| <p><b>Subcircuit.</b> See <i>Working with Subcircuits</i> chapter for details.</p> |                         |       |   |



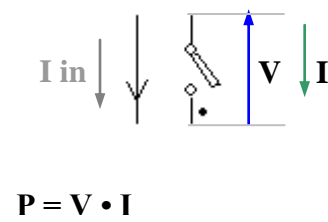
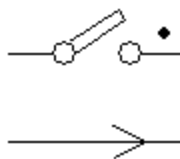
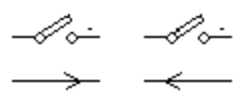
|   |                           |        |                                    |        |   |   |
|---|---------------------------|--------|------------------------------------|--------|---|---|
| <b>S</b>  | Voltage controlled switch | Models | Switch<br>Pulse<br>Steps<br>SubCir | Traces |  |  <p><b>P = V • I</b></p> |
|   | Views                     |        |                                    |        |   |   |
|  |                           |        |                                    |        |   |   |

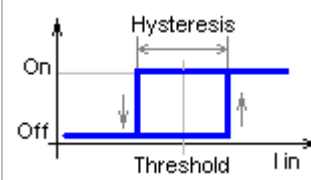
|  |                           |                    |                            |  |  |
|--|---------------------------|--------------------|----------------------------|--|--|
| <b>S</b>   | Voltage controlled switch | S/S_VCS_Switch.nl5 |                            |  |  |
| Model  | Parameter                 | Units              | Description                |  |  |
| <b>Switch</b>  | Threshold                 | V                  | Voltage threshold.         |  |  |
|  | Hysteresis                | V                  | Hysteresis.                |  |  |
|  | Active                    |                    | Active state: Off/On.      |  |  |
|  | IC                        |                    | Initial condition: Off/On. |  |  |
| <p><b>Voltage controlled switch.</b> Switch is set to active or non-active state using following rules:</p> <p style="margin-left: 40px;"> <math>V_{in} &gt; \text{Threshold} + \text{Hysteresis}/2 \dots</math> : active<br/> <math>V_{in} &lt; \text{Threshold} - \text{Hysteresis}/2 \dots</math> : non-active<br/>                 Otherwise ..... : previous state             </p> <p>When calculating DC operating point switch is set to the state defined in “IC”.</p> <p>The following is switching diagram for “Active” = On:</p> |                           |                    |                            |  |  |
|   |                           |                    |                            |  |  |

| <b>S</b>     | Voltage controlled switch  |       | S/S_VCS_Pulse.n15     |
|--------------|--|-------|-----------------------|
| Model        | Parameter  | Units | Description           |
| <b>Pulse</b> | Width  | s     | Pulse width.          |
|              | Threshold  | V     | Voltage threshold.    |
|              | Active   |       | Active state: Off/On. |
|              | <p><b>One-shot pulse generator.</b> When increasing input voltage <math>V_{in}</math> crosses “Threshold” value, switch is set to active state for “Width” time interval. If increasing <math>V_{in}</math> crosses “Threshold” value while switch is in active state, the pulse is restarted.</p> |       |                       |

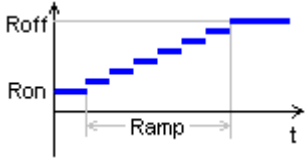
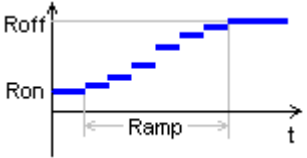
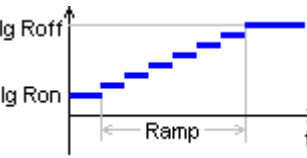
| <b>S</b>  | Voltage controlled switch |       | S/S_VCS_Steps.n15                          |
|---|---------------------------|-------|--|
| Model   | Parameter                 | Units | Description                                |
| <b>Steps</b>  | Threshold                 | V     | Voltage threshold.                         |
|   | Hysteresis                | V     | Hysteresis.                                |
|   | Roff                      | Ohm   | Off state resistance.                      |
|   | Ron                       | Ohm   | On state resistance.                       |
|   | Slope                     |       | Type of resistance change: Linear/Cos/Log. |
|   | Ramp                      | s     | Resistance ramp time.                      |
|   | Steps                     |       | Number of resistance steps in the ramp.    |
|   | IC                        |       | Initial condition: Off/On.                 |
| <p><b>Switch with resistance ramping.</b> When increasing input voltage <math>V_{in}</math> crosses “Threshold” plus “Hysteresis”/2 value, switch resistance starts ramping from “Roff” to “Ron”. When decreasing input voltage <math>V_{in}</math> crosses “Threshold” minus “Hysteresis”/2 value, switch resistance starts ramping from “Ron” to “Roff”.</p> <p>Resistance is changing during “Ramp” time interval, with number of steps specified by “Steps” parameter. If “Steps” = 0, resistance is changed instantly.</p> <p>“Slope” parameter specifies how resistance is changing during the ramp. The following slope types are available (“Steps” = 6):</p> |                           |       |  |
| <div style="display: flex; justify-content: space-around; align-items: flex-end;"> <div style="text-align: center;">  <p><b>Linear</b></p> </div> <div style="text-align: center;">  <p><b>Cos</b></p> </div> <div style="text-align: center;">  <p><b>Log</b></p> </div> </div> <p>When calculating DC operating point switch is set to the state specified in “IC”.</p>                                 |                           |       |  |

| Model         | Parameter   | Units | Description                                 |
|---------------|---|-------|---|
| <b>S</b>      | Voltage controlled switch   |       | S/S_VCS_SubCir.n15                          |
| <b>SubCir</b> | File  |       | File name of subcircuit schematic.          |
|               | Pin1  |       | Name of subcircuit label connected to pin 1 |
|               | Pin2  |       | Name of subcircuit label connected to pin 2 |
|               | Pin3  |       | Name of subcircuit label connected to pin 3 |
|               | Pin4  |       | Name of subcircuit label connected to pin 4 |
|               | Cmd   |       | Subcircuit start-up command string          |
|               | IC  |       | Subcircuit Initial conditions string        |
|               | <b>Subcircuit.</b> See <i>Working with Subcircuits</i> chapter for details. |       |   |

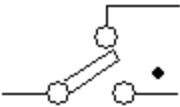
|          |   |        |                                    |        |   |
|----------|---|--------|------------------------------------|--------|---|
| <b>S</b> | <b>Current controlled switch</b>  | Models | Switch<br>Pulse<br>Steps<br>SubCir | Traces |  |
|          |  |        |                                    |        |   |
| Views    |  |        |                                    |        |   |

| <b>S</b>   | Current controlled switch | S/S_CCS_Switch.n15 |                            |  |
|--|---------------------------|--------------------|----------------------------|--|
| Model  | Parameter                 | Units              | Description                |  |
| <b>Switch</b>  | Threshold                 | A                  | Current threshold.         |  |
|  | Hysteresis                | A                  | Hysteresis.                |  |
|  | Active                    |                    | Active state: Off/On.      |  |
|  | IC                        |                    | Initial condition: Off/On. |  |
| <p><b>Current controlled switch.</b> Switch is set to active or non-active state using following rules:</p> <p style="margin-left: 40px;"> <math>I_{in} &gt; \text{Threshold} + \text{Hysteresis}/2 \dots</math> : active<br/> <math>I_{in} &lt; \text{Threshold} - \text{Hysteresis}/2 \dots</math> : non-active<br/>                     Otherwise ..... : previous state                 </p> <p>When calculating DC operating point switch is set to the state defined in "IC".</p> <p>The following is switching diagram for "Active" = On:</p> |                           |                    |                            |  |
|   |                           |                    |                            |  |

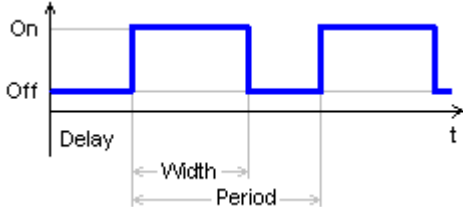
|          |  |       |                       |                   |
|----------|--|-------|-----------------------|-------------------|
| <b>S</b> | Current controlled switch  |       |                       | S/S_CCS_Pulse.n15 |
| Model    | Parameter  | Units | Description           |                   |
| Pulse    | Width  | s     | Pulse width.          |                   |
|          | Threshold  | A     | Current threshold.    |                   |
|          | Active   |       | Active state: Off/On. |                   |
|          | <p><b>One-shot pulse generator.</b> When increasing input current <math>i_{in}</math> crosses “Threshold” value, switch is set to active state for “Width” time interval. If increasing <math>i_{in}</math> crosses “Threshold” value while switch is in active state, the pulse is restarted.</p> |       |                       |                   |

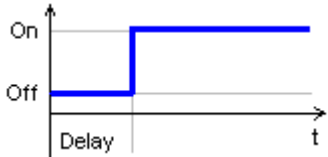
|   |                           |       |  |                   |
|---|---------------------------|-------|--|-------------------|
| <b>S</b>  | Current controlled switch |       |  | S/S_CCS_Steps.n15 |
| Model   | Parameter                 | Units | Description                                |                   |
| Steps   | Threshold                 | A     | Current threshold.                         |                   |
|   | Hysteresis                | A     | Hysteresis.                                |                   |
|   | Roff                      | Ohm   | Off state resistance.                      |                   |
|   | Ron                       | Ohm   | On state resistance.                       |                   |
|   | Slope                     |       | Type of resistance change: Linear/Cos/Log. |                   |
|   | Ramp                      | s     | Resistance ramp time.                      |                   |
|   | Steps                     |       | Number of resistance steps in the ramp.    |                   |
|   | IC                        |       | Initial condition: Off/On.                 |                   |
| <p><b>Switch with resistance ramping.</b> When increasing input current <math>i_{in}</math> crosses “Threshold” plus “Hysteresis”/2 value, switch resistance starts ramping from “Roff” to “Ron”. When decreasing input current <math>i_{in}</math> crosses “Threshold” minus “Hysteresis”/2 value, switch resistance starts ramping from “Ron” to “Roff”.</p> <p>Resistance is changing during “Ramp” time interval, with number of steps specified by “Steps” parameter. If “Steps” = 0, resistance is changed instantly.</p> <p>“Slope” parameter specifies how resistance is changing during the ramp. The following slope types are available (“Steps” = 6):</p> |                           |       |  |                   |
| <div style="display: flex; justify-content: space-around; align-items: flex-end;"> <div style="text-align: center;">  <p><b>Linear</b></p> </div> <div style="text-align: center;">  <p><b>Cos</b></p> </div> <div style="text-align: center;">  <p><b>Log</b></p> </div> </div> <p>When calculating DC operating point switch is set to the state specified in “IC”.</p>                                 |                           |       |  |                   |

| S             | Current controlled switch |       | S/S_CCS_SubCir.n15                          |
|---------------|---------------------------|-------|---|
| Model         | Parameter                 | Units | Description                                 |
| <b>SubCir</b> | File                      |       | File name of subcircuit schematic.          |
|               | Pin1                      |       | Name of subcircuit label connected to pin 1 |
|               | Pin2                      |       | Name of subcircuit label connected to pin 2 |
|               | Pin3                      |       | Name of subcircuit label connected to pin 3 |
|               | Pin4                      |       | Name of subcircuit label connected to pin 4 |
|               | Cmd                       |       | Subcircuit start-up command string          |
|               | IC                        |       | Subcircuit Initial conditions string        |
|               |                           |       |   |

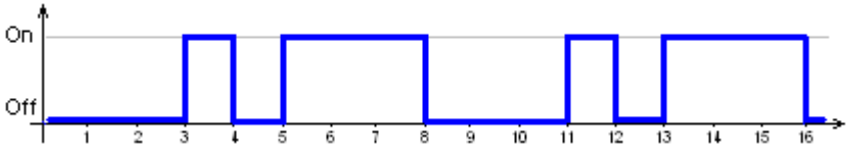
|          |   |        |   |        |
|----------|---|--------|---|--------|
| <b>S</b> | <b>SPDT switch</b>  | Models | Switch<br>Pulse<br>Step<br>List<br>File<br>SubCir | Traces |
|          |  |        |   |        |

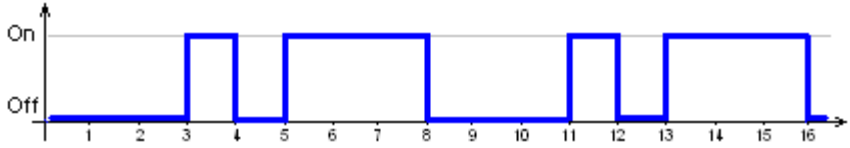
|               |  |                            |                       |  |
|---------------|--|----------------------------|-----------------------|--|
| <b>S</b>      | SPDT switch  | S/S_SPDT_Switch_Switch.n15 |                       |  |
| Model         | Parameter  | Units                      | Description           |  |
| <b>Switch</b> | Switch   |                            | Switch state: Off/On. |  |
|               | <p><b>SPDT (single pole, double throw) switch.</b></p> <p>Off state: "common to pin with dot" - open, "common to another pin" - closed.<br/>                 On state: "common to pin with dot" - closed, "common to another pin" - open.<br/>                 Open state has infinite resistance, closed state has zero resistance.</p> |                            |                       |  |

|   |             |                           |                                  |  |
|---|-------------|---------------------------|----------------------------------|--|
| <b>S</b>  | SPDT switch | S/S_SPDT_Switch_Pulse.n15 |                                  |  |
| Model   | Parameter   | Units                     | Description                      |  |
| <b>Pulse</b>  | Period      | s                         | Period.                          |  |
|   | Width       | s                         | Pulse width.                     |  |
|   | Delay       | s                         | Delay before first pulse starts. |  |
|   | Active      |                           | Active switch state: Off/On.     |  |
| <p><b>SPDT (single pole, double throw) pulse switch.</b></p> <p>Off state: "common to pin with dot" - open, "common to another pin" - closed.<br/>                 On state: "common to pin with dot" - closed, "common to another pin" - open.<br/>                 Open state has infinite resistance, closed state has zero resistance.</p> <p>Switching starts after "Delay" time. Switch is in active state during "Width" time. The following switching diagram shows state of "common to dotted pin" path for "Active" = On. "Common to another pin" always has an opposite state.</p> |             |                           |                                  |  |
|    |             |                           |                                  |  |

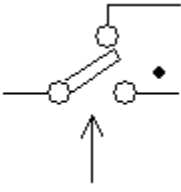
|  |             |       |                              |
|--|-------------|-------|------------------------------|
| <b>S</b>   | SPDT switch |       | S/S_ SPDT_Switch_Step.n15    |
| Model  | Parameter   | Units | Description                  |
| <b>Step</b>  | Delay       | s     | Delay before active state.   |
|  | Active      | s     | Active switch state: Off/On. |
| <p><b>SPDT (single pole, double throw) step switch.</b></p> <p>Off state: "common to pin with dot" - open, "common to another pin" - closed.<br/>                 On state: "common to pin with dot" - closed, "common to another pin" - open.<br/>                 Open state has infinite resistance, closed state has zero resistance.</p> <p>Switch is in active state after "Delay" time. The following switching diagram shows state of "common to dotted pin" path for "Active" = On. "Common to another pin" always has an opposite state.</p>  |             |       |                              |



| S  | SPDT switch |       |                           | V/S_ SPDT_Switch_List.n15 |
|--|-------------|-------|---------------------------|---------------------------|
| Model  | Parameter   | Units | Description               |                           |
| <b>List</b>  | List        |       | Comma-separated string.   |                           |
|  | Cycle       |       | Cycling (repeat): No/Yes. |                           |
|  | Delay       | s     | Delay.                    |                           |
| <p><b>SPDT (single pole, double throw) list switch.</b></p> <p>Off state: “common to pin with dot” - open, “common to another pin” - closed.<br/> On state: “common to pin with dot” - closed, “common to another pin” - open.<br/> Open state has infinite resistance, closed state has zero resistance.</p> <p>Switching sequence is defined in the “List” parameter in the csv (“comma-separated values”) format, as follows:</p> $t_0, s_0, t_1, s_1, \dots, t_n, s_n$ <p><math>s_0 \dots s_n</math> defines switch state: positive number corresponds to On state, zero or negative number - Off state. If <math>t &lt; t_0</math>, switch is in <math>s_0</math> state. At <math>t_0</math> switch is set to <math>s_0</math> state. At <math>t_1</math> switch is set to <math>s_1</math> state, and so on. At <math>t &gt; t_n</math>, switch remains in <math>s_n</math> state if “Cycle” parameter is set to “No”, otherwise states sequence defined in <math>t_0 \dots t_n</math> interval is repeated continuously. In addition, the whole signal is delayed by “Delay” time.</p> <p>Example:<br/> List = 0,0,3,1,4,0,5,1,8,0</p> <p>The following switching diagram shows state of “common to dotted pin” path for “Cycle” = Yes, “Delay” = 0. “Common to another pin” always has an opposite state.</p>  |             |       |                           |                           |

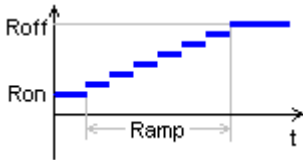
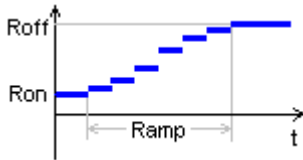
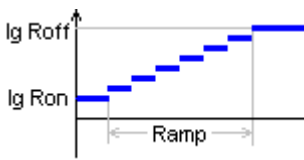
|   |             |       |                           |                           |
|---|-------------|-------|---------------------------|---------------------------|
| <b>S</b>  | SPDT switch |       |                           | V/S_ SPDT_Switch_File.n15 |
| Model   | Parameter   | Units | Description               |                           |
| <b>File</b>   | File        |       | File name.                |                           |
|   | Cycle       |       | Cycling (repeat): No/Yes. |                           |
|   | Delay       | s     | Delay.                    |                           |
| <p><b>SPDT (single pole, double throw) file switch.</b></p> <p>Off state: "common to pin with dot" - open, "common to another pin" - closed.<br/>                 On state: "common to pin with dot" - closed, "common to another pin" - open.<br/>                 Open state has infinite resistance, closed state has zero resistance.</p> <p>Switching sequence is defined in the text file. "File" parameter is a file name, with full path to the file. If the file is located in the same directory as schematic file, the path can be omitted. Switching sequence is defined in the csv ("comma-separated values") format, as follows:</p> <pre>                 &lt;if first line does not start with a number, it is ignored&gt;                 t0,s0                 t1,s1                 .....                 tn,sn             </pre> <p>s0...sn defines switch state: positive number corresponds to On state, zero or negative number - Off state. If <math>t &lt; t_0</math>, switch is in s0 state. At <math>t_0</math> switch is set to s0 state. At <math>t_1</math> switch is set to s1 state, and so on. At <math>t &gt; t_n</math>, switch remains in sn state if "Cycle" parameter is set to "No", otherwise states sequence defined in <math>t_0...t_n</math> interval is repeated continuously. In addition, the whole signal is delayed by "Delay" time.</p> <p>Example:</p> <pre>                 0,0                 3,1                 4,0                 5,1                 8,0             </pre> <p>The following switching diagram shows state of "common to dotted pin" path for "Cycle" = Yes, "Delay" = 0. "Common to another pin" always has an opposite state.</p>  |             |       |                           |                           |

| Model  | Parameter   | Units | Description                                 |
|--------|---|-------|---|
| SubCir | File  |       | File name of subcircuit schematic.          |
|        | Pin1  |       | Name of subcircuit label connected to pin 1 |
|        | Pin2  |       | Name of subcircuit label connected to pin 2 |
|        | Cmd   |       | Subcircuit start-up command string          |
|        | IC  |       | Subcircuit Initial conditions string        |
|        | <b>Subcircuit.</b> See <i>Working with Subcircuits</i> chapter for details. |       |   |

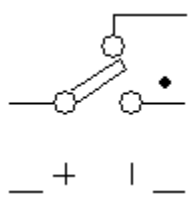
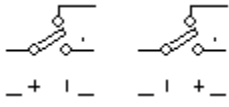
|          |   |        |                                    |        |  |
|----------|---|--------|------------------------------------|--------|--|
| <b>S</b> | SPDT logic controlled switch  | Models | Switch<br>Pulse<br>Steps<br>SubCir | Traces |  |
|          |  |        |                                    |        |  |

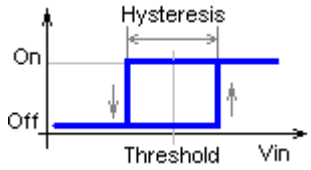
|   |                              |       |                            |                         |
|---|------------------------------|-------|----------------------------|-------------------------|
| <b>S</b>  | SPDT logic controlled switch |       |                            | S/S_SPDT_LCS_Switch.nl5 |
| Model   | Parameter                    | Units | Description                |                         |
| <b>Switch</b>   | Active                       |       | Active state: Off/On.      |                         |
|   | IC                           |       | Initial condition: Off/On. |                         |
| <p><b>Logic controlled switch.</b></p> <p>Off state: “common to pin with dot” - open, “common to another pin” - closed.<br/>                 On state: “common to pin with dot” - closed, “common to another pin” - open.<br/>                 Open state has infinite resistance, closed state has zero resistance.</p> <p>Switch is set to active or non-active state using following rules:</p> <p style="padding-left: 40px;"> <math>V_{in} &gt; \text{logical threshold} \dots : \text{active}</math><br/> <math>V_{in} &lt; \text{logical threshold} \dots : \text{non-active}</math> </p> <p>To see and set logical levels and threshold go to <b>Transient   Settings</b>, or <b>AC   Settings</b>, then click <b>Advanced</b> button.</p> <p>When calculating DC operating point switch is set to the state defined in “IC”.</p> |                              |       |                            |                         |

| S            | SPDT logic controlled switch  |       | S/S _SPDT_LCS_Pulse.n15 |
|--------------|---|-------|-------------------------|
| Model        | Parameter   | Units | Description             |
| <b>Pulse</b> | Width   | s     | Pulse width.            |
|              | Active  |       | Active state: Off/On.   |
|              | <p><b>One-shot pulse generator.</b></p> <p>Off state: “common to pin with dot” - open, “common to another pin” - closed.<br/>           On state: “common to pin with dot” - closed, “common to another pin” - open.<br/>           Open state has infinite resistance, closed state has zero resistance.</p> <p>When increasing input voltage <math>V_{in}</math> crosses logical threshold, switch is set to active state for “Width” time interval. If increasing <math>V_{in}</math> crosses logical threshold value while switch is in active state, the pulse is restarted.</p> <p>To see and set logical levels and threshold go to <b>Transient   Settings</b>, or <b>AC   Settings</b>, then click <b>Advanced</b> button.</p> |       |                         |

| S     |  | SPDT logic controlled switch |  | S/S_SPDT_LCS_Steps.n15 |
|-------|--|------------------------------|--|------------------------|
| Model | Parameter  | Units                        | Description                                |                        |
| Steps | Roff   | Ohm                          | Off state resistance.                      |                        |
|       | Ron  | Ohm                          | On state resistance.                       |                        |
|       | Slope  |                              | Type of resistance change: Linear/Cos/Log. |                        |
|       | Ramp   | s                            | Resistance ramp time.                      |                        |
|       | Steps  |                              | Number of resistance steps in the ramp.    |                        |
|       | IC   |                              | Initial condition: Off/On.                 |                        |
|       | <p><b>Switch with resistance ramping.</b></p> <p>When increasing input voltage <math>V_{in}</math> crosses logical threshold, resistance of “common to pin with dot” path starts ramping from “Roff” to “Ron”, resistance of “common to another pin” path starts ramping from “Ron” to “Roff”</p> <p>When decreasing input voltage <math>V_{in}</math> crosses logical threshold, resistance of “common to pin with dot” path starts ramping from “Ron” to “Roff”, resistance of “common to another pin” path starts ramping from “Roff” to “Ron”</p> <p>Resistance is changing during “Ramp” time interval, with number of steps specified by “Steps” parameter. If “Steps” = 0, resistance is changed instantly.</p> <p>“Slope” parameter specifies how resistance is changing during the ramp. The following slope types are available (“Steps” = 6):</p> <div style="display: flex; justify-content: space-around; align-items: flex-end;"> <div style="text-align: center;">  <p><b>Linear</b></p> </div> <div style="text-align: center;">  <p><b>Cos</b></p> </div> <div style="text-align: center;">  <p><b>Log</b></p> </div> </div> <p>To see and set logical levels and threshold go to <b>Transient   Settings</b>, or <b>AC   Settings</b>, then click <b>Advanced</b> button.</p> <p>When calculating DC operating point switch is set to the state specified in “IC”.</p> |                              |  |                        |

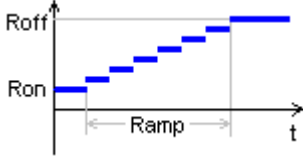
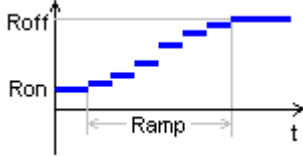
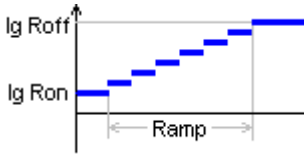
| S      | SPDT logic controlled switch |       |   |
|--------|------------------------------|-------|---|
| Model  | Parameter                    | Units | Description                                 |
| SubCir | File                         |       | File name of subcircuit schematic.          |
|        | Pin1                         |       | Name of subcircuit label connected to pin 1 |
|        | Pin2                         |       | Name of subcircuit label connected to pin 2 |
|        | Pin3                         |       | Name of subcircuit label connected to pin 3 |
|        | Pin4                         |       | Name of subcircuit label connected to pin 4 |
|        | Cmd                          |       | Subcircuit start-up command string          |
|        | IC                           |       | Subcircuit Initial conditions string        |
|        |                              |       |   |

|          |   |        |                                    |        |
|----------|---|--------|------------------------------------|--------|
| <b>S</b> | <b>SPDT voltage controlled switch</b>   | Models | Switch<br>Pulse<br>Steps<br>SubCir | Traces |
|          |  |        |                                    |        |
| Views    |  |        |                                    |        |

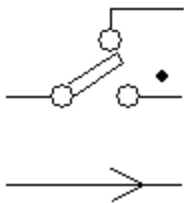
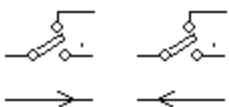
|   |                                       |                         |                            |  |
|---|---------------------------------------|-------------------------|----------------------------|--|
| <b>S</b>  | <b>SPDT voltage controlled switch</b> | S/S_SPDT_VCS_Switch.nl5 |                            |  |
| Model   | Parameter                             | Units                   | Description                |  |
| <b>Switch</b>   | Threshold                             | V                       | Voltage threshold.         |  |
|   | Hysteresis                            | V                       | Hysteresis.                |  |
|   | Active                                |                         | Active state: Off/On.      |  |
|   | IC                                    |                         | Initial condition: Off/On. |  |
| <p><b>Voltage controlled switch.</b></p> <p>Off state: “common to pin with dot” - open, “common to another pin” - closed.<br/>                 On state: “common to pin with dot” - closed, “common to another pin” - open.<br/>                 Open state has infinite resistance, closed state has zero resistance.</p> <p>Switch is set to active or non-active state using following rules:</p> <p style="margin-left: 40px;"> <math>V_{in} &gt; \text{Threshold} + \text{Hysteresis}/2 \dots</math> : active<br/> <math>V_{in} &lt; \text{Threshold} - \text{Hysteresis}/2 \dots</math> : non-active<br/>                     Otherwise . . . . . : previous state                 </p> <p>When calculating DC operating point switch is set to the state defined in “IC”.</p> <p>The following is switching diagram for “common to pin with dot” path, “Active” = On:<br/>                 The following switching diagram shows state of “common to dotted pin” path for “Active” = On.<br/>                 “Common to another pin” always has an opposite state.</p> <div style="text-align: center;">  </div> |                                       |                         |                            |  |

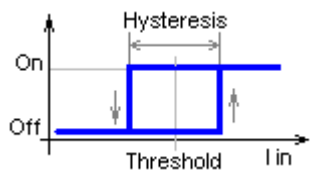


| S            | SPDT voltage controlled switch  |       | S/S_SPDT_VCS_Pulse.n15 |
|--------------|---|-------|------------------------|
| Model        | Parameter   | Units | Description            |
| <b>Pulse</b> | Width   | s     | Pulse width.           |
|              | Threshold   | V     | Voltage threshold.     |
|              | Active  |       | Active state: Off/On.  |
|              | <p><b>One-shot pulse generator.</b></p> <p>Off state: “common to pin with dot” - open, “common to another pin” - closed.<br/>           On state: “common to pin with dot” - closed, “common to another pin” - open.<br/>           Open state has infinite resistance, closed state has zero resistance.</p> <p>When increasing input voltage <math>V_{in}</math> crosses “Threshold” value, switch is set to active state for “Width” time interval. If increasing <math>V_{in}</math> crosses “Threshold” value while switch is in active state, the pulse is restarted.</p> |       |                        |

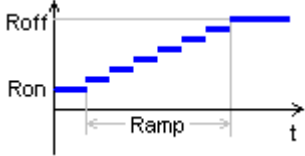
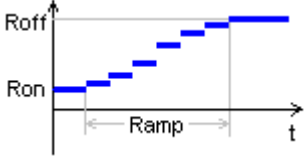
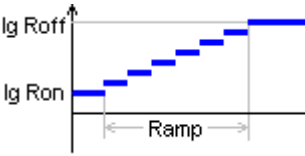
| S  | SPDT voltage controlled switch |       | S/S_SPDT_VCS_Steps.nl5                     |
|--|--------------------------------|-------|--|
| Model  | Parameter                      | Units | Description                                |
| <b>Steps</b>   | Threshold                      | V     | Voltage threshold.                         |
|  | Hysteresis                     | V     | Hysteresis.                                |
|  | Roff                           | Ohm   | Off state resistance.                      |
|  | Ron                            | Ohm   | On state resistance.                       |
|  | Slope                          |       | Type of resistance change: Linear/Cos/Log. |
|  | Ramp                           | s     | Resistance ramp time.                      |
|  | Steps                          |       | Number of resistance steps in the ramp.    |
|  | IC                             |       | Initial condition: Off/On.                 |
| <p><b>Switch with resistance ramping.</b></p> <p>When increasing input voltage <math>V_{in}</math> crosses “Threshold” plus “Hysteresis”/2 value, resistance of “common to pin with dot” path starts ramping from “Roff” to “Ron”, resistance of “common to another pin” path starts ramping from “Ron” to “Roff”.</p> <p>When decreasing input voltage <math>V_{in}</math> crosses “Threshold” minus “Hysteresis”/2 value, resistance of “common to pin with dot” path starts ramping from “Ron” to “Roff”, resistance of “common to another pin” path starts ramping from “Roff” to “Ron”</p> <p>Resistance is changing during “Ramp” time interval, with number of steps specified by “Steps” parameter. If “Steps” = 0, resistance is changed instantly.</p> <p>“Slope” parameter specifies how resistance is changing during the ramp. The following slope types are available (“Steps” = 6):</p> <div style="display: flex; justify-content: space-around; align-items: flex-end;"> <div style="text-align: center;">  <p><b>Linear</b></p> </div> <div style="text-align: center;">  <p><b>Cos</b></p> </div> <div style="text-align: center;">  <p><b>Log</b></p> </div> </div> <p>When calculating DC operating point switch is set to the state specified in “IC”.</p> |                                |       |  |

| S   |           | SPDT voltage controlled switch |   |
|---|-----------|--------------------------------|---|
| Model   | Parameter | Units                          | Description                                 |
| SubCir  | File      |                                | File name of subcircuit schematic.          |
|   | Pin1      |                                | Name of subcircuit label connected to pin 1 |
|   | Pin2      |                                | Name of subcircuit label connected to pin 2 |
|   | Pin3      |                                | Name of subcircuit label connected to pin 3 |
|   | Pin4      |                                | Name of subcircuit label connected to pin 4 |
|   | Cmd       |                                | Subcircuit start-up command string          |
|   | IC        |                                | Subcircuit Initial conditions string        |
| <b>Subcircuit.</b> See <i>Working with Subcircuits</i> chapter for details. |           |                                |   |

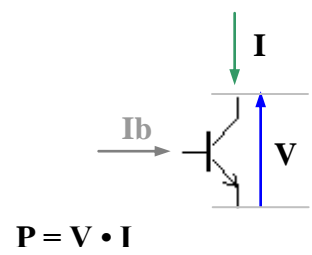
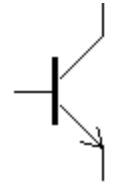
|          |   |        |                                    |        |
|----------|---|--------|------------------------------------|--------|
| <b>S</b> | <b>SPDT current controlled switch</b>   | Models | Switch<br>Pulse<br>Steps<br>SubCir | Traces |
|          |  |        |                                    |        |
| Views    |  |        |                                    |        |

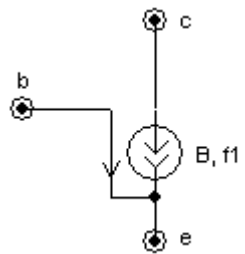
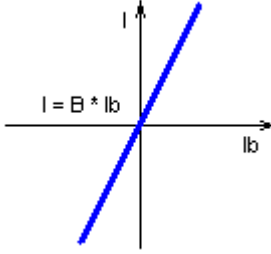
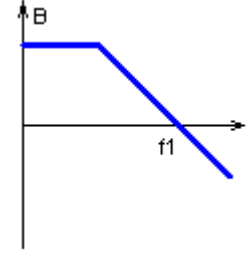
|  |                                |                         |                            |  |
|--|--------------------------------|-------------------------|----------------------------|--|
| <b>S</b>   | SPDT current controlled switch | S/S_SPDT_CCS_Switch.n15 |                            |  |
| Model  | Parameter                      | Units                   | Description                |  |
| <b>Switch</b>  | Threshold                      | A                       | Current threshold.         |  |
|  | Hysteresis                     | A                       | Hysteresis.                |  |
|  | Active                         |                         | Active state: Off/On.      |  |
|  | IC                             |                         | Initial condition: Off/On. |  |
| <p><b>Current controlled switch.</b></p> <p>Off state: "common to pin with dot" - open, "common to another pin" - closed.<br/>                 On state: "common to pin with dot" - closed, "common to another pin" - open.<br/>                 Open state has infinite resistance, closed state has zero resistance.</p> <p>Switch is set to active or non-active state using following rules:</p> <p style="padding-left: 40px;"> <math>I_{in} &gt; \text{Threshold} + \text{Hysteresis}/2 \dots</math> : active<br/> <math>I_{in} &lt; \text{Threshold} - \text{Hysteresis}/2 \dots</math> : non-active<br/>                     Otherwise ..... : previous state                 </p> <p>When calculating DC operating point switch is set to the state defined in "IC".</p> <p>The following is switching diagram for "common to pin with dot" path, "Active" = On:<br/>                     The following switching diagram shows state of "common to dotted pin" path for "Active" = On.<br/>                     "Common to another pin" always has an opposite state.</p> <div style="text-align: center;">  </div> |                                |                         |                            |  |

| S            | SPDT current controlled switch  |       | S/S_SPDT _CCS_Pulse.n15 |
|--------------|---|-------|-------------------------|
| Model        | Parameter   | Units | Description             |
| <b>Pulse</b> | Width   | s     | Pulse width.            |
|              | Threshold   | A     | Current threshold.      |
|              | Active  |       | Active state: Off/On.   |
|              | <p><b>One-shot pulse generator.</b></p> <p>Off state: “common to pin with dot” - open, “common to another pin” - closed.<br/>           On state: “common to pin with dot” - closed, “common to another pin” - open.<br/>           Open state has infinite resistance, closed state has zero resistance.</p> <p>When increasing input current <math>I_{in}</math> crosses “Threshold” value, switch is set to active state for “Width” time interval. If increasing <math>I_{in}</math> crosses “Threshold” value while switch is in active state, the pulse is restarted.</p> |       |                         |

| S   | SPDT current controlled switch |       | S/S_SPDT_CCS_Steps.nl5                     |
|---|--------------------------------|-------|--|
| Model   | Parameter                      | Units | Description                                |
| <b>Steps</b>  | Threshold                      | A     | Current threshold.                         |
|   | Hysteresis                     | A     | Hysteresis.                                |
|   | Roff                           | Ohm   | Off state resistance.                      |
|   | Ron                            | Ohm   | On state resistance.                       |
|   | Slope                          |       | Type of resistance change: Linear/Cos/Log. |
|   | Ramp                           | s     | Resistance ramp time.                      |
|   | Steps                          |       | Number of resistance steps in the ramp.    |
|   | IC                             |       | Initial condition: Off/On.                 |
| <p><b>Switch with resistance ramping.</b></p> <p>When increasing input current <math>I_{in}</math> crosses “Threshold” plus “Hysteresis”/2 value, resistance of “common to pin with dot” path starts ramping from “Roff” to “Ron”, resistance of “common to another pin” path starts ramping from “Ron” to “Roff”.</p> <p>When decreasing input current <math>I_{in}</math> crosses “Threshold” minus “Hysteresis”/2 value, resistance of “common to pin with dot” path starts ramping from “Ron” to “Roff”, resistance of “common to another pin” path starts ramping from “Roff” to “Ron”.</p> <p>Resistance is changing during “Ramp” time interval, with number of steps specified by “Steps” parameter. If “Steps” = 0, resistance is changed instantly.</p> <p>“Slope” parameter specifies how resistance is changing during the ramp. The following slope types are available (“Steps” = 6):</p> <div style="display: flex; justify-content: space-around; align-items: flex-end;"> <div style="text-align: center;">  <p><b>Linear</b></p> </div> <div style="text-align: center;">  <p><b>Cos</b></p> </div> <div style="text-align: center;">  <p><b>Log</b></p> </div> </div> <p>When calculating DC operating point switch is set to the state specified in “IC”.</p> |                                |       |  |

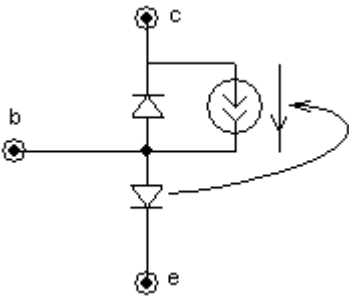
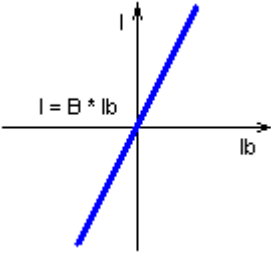
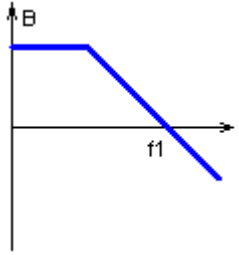
| S   |           | SPDT current controlled switch |   |
|---|-----------|--------------------------------|---|
| Model   | Parameter | Units                          | Description                                 |
| SubCir  | File      |                                | File name of subcircuit schematic.          |
|   | Pin1      |                                | Name of subcircuit label connected to pin 1 |
|   | Pin2      |                                | Name of subcircuit label connected to pin 2 |
|   | Pin3      |                                | Name of subcircuit label connected to pin 3 |
|   | Pin4      |                                | Name of subcircuit label connected to pin 4 |
|   | Cmd       |                                | Subcircuit start-up command string          |
|   | IC        |                                | Subcircuit Initial conditions string        |
| <b>Subcircuit.</b> See <i>Working with Subcircuits</i> chapter for details. |           |                                |   |

|          |   |        |  |        |   |
|----------|---|--------|--|--------|---|
| <b>T</b> | NPN transistor  | Models | Linear<br>Switch<br>Transistor<br>SubCir | Traces |  <p><math>P = V \cdot I</math></p> |
|          |  |        |  |        |   |

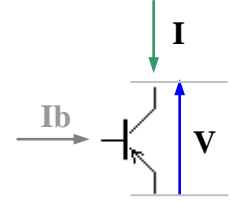
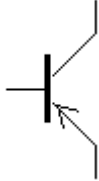
|  |                |  |                                       |   |
|--|----------------|--|---------------------------------------|---|
| <b>T</b>   | NPN transistor | T/T_NPN_Linear.nl5   |                                       |   |
| Model  | Parameter      | Units  | Description                           |   |
| <b>Linear</b>  | B              | A/A  | Gain (beta)                           |   |
|  | f1             | Hz   | Unit gain frequency.                  |   |
|  | IC             | A  | Initial condition: collector current. |   |
| <p><b>Linear BJT transistor.</b> Current controlled current source with specified bandwidth. “B” is open loop gain (beta). Frequency response consists of one pole, “f1” is unit gain frequency. “B” and “f1” can be set to infinity (“inf”).</p> <p>When calculating DC operating point, if “f1” is not infinity and “IC” is defined, collector current is set to specified output current “IC”. If “IC” is blank, static characteristic is used.</p> |                |  |                                       |   |
|  <p style="text-align: center;"><b>Equivalent schematic</b></p>   |                |  <p style="text-align: center;"><b>Static characteristic</b></p> |                                       |  <p style="text-align: center;"><b>AC response</b></p> |

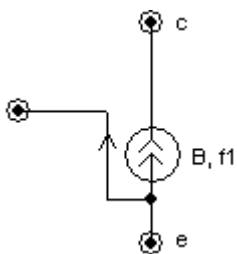
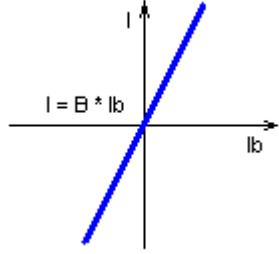
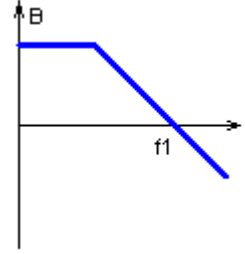


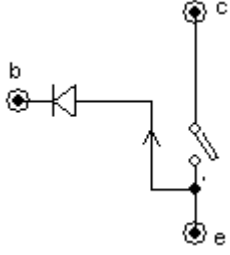
| T  | NPN transistor |       | T/T_NPN_Switch.n15                               |
|--|----------------|-------|--|
| Model  | Parameter      | Units | Description                                      |
| <b>Switch</b>  | Vbe            | V     | Forward voltage drop of base-emitter diode.      |
|  | IC             |       | Initial condition of base-emitter diode: Off/On. |
| <p><b>BJT transistor switch.</b> Current controlled switch with a base-emitter diode. Switch is closed if diode current is non-zero.</p> <p>When calculating DC operating point the diode is set to the state specified in "IC".</p> <div data-bbox="440 527 667 779" style="text-align: center;"> </div> <p><b>Equivalent schematic</b></p> |                |       |  |

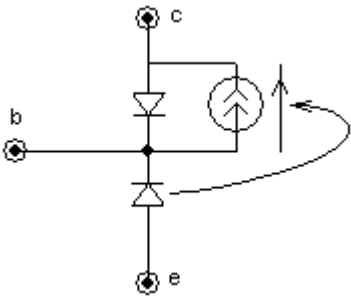
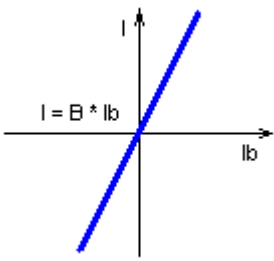
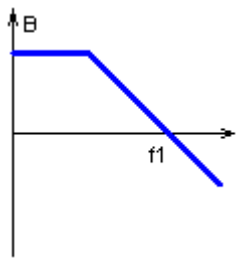
| T  |           | NPN transistor  |  | T/T_NPN_Transistor.nl5  |
|--|-----------|---|--|---|
| Model  | Parameter | Units   | Description  |   |
| Transistor   | B         | A/A   | Gain (beta)  |   |
|  | f1        | Hz  | Unit gain frequency.                               |   |
|  | Vbe       | V   | Forward voltage drop of base-emitter diode.        |   |
|  | Vsat      | V   | Collector-emitter saturation voltage drop.         |   |
|  | IC        | A   | Initial condition: collector current.              |   |
|  | ICbe      |   | Initial condition of base-emitter diode: Off/On.   |   |
|  | ICbc      |   | Initial condition of base-collector diode: Off/On. |   |
| <p><b>BJT transistor.</b> Simplified Ebers-Moll BJT transistor model with saturation. It consists of two diodes (base-emitter and base-collector), and current source controlled by current through base-emitter diode with gain “alpha”:</p> $\alpha = \frac{\beta}{1 + \beta}$ <p>If collector-emitter voltage is higher than “Vsat”, base-collector diode is open, transistor is not saturated, and behaves as “Linear” model (current controlled current source with specified bandwidth). “B” is open loop gain (beta). Low signal frequency response consists of one pole, “f1” is unit gain frequency. “B” and “f1” can be set to infinity (“inf”).</p> <p>If collector voltage drops below “Vsat”, base-collector diode is closed, and transistor is saturated: collector-emitter voltage is equal to “Vsat”.</p> <p>When calculating DC operating point, if “f1” is not infinity and “IC” is defined, collector current is set to specified output current “IC”. If “IC” is blank, static characteristic is used. Base-emitter diode is set to the state specified in “ICbe”, Base-collector diode is set to the state specified in “ICbc”.</p> |           |   |  |   |
|  <p>Equivalent schematic</p>  |           |  <p>Non-saturated static characteristic</p> |  |  <p>Low signal AC response</p> |

| <b>T</b>  |           | NPN transistor |   | T/T_NPN_SubCir.n15 |
|---|-----------|----------------|---|--------------------|
| Model   | Parameter | Units          | Description                                 |                    |
| <b>SubCir</b>   | File      |                | File name of subcircuit schematic.          |                    |
|   | Pin1      |                | Name of subcircuit label connected to pin 1 |                    |
|   | Pin2      |                | Name of subcircuit label connected to pin 2 |                    |
|   | Pin3      |                | Name of subcircuit label connected to pin 3 |                    |
|   | Cmd       |                | Subcircuit start-up command string          |                    |
|   | IC        |                | Subcircuit Initial conditions string        |                    |
| <b>Subcircuit.</b> See <i>Working with Subcircuits</i> chapter for details. |           |                |   |                    |

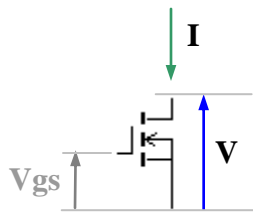
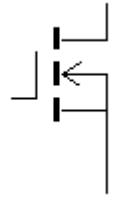
|          |   |        |  |        |   |
|----------|---|--------|--|--------|---|
| <b>T</b> | PNP transistor  | Models | Linear<br>Switch<br>Transistor<br>SubCir | Traces |  <p><math>P = V \cdot I</math></p> |
|          |  |        |  |        |   |

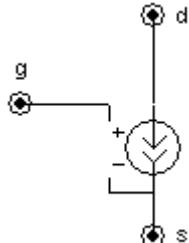
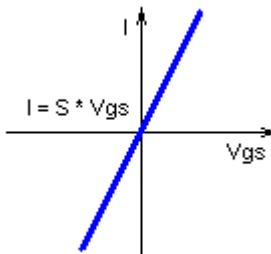
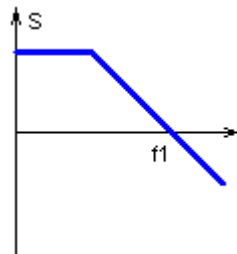
|  |                |  |                                       |   |
|--|----------------|--|---------------------------------------|---|
| <b>T</b>   | PNP transistor | T/T_PNP_Linear.n15   |                                       |   |
| Model  | Parameter      | Units  | Description                           |   |
| Linear   | B              | A/A  | Gain (beta)                           |   |
|  | f1             | Hz   | Unit gain frequency.                  |   |
|  | IC             | A  | Initial condition: collector current. |   |
| <p><b>Linear BJT transistor.</b> Current controlled current source with specified bandwidth. “B” is open loop gain (beta). Frequency response consists of one pole, “f1” is unit gain frequency. “B” and “f1” can be set to infinity (“inf”).</p> <p>When calculating DC operating point, if “f1” is not infinity and “IC” is defined, collector current is set to specified output current “IC”. If “IC” is blank, static characteristic is used.</p> |                |  |                                       |   |
|   |                |  |                                       |  |
| Equivalent schematic   |                | Static characteristic  |                                       | AC response   |

| T  |           | PNP transistor |  | T/T_PNP_Switch.nl5 |
|--|-----------|----------------|--|--------------------|
| Model  | Parameter | Units          | Description                                      |                    |
| Switch   | Vbe       | V              | Forward voltage drop of base-emitter diode.      |                    |
|  | IC        |                | Initial condition of base-emitter diode: Off/On. |                    |
| <p><b>BJT transistor switch.</b> Current controlled switch with a base-emitter diode. Switch is closed if diode current is non-zero.</p> <p>When calculating DC operating point the diode is set to the state specified in "IC".</p> |           |                |  |                    |
|  <p style="text-align: center;"><b>Equivalent schematic</b></p>   |           |                |  |                    |

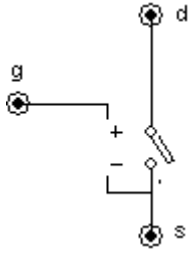
| T   |           | PNP transistor  |  | T/T_PNP_Transistor.nl5  |
|---|-----------|---|--|---|
| Model   | Parameter | Units   | Description  |   |
| Transistor  | B         | A/A   | Gain (beta)  |   |
|   | f1        | Hz  | Unit gain frequency.                               |   |
|   | Vbe       | V   | Forward voltage drop of base-emitter diode.        |   |
|   | Vsat      | V   | Collector-emitter saturation voltage drop.         |   |
|   | IC        | A   | Initial condition: collector current.              |   |
|   | ICbe      |   | Initial condition of base-emitter diode: Off/On.   |   |
|   | ICbc      |   | Initial condition of base-collector diode: Off/On. |   |
| <p><b>BJT transistor.</b> Simplified Ebers-Moll BJT transistor model with saturation. It consists of two diodes (base-emitter and base-collector), and current source controlled by current through base-emitter diode with gain “alpha”:</p> $\alpha = \frac{\beta}{1 + \beta}$ <p>If collector-emitter voltage is negative and less than -“Vsat”, base-collector diode is open, transistor is not saturated, and behaves as “Linear” model (current controlled current source with specified bandwidth). “B” is open loop gain (beta). Low signal frequency response consists of one pole, “f1” is unit gain frequency. “B” and “f1” can be set to infinity (“inf”).</p> <p>If collector voltage is higher than -“Vsat”, base-collector diode is closed, and transistor is saturated: collector-emitter voltage is equal to -“Vsat”.</p> <p>When calculating DC operating point, if “f1” is not infinity and “IC” is defined, collector current is set to specified output current “IC”. If “IC” is blank, static characteristic is used. Base-emitter diode is set to the state specified in “ICbe”. Base-collector diode is set to the state specified in “ICbc”.</p> |           |   |  |   |
|  <p>Equivalent schematic</p>   |           |  <p>Non-saturated static characteristic</p> |  |  <p>Low signal AC response</p> |

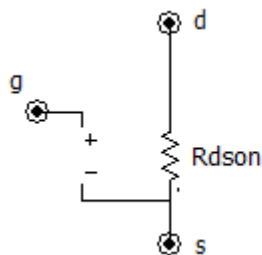
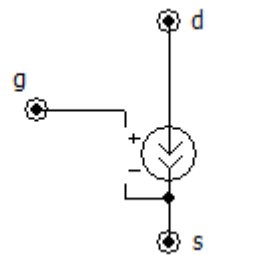
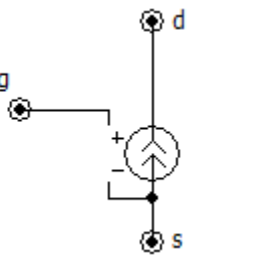
| <b>T</b>  |           | PNP transistor |   | T/T_PNP_SubCir.n15 |
|---|-----------|----------------|---|--------------------|
| Model   | Parameter | Units          | Description                                 |                    |
| <b>SubCir</b>   | File      |                | File name of subcircuit schematic.          |                    |
|   | Pin1      |                | Name of subcircuit label connected to pin 1 |                    |
|   | Pin2      |                | Name of subcircuit label connected to pin 2 |                    |
|   | Pin3      |                | Name of subcircuit label connected to pin 3 |                    |
|   | Cmd       |                | Subcircuit start-up command string          |                    |
|   | IC        |                | Subcircuit Initial conditions string        |                    |
| <b>Subcircuit.</b> See <i>Working with Subcircuits</i> chapter for details. |           |                |   |                    |

|          |   |        |                                   |        |   |
|----------|---|--------|-----------------------------------|--------|---|
| <b>T</b> | <b>N-FET</b>  | Models | Linear<br>Switch<br>FET<br>SubCir | Traces |  <p><b>P = V • I</b></p> |
|          |  |        |                                   |        |   |

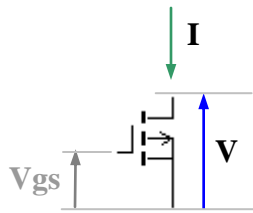
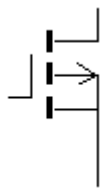
|  |           |  |                                   |   |
|--|-----------|--|-----------------------------------|---|
| <b>T</b>   | N-FET     | T/T_NFET_Linear.nl5  |                                   |   |
| Model  | Parameter | Units  | Description                       |   |
| <b>Linear</b>  | S         | A/V  | Slope                             |   |
|  | f1        | Hz   | Unit gain frequency.              |   |
|  | IC        | A  | Initial condition: drain current. |   |
| <p><b>Linear FET transistor.</b> Voltage controlled current source with specified bandwidth. “S” is open loop slope. Frequency response consists of one pole, “f1” is unit gain frequency. “S” and “f1” can be set to infinity (“inf”).</p> <p>When calculating DC operating point, if “f1” is not infinity and “IC” is defined, drain current is set to specified output current “IC”. If “IC” is blank, static characteristic is used.</p> |           |  |                                   |   |
|  <p><b>Equivalent schematic</b></p>   |           |  <p><b>Static characteristic</b></p> |                                   |  <p><b>AC response</b></p> |

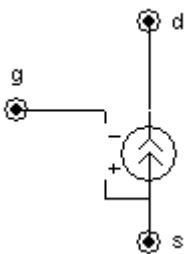
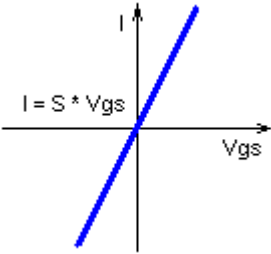
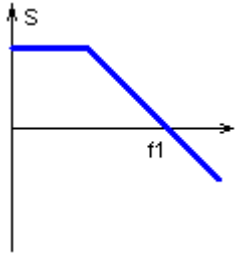


| T      |   | N-FET |  | T/T_NFET_Switch.nl5 |
|--------|---|-------|--|---------------------|
| Model  | Parameter   | Units | Description                              |                     |
| Switch | Vth   | V     | Threshold.                               |                     |
|        | IC  |       | Initial condition of the switch: Off/On. |                     |
|        | <p><b>FET switch.</b> Voltage controlled switch. Switch is closed if gate-source voltage exceeds threshold "Vth".</p> <p>When calculating DC operating point switch is set to the state specified in "IC".</p>  <p><b>Equivalent schematic</b></p> |       |  |                     |

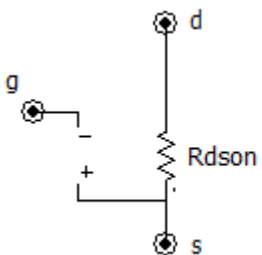
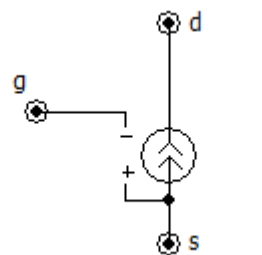
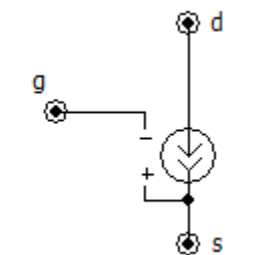
| T  |           | N-FET |                   | T/T_NFET_FET.n15 |
|--|-----------|-------|-------------------|------------------|
| Model  | Parameter | Units | Description       |                  |
| FET  | S         | A/V   | Slope.            |                  |
|  | Vth       | V     | Threshold.        |                  |
|  | Rdson     | Ohm   | Rdson resistance. |                  |
| <p><b>FET transistor.</b> The model has 3 modes of operation.</p> <ol style="list-style-type: none"> <li>1. <math>V_{gs} \leq V_{th}</math>: . . . . . : <math>I = 0</math> (open)</li> <li>2. <math>V_{gs} &gt; V_{th}</math>, <math>V_{ds} \leq (V_{gs} - V_{th}) * S * R_{dson}</math> . . : <math>V = I * R_{dson}</math> (resistor)</li> <li>3. <math>V_{gs} &gt; V_{th}</math>, <math>V_{ds} &gt; (V_{gs} - V_{th}) * S * R_{dson}</math> . . : <math>I = (V_{gs} - V_{th}) * S</math> (current source)</li> </ol> <p>FET works similar for positive and negative drain-source voltage, current direction changes accordingly. Equivalent schematics (<math>V_{gs} &gt; V_{th}</math>):</p> <div style="display: flex; justify-content: space-around; align-items: flex-end;"> <div style="text-align: center;">  <p><math>V_{ds} \leq (V_{gs} - V_{th}) * S * R_{dson}</math></p> </div> <div style="text-align: center;">  <p><math>V_{ds} &gt; (V_{gs} - V_{th}) * S * R_{dson}</math><br/><math>V_{ds} &gt; 0</math></p> </div> <div style="text-align: center;">  <p><math>V_{ds} &gt; (V_{gs} - V_{th}) * S * R_{dson}</math><br/><math>V_{ds} &lt; 0</math></p> </div> </div> |           |       |                   |                  |

| T  |           | N-FET |   | T/T_NFET_SubCir.n15 |
|--|-----------|-------|---|---------------------|
| Model  | Parameter | Units | Description                                 |                     |
| SubCir   | File      |       | File name of subcircuit schematic.          |                     |
|  | Pin1      |       | Name of subcircuit label connected to pin 1 |                     |
|  | Pin2      |       | Name of subcircuit label connected to pin 2 |                     |
|  | Pin3      |       | Name of subcircuit label connected to pin 3 |                     |
|  | Cmd       |       | Subcircuit start-up command string          |                     |
|  | IC        |       | Subcircuit Initial conditions string        |                     |
| <p><b>Subcircuit.</b> See <i>Working with Subcircuits</i> chapter for details.</p> |           |       |   |                     |

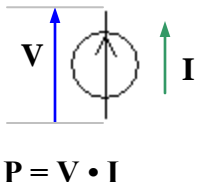
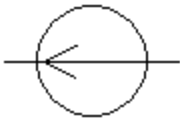
|          |   |        |                                   |        |   |
|----------|---|--------|-----------------------------------|--------|---|
| <b>T</b> | <b>P-FET</b>  | Models | Linear<br>Switch<br>FET<br>SubCir | Traces |  <p><b>P = V • I</b></p> |
|          |  |        |                                   |        |   |

|  |           |  |                                   |   |
|--|-----------|--|-----------------------------------|---|
| <b>T</b>   | P-FET     | T/T_PFET_Linear.n15  |                                   |   |
| Model  | Parameter | Units  | Description                       |   |
| <b>Linear</b>  | S         | A/V  | Slope                             |   |
|  | f1        | Hz   | Unit gain frequency.              |   |
|  | IC        | A  | Initial condition: drain current. |   |
| <p><b>Linear FET transistor.</b> Voltage controlled current source with specified bandwidth. “S” is open loop slope. Frequency response consists of one pole, “f1” is unit gain frequency. “S” and “f1” can be set to infinity (“inf”).</p> <p>When calculating DC operating point, if “f1” is not infinity and “IC” is defined, drain current is set to specified output current “IC”. If “IC” is blank, static characteristic is used.</p> |           |  |                                   |   |
|  <p><b>Equivalent schematic</b></p>   |           |  <p><b>Static characteristic</b></p> |                                   |  <p><b>AC response</b></p> |

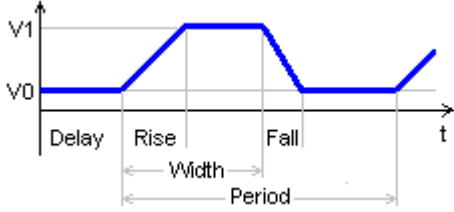
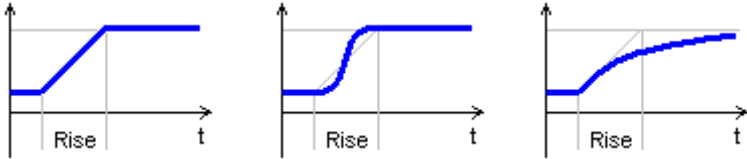
| T      | P-FET   |       | T/T_PFET_Switch.n15                      |
|--------|---|-------|--|
| Model  | Parameter   | Units | Description                              |
| Switch | Vth   | V     | Threshold.                               |
|        | IC  |       | Initial condition of the switch: Off/On. |
|        | <p><b>FET switch.</b> Voltage controlled switch. Switch is closed if gate-source voltage is less than threshold "Vth".</p> <p>When calculating DC operating point switch is set to the state specified in "IC".</p> <div data-bbox="430 535 617 787" style="text-align: center;"> </div> <p><b>Equivalent schematic</b></p> |       |  |

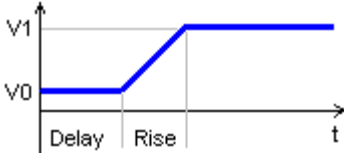
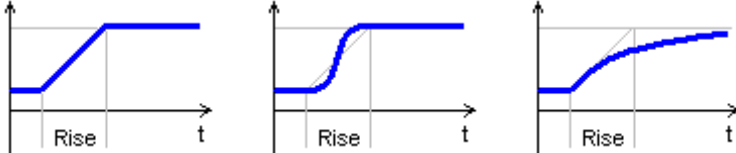
| T   |           | P-FET |                   | T/T_PFET_FET.n15 |
|---|-----------|-------|-------------------|------------------|
| Model   | Parameter | Units | Description       |                  |
| FET   | S         | A/V   | Slope.            |                  |
|   | Vth       | V     | Threshold.        |                  |
|   | Rdson     | Ohm   | Rdson resistance. |                  |
| <p><b>FET transistor.</b> The model has 3 modes of operation.</p> <ol style="list-style-type: none"> <li>1. <math>V_{gs} \geq V_{th}</math>: ..... : <math>I = 0</math> (open)</li> <li>2. <math>V_{gs} &lt; V_{th}</math>, <math>V_{ds} \geq (V_{gs} - V_{th}) * S * R_{dson}</math> .. : <math>V = I * R_{dson}</math> (resistor)</li> <li>3. <math>V_{gs} &lt; V_{th}</math>, <math>V_{ds} &lt; (V_{gs} - V_{th}) * S * R_{dson}</math> ... : <math>I = (V_{gs} - V_{th}) * S</math> (current source)</li> </ol> <p>FET works similar for positive and negative drain-source voltage, current direction changes accordingly. Equivalent schematics (<math>V_{gs} &lt; V_{th}</math>):</p> <div style="display: flex; justify-content: space-around; align-items: flex-end;"> <div style="text-align: center;">  <p><math>V_{ds} \geq (V_{gs} - V_{th}) * S * R_{dson}</math></p> </div> <div style="text-align: center;">  <p><math>V_{ds} &lt; (V_{gs} - V_{th}) * S * R_{dson}</math><br/><math>V_{ds} &lt; 0</math></p> </div> <div style="text-align: center;">  <p><math>V_{ds} &lt; (V_{gs} - V_{th}) * S * R_{dson}</math><br/><math>V_{ds} &gt; 0</math></p> </div> </div> |           |       |                   |                  |

| T  |           | P-FET |   | T/T_PFET_SubCir.n15 |
|--|-----------|-------|---|---------------------|
| Model  | Parameter | Units | Description                                 |                     |
| SubCir   | File      |       | File name of subcircuit schematic.          |                     |
|  | Pin1      |       | Name of subcircuit label connected to pin 1 |                     |
|  | Pin2      |       | Name of subcircuit label connected to pin 2 |                     |
|  | Pin3      |       | Name of subcircuit label connected to pin 3 |                     |
|  | Cmd       |       | Subcircuit start-up command string          |                     |
|  | IC        |       | Subcircuit Initial conditions string        |                     |
| <p><b>Subcircuit.</b> See <i>Working with Subcircuits</i> chapter for details.</p> |           |       |   |                     |

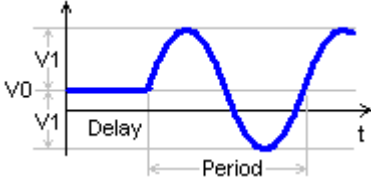
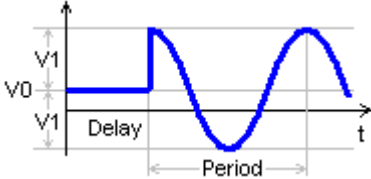
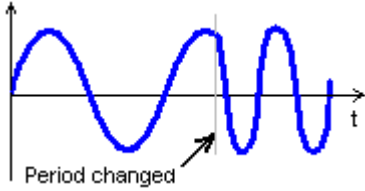
|   |                |  |   |
|---|----------------|--|---|
| <b>V</b>  | Voltage source | <p style="writing-mode: vertical-rl; transform: rotate(180deg);">Models</p> <p>V<br/>Pulse<br/>Step<br/>Sin<br/>PWL<br/>Function<br/>File<br/>SubCir</p> | <p style="writing-mode: vertical-rl; transform: rotate(180deg);">Traces</p>  <p><math>P = V \cdot I</math></p> |
|  |                |  |   |

|          |  |       |             |              |
|----------|--|-------|-------------|--------------|
| <b>V</b> | Voltage source                                 |       |             | V/V_VS_V.n15 |
| Model    | Parameter                                      | Units | Description |              |
| V        | V  | V     | Voltage.    |              |
|          | <b>Constant voltage source.</b> Voltage = "V". |       |             |              |

| V  |           | Voltage source |                                  | V/V_VS_Pulse.n15 |
|--|-----------|----------------|----------------------------------|------------------|
| Model  | Parameter | Units          | Description                      |                  |
| Pulse  | V1        | V              | Pulse On voltage.                |                  |
|  | V0        | V              | Pulse Off voltage.               |                  |
|  | Period    | s              | Period.                          |                  |
|  | Width     | s              | Pulse width.                     |                  |
|  | Slope     |                | Slope type: Linear/Cos/Exp       |                  |
|  | Rise      | s              | Pulse rise length.               |                  |
|  | Fall      | s              | Pulse fall length.               |                  |
|  | Delay     | s              | Delay before first pulse starts. |                  |
| <p><b>Pulse voltage source.</b> Pulses start after “Delay” time. “Rise” time is included into “Width”, “Fall” time is not included into “Width”. Almost every parameter can be set to zero and infinity (“inf”), otherwise the error message will be displayed.</p>  <p>Slope type applies both to pulse rise and fall. The following slope types are available:</p>  <p style="text-align: center;"> <span>Linear</span>                      <b>Cos (cosine)</b>                      <b>Exp (exponential)</b> </p> |           |                |                                  |                  |

| V   |           | Voltage source |                            | V/V_VS_Step.n15 |
|---|-----------|----------------|----------------------------|-----------------|
| Model   | Parameter | Units          | Description                |                 |
| <b>Step</b>   | V1        | V              | Step On voltage.           |                 |
|   | V0        | V              | Step Off voltage.          |                 |
|   | Slope     | s              | Slope type: Linear/Cos/Exp |                 |
|   | Rise      | s              | Step rise length.          |                 |
|   | Delay     | s              | Delay before step starts.  |                 |
| <p><b>Step voltage source.</b> Step starts after “Delay” time.</p>  <p>The following slope types are available:</p>  <p style="text-align: center;"> <span>Linear</span>                      <span>Cos (cosine)</span>                      <span>Exp (exponential)</span> </p> |           |                |                            |                 |



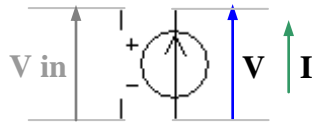
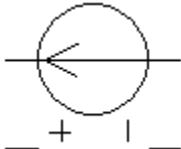
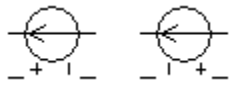
| V   |           | Voltage source |                                  | V/V_VS_Sin.n15 |
|---|-----------|----------------|----------------------------------|----------------|
| Model   | Parameter | Units          | Description                      |                |
| Sin   | V1        | V              | Voltage amplitude.               |                |
|   | V0        | V              | Voltage baseline.                |                |
|   | Period    | s              | Period.                          |                |
|   | Phase     | deg            | Phase.                           |                |
|   | Delay     | s              | Delay before sine signal starts. |                |
| <p><b>Sine voltage source.</b> Sine signal starts after “Delay” time. “Phase” is sine phase in degrees at the moment when signal starts:</p> <div style="display: flex; justify-content: space-around; align-items: flex-end;"> <div style="text-align: center;">  <p><b>Phase = 0</b></p> </div> <div style="text-align: center;">  <p><b>Phase = 90</b></p> </div> </div> <p>If transient is paused, sine period changed, then transient is continued, the phase of the signal remains continuous, providing smooth sine signal of variable frequency:</p> <div style="text-align: center;">  <p>Period changed</p> </div> |           |                |                                  |                |

|   |                |       |                           |                |
|---|----------------|-------|---------------------------|----------------|
| <b>V</b>  | Voltage source |       |                           | V/V_VS_PWL.n15 |
| Model   | Parameter      | Units | Description               |                |
| <b>PWL</b>  | pwl            |       | Comma-separated string.   |                |
|   | Cycle          |       | Cycling (repeat): No/Yes. |                |
|   | Delay          | s     | Delay.                    |                |
| <p><b>Piece-wise linear voltage source.</b> Signal is defined by “pwl” parameter in the csv (“comma-separated values”) format, as follows:</p> $t_0, V_0, t_1, V_1, \dots, t_n, V_n$ <p>where all <math>t</math> and <math>V</math> can be numerical values or expressions. If <math>t &lt; t_0</math>, signal is <math>V_0</math>. If <math>t_0 &lt; t &lt; t_1</math>, signal value is linearly interpolated between <math>V_0</math> and <math>V_1</math>, etc. If <math>t &gt; t_n</math>, then signal value is <math>V_n</math> if “Cycle” parameter is set to “No”, otherwise signal defined in <math>t_0 \dots t_n</math> interval is repeated continuously. In addition, the whole signal is delayed by “Delay” time.</p> <p>Example:<br/> <math>pwl = 0, 0, 1, 2, 4, 3, 5, 0, 8, 0</math></p> <p>If “Cycle” = Yes, “Delay” = 0, the following voltage will be generated:</p> <p>See <i>Working with PWL source</i> chapter for more details.</p> |                |       |                           |                |

|                 |   |       |             |                     |
|-----------------|---|-------|-------------|---------------------|
| <b>V</b>        | Voltage source  |       |             | V/V_VS_Function.n15 |
| Model           | Parameter   | Units | Description |                     |
| <b>Function</b> | F(t)  | V     | Function    |                     |
|                 | <p><b>Arbitrary function.</b> F(t) defines voltage as a function of the following variables:</p> <ul style="list-style-type: none"> <li><b>t</b> - current time</li> <li><b>V(name)</b> - voltage on the component <b>name</b></li> <li><b>I(name)</b> - current through the component <b>name</b></li> <li><b>P(name)</b> – power on the component <b>name</b></li> </ul> <p>where <b>name</b> is the name of any component in the schematic. If F(t) is blank, voltage is zero.</p> <p>Example:<br/> <math>F(t) = \sin(t) * (1 + \cos(t * .01))</math><br/> <math>F(t) = V(R1) * I(R1)</math></p> <p>Please note that <b>V</b>, <b>I</b>, and <b>P</b> variables are taken at previous calculation step. This may affect stability of the schematic with closed loop.</p> |       |             |                     |

|             |   |       |                           |                 |
|-------------|---|-------|---------------------------|-----------------|
| <b>V</b>    | Voltage source  |       |                           | V/V_VS_File.n15 |
| Model       | Parameter   | Units | Description               |                 |
| <b>File</b> | File  |       | File name.                |                 |
|             | Cycle   |       | Cycling (repeat): No/Yes. |                 |
|             | Delay   | s     | Delay.                    |                 |
|             | <p><b>Voltage source defined in the text file.</b> “File” parameter is a file name, with full path to the file. If the file is located in the same directory as schematic file, the path can be omitted. Signal is defined in following format:</p> <pre> &lt;if first line does not start with a number, it is ignored&gt; t0,V0 t1,V1 ..... tn,Vn                 </pre> <p>where all t and V can be numerical values or expressions. If <math>t &lt; t_0</math>, signal is <math>V_0</math>. If <math>t_0 &lt; t &lt; t_1</math>, signal value is linearly interpolated between <math>V_0</math> and <math>V_1</math>, etc. If <math>t &gt; t_n</math>, then signal value is <math>V_n</math> if “Cycle” parameter is set to “No”, otherwise signal defined in <math>t_0 \dots t_n</math> interval is repeated continuously. In addition, the whole signal is delayed by “Delay” time.</p> <p>Example:</p> <pre> 0,0 1,2 4,3 5,0 8,0                 </pre> <p>If “Cycle” = Yes, “Delay” = 0, the following voltage will be generated:</p> |       |                           |                 |

|  |                |       |   |                   |
|--|----------------|-------|---|-------------------|
| <b>V</b>   | Voltage source |       |   | V/V_VS_SubCir.n15 |
| Model  | Parameter      | Units | Description                                 |                   |
| <b>SubCir</b>  | File           |       | File name of subcircuit schematic.          |                   |
|  | Pin1           |       | Name of subcircuit label connected to pin 1 |                   |
|  | Pin2           |       | Name of subcircuit label connected to pin 2 |                   |
|  | Cmd            |       | Subcircuit start-up command string          |                   |
|  | IC             |       | Subcircuit Initial conditions string        |                   |
| <p><b>Subcircuit.</b> See <i>Working with Subcircuits</i> chapter for details.</p> |                |       |   |                   |

|   |   |  |   |
|---|---|--|---|
| <b>V</b>  | Voltage controlled voltage source   | Models<br>Linear<br>Function<br>PWL<br>VCO<br>Pulse<br>PWM<br>SubCir |  <p><math>P = V \cdot I</math></p> |
|  |   |  |   |
| Views   |  |  |   |

|               |   |       |             |                    |
|---------------|---|-------|-------------|--------------------|
| <b>V</b>      | Voltage controlled voltage source                                   |       |             | VV_VCVS_Linear.n15 |
| Model         | Parameter   | Units | Description |                    |
| <b>Linear</b> | K   | V/V   | Gain        |                    |
|               | <b>Linear voltage controlled voltage source.</b> $V = K * V_{in}$ . |       |             |                    |

| <b>V</b>  | Voltage controlled voltage source |       | V/V_VCVS_Function.nl5                    |
|---|-----------------------------------|-------|--|
| Model   | Parameter                         | Units | Description                              |
| <b>Function</b>   | F(x)                              | V     | Output as function of the input.         |
|   | F(s)                              |       | AC transfer function in <i>s</i> domain. |
|   | IC                                | V     | Initial condition: output voltage.       |
| <p><b>Arbitrary function.</b></p> <p><b>Transient analysis.</b> F(x) defines output voltage as a function of the following variables:</p> <p><b>x</b> – input voltage Vin<br/> <b>t</b> - current time<br/> <b>V(name)</b> - voltage on the component <i>name</i><br/> <b>I(name)</b> - current through the component <i>name</i><br/> <b>P(name)</b> – power on the component <i>name</i></p> <p>where <i>name</i> is the name of any component in the schematic. If F(x) is blank, output is zero. F(s) is ignored.</p> <p>Example:<br/> <math>F(x) = x^3</math><br/> <math>F(x) = x * \sin(t)</math><br/> <math>F(x) = P(r1)+P(r2)</math></p> <p>Please note that input voltage <b>x</b>, and variables <b>V</b>, <b>I</b>, and <b>P</b> are taken at previous calculation step. This may affect stability of the schematic with closed loop.</p> <p><b>AC analysis.</b> F(s) defines transfer function in <i>s</i> domain. Only operators and functions that support complex numbers can be used in this function. The following variables can be used in the function:</p> <p><b>f</b> – current AC frequency, Hz<br/> <b>w</b> – angular AC frequency, <math>w = 2\pi f</math> .<br/> <b>s</b> or <b>p</b> – Laplace parameter, <math>s = p = j*2\pi f</math>.</p> <p>Example:<br/> <math>F(s) = 1/(1+s)</math><br/> <math>F(s) = \exp(-1mk*s)</math></p> <p>F(s) is calculated at each frequency. If F(s) is blank, it is assumed to be 1. Also, if F(x) is not blank, it is linearized at DC operating point, and F(s) is multiplied by linearized gain.</p> <p>When calculating DC operating point for transient or AC analysis, output is set to specified output voltage “IC”.</p> |                                   |       |  |

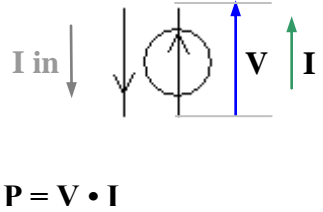
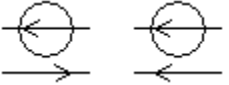
|            |  |       |                                |                  |
|------------|--|-------|--------------------------------|------------------|
| <b>V</b>   | Voltage controlled voltage source  |       |                                | V/V_VCVS_PWL.n15 |
| Model      | Parameter  | Units | Description                    |                  |
| <b>PWL</b> | pwl  |       | Comma-separated string, K(Vin) |                  |
|            | <p><b>Piece-wise linear voltage controlled voltage source.</b> Source gain K is defined by “pwl” string as a function of input voltage Vin. See <i>Working with PWL model</i> chapter for details.</p> |       |                                |                  |

|  |                                   |       |   |                  |
|--|-----------------------------------|-------|---|------------------|
| <b>V</b>   | Voltage controlled voltage source |       |   | V/V_VCVS_VCO.n15 |
| Model  | Parameter                         | Units | Description   |                  |
| <b>VCO</b>   | V1                                | V     | Voltage amplitude (Sin), or Pulse On voltage (Pulse). |                  |
|  | V0                                | V     | Voltage baseline (Sin), or Pulse Off voltage (Pulse). |                  |
|  | dFdV                              | Hz/V  | Gain.   |                  |
|  | Phase                             | deg   | Phase.  |                  |
|  | Type                              |       | Signal type: Sin/Square/Triangle/Sawtooth.            |                  |
| <p><b>Voltage controlled oscillator.</b> Output voltage is a signal with frequency equal to:<br/> <math>f(\text{Hz}) = \text{dFdV} * \text{Vin}</math>.</p> <p>For Sine signal, “V0” is baseline, and “V1” is amplitude. For Square, Triangle, and Sawtooth signals, “V0” is Off level, “V1” is On level. “Phase” is additional phase of the signal, in degrees.</p> |                                   |       |   |                  |

|   |                                   |       |                    |                    |
|---|-----------------------------------|-------|--------------------|--------------------|
| <b>V</b>  | Voltage controlled voltage source |       |                    | V/V_VCVS_Pulse.n15 |
| Model   | Parameter                         | Units | Description        |                    |
| <b>Pulse</b>  | Width                             | s     | Pulse width.       |                    |
|   | Threshold                         | V     | Voltage threshold. |                    |
|   | V1                                | V     | Pulse On voltage.  |                    |
|   | V0                                | V     | Pulse Off voltage. |                    |
| <p><b>One-shot pulse generator.</b> When increasing input voltage Vin crosses “Threshold” value, voltage pulse of “Width” duration is generated. “V0” is pulse Off level, “V1” is pulse On level. If increasing Vin crosses “Threshold” value while pulse is generated, the pulse is restarted.</p> |                                   |       |                    |                    |

| V  |           | Voltage controlled voltage source |   | I/I_VCVS_PWM.n15 |
|--|-----------|-----------------------------------|---|------------------|
| Model  | Parameter | Units                             | Description                               |                  |
| PWM  | V1        | V                                 | Pulse On voltage.                         |                  |
|  | V0        | V                                 | Pulse Off voltage.                        |                  |
|  | F         | Hz                                | Frequency.                                |                  |
|  | Phase     | deg                               | Phase.                                    |                  |
|  | Vmax      | V                                 | Input voltage corresponding to 100% duty. |                  |
| <p><b>Voltage controlled Pulse-Width Modulator.</b> Output voltage is a pulse signal of frequency “F” shifted by “Phase”. Input voltage <math>V_{in}</math> is sampled at the beginning of each cycle of the signal, and width of the output pulse during this cycle is calculated according to the equation:</p> $\text{width} = 1/F * (V_{in} / V_{max})$ <p>or</p> $\text{duty} = 100\% * (V_{in} / V_{max});$ <p>If the width is equal or less than zero, a short “On” pulse with the width equal to the minimum calculation step at that moment will be generated. If the width is equal or greater than period of frequency “F”, a short “Off” pulse at the end of the period will be generated. Due to that, the frequency of the output signal is always “F”. Such a signal can be, for instance, divided by D-trigger to create a signal with duty cycle less than 50%.</p> |           |                                   |   |                  |

| V  |           | Voltage controlled voltage source |   | V/V_VCVS_SubCir.n15 |
|--|-----------|-----------------------------------|---|---------------------|
| Model  | Parameter | Units                             | Description                                 |                     |
| SubCir   | File      |                                   | File name of subcircuit schematic.          |                     |
|  | Pin1      |                                   | Name of subcircuit label connected to pin 1 |                     |
|  | Pin2      |                                   | Name of subcircuit label connected to pin 2 |                     |
|  | Pin3      |                                   | Name of subcircuit label connected to pin 3 |                     |
|  | Pin4      |                                   | Name of subcircuit label connected to pin 4 |                     |
|  | Cmd       |                                   | Subcircuit start-up command string          |                     |
|  | IC        |                                   | Subcircuit Initial conditions string        |                     |
| <p><b>Subcircuit.</b> See <i>Working with Subcircuits</i> chapter for details.</p> |           |                                   |   |                     |

|          |   |  |        |   |
|----------|---|--|--------|---|
| <b>V</b> | Current controlled voltage source   | Models<br>Linear<br>Function<br>PWL<br>CCO<br>Pulse<br>PWM<br>SubCir | Traces |  <p><math>P = V \cdot I</math></p> |
| Views    |  |  |        |   |

|          |   |       |             |                     |
|----------|---|-------|-------------|---------------------|
| <b>V</b> | Current controlled voltage source   |       |             | V/V_CCVS_Linear.nl5 |
| Model    | Parameter   | Units | Description |                     |
| Linear   | K   | V/A   | Gain        |                     |
|          | <b>Linear current controlled voltage source. <math>V = K * I_{in}</math>.</b> |       |             |                     |



|   |                                   |       |  |
|---|-----------------------------------|-------|--|
| <b>V</b>  | Current controlled voltage source |       | V/V_CCVS_Function.nl5                    |
| Model   | Parameter                         | Units | Description                              |
| <b>Function</b>   | F(x)                              | V     | Output as function of the input.         |
|   | F(s)                              |       | AC transfer function in <b>s</b> domain. |
|   | IC                                | V     | Initial condition: output voltage.       |
| <p><b>Arbitrary function.</b></p> <p><b>Transient analysis.</b> F(x) defines output voltage as a function of the following variables:</p> <p><b>x</b> – input current lin<br/> <b>t</b> - current time<br/> <b>V(name)</b> - voltage on the component <b>name</b><br/> <b>I(name)</b> - current through the component <b>name</b><br/> <b>P(name)</b> – power on the component <b>name</b></p> <p>where <b>name</b> is the name of any component in the schematic. If F(x) is blank, output is zero. F(s) is ignored.</p> <p>Example:<br/> <math>F(x) = x^3</math><br/> <math>F(x) = x * \sin(t)</math><br/> <math>F(x) = P(r1)+P(r2)</math></p> <p>Please note that input current <b>x</b>, and variables <b>V</b>, <b>I</b>, and <b>P</b> are taken at previous calculation step. This may affect stability of the schematic with closed loop.</p> <p><b>AC analysis.</b> F(s) defines transfer function in <b>s</b> domain. Only operators and functions that support complex numbers can be used in this function. The following variables can be used in the function:</p> <p><b>f</b> – current AC frequency, Hz<br/> <b>w</b> – angular AC frequency, <math>w = 2\pi f</math> .<br/> <b>s</b> or <b>p</b> – Laplace parameter, <math>s = p = j*2\pi f</math>.</p> <p>Example:<br/> <math>F(s) = 1/(1+s)</math><br/> <math>F(s) = \exp(-1mk*s)</math></p> <p>F(s) is calculated at each frequency. If F(s) is blank, it is assumed to be 1. Also, if F(x) is not blank, it is linearized at DC operating point, and F(s) is multiplied by linearized gain.</p> <p>When calculating DC operating point for transient or AC analysis, output is set to specified output voltage “IC”.</p> |                                   |       |  |


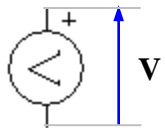
|          |  |       |                                |                  |
|----------|--|-------|--------------------------------|------------------|
| <b>V</b> | Current controlled voltage source  |       |                                | V/V_CCVS_PWL.n15 |
| Model    | Parameter  | Units | Description                    |                  |
| PWL      | pwl  |       | Comma-separated string, K(lin) |                  |
|          | <p><b>Piece-wise linear current controlled voltage source.</b> Source gain K is defined by “pwl” string as a function of input current lin. See <i>Working with PWL model</i> chapter for details.</p> |       |                                |                  |

|  |                                   |       |   |                  |
|--|-----------------------------------|-------|---|------------------|
| <b>V</b>   | Current controlled voltage source |       |   | V/V_CCVS_CCO.n15 |
| Model  | Parameter                         | Units | Description   |                  |
| CCO  | V1                                | V     | Voltage amplitude (Sin), or Pulse On voltage (Pulse). |                  |
|  | V0                                | V     | Voltage baseline (Sin), or Pulse Off voltage (Pulse). |                  |
|  | dFdl                              | Hz/A  | Gain.   |                  |
|  | Phase                             | deg   | Phase.  |                  |
|  | Type                              |       | Signal type: Sin/Square/Triangle/Sawtooth.            |                  |
| <p><b>Current controlled oscillator.</b> Output voltage is a signal with frequency equal to:<br/> <math>f(\text{Hz}) = \text{dFdl} * \text{lin}</math>.</p> <p>For Sine signal, “V0” is baseline, and “V1” is amplitude. For Square, Triangle, and Sawtooth signals, “V0” is Off level, “V1” is On level. “Phase” is additional phase of the signal, in degrees.</p> |                                   |       |   |                  |

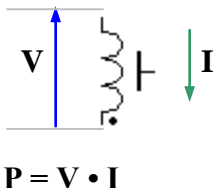
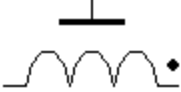
|   |                                   |       |                    |                    |
|---|-----------------------------------|-------|--------------------|--------------------|
| <b>V</b>  | Current controlled voltage source |       |                    | V/V_CCVS_Pulse.n15 |
| Model   | Parameter                         | Units | Description        |                    |
| Pulse   | Width                             | s     | Pulse width.       |                    |
|   | Threshold                         | A     | Current threshold. |                    |
|   | V1                                | V     | Pulse On voltage.  |                    |
|   | V0                                | V     | Pulse Off voltage. |                    |
| <p><b>One-shot pulse generator.</b> When increasing input current lin crosses “Threshold” value, voltage pulse of “Width” duration is generated. “V0” is pulse Off level, “V1” is pulse On level. If increasing lin crosses “Threshold” value while pulse is generated, the pulse is restarted.</p> |                                   |       |                    |                    |

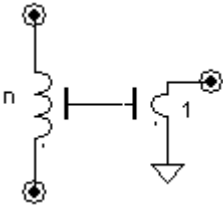
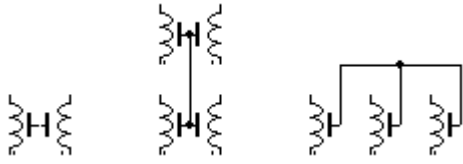
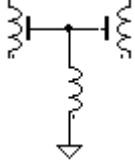
| V   |                  | Current controlled voltage source |   | I/I_CCVS_PWM.n15 |
|---|------------------|-----------------------------------|---|------------------|
| Model   | Parameter        | Units                             | Description                               |                  |
| PWM   | V1               | V                                 | Pulse On voltage.                         |                  |
|   | V0               | V                                 | Pulse Off voltage.                        |                  |
|   | F                | Hz                                | Frequency.                                |                  |
|   | Phase            | deg                               | Phase.                                    |                  |
|   | I <sub>max</sub> | A                                 | Input current corresponding to 100% duty. |                  |
| <p><b>Current controlled Pulse-Width Modulator.</b> Output voltage is a pulse signal of frequency “F” shifted by “Phase”. Input current I<sub>in</sub> is sampled at the beginning of each cycle of the signal, and width of the output pulse during this cycle is calculated according to the equation:</p> $\text{width} = 1/F * (I_{in} / I_{max})$ <p>or</p> $\text{duty} = 100\% * (I_{in} / I_{max});$ <p>If the width is equal or less than zero, a short “On” pulse with the width equal to the minimum calculation step at that moment will be generated. If the width is equal or greater than period of frequency “F”, a short “Off” pulse at the end of the period will be generated. Due to that, the frequency of the output signal is always “F”. Such a signal can be, for instance, divided by D-trigger to create a signal with duty cycle less than 50%.</p> |                  |                                   |   |                  |

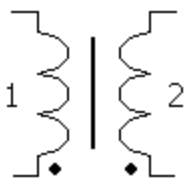
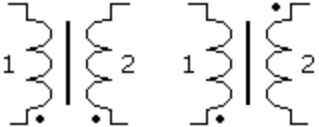
| V  |           | Current controlled voltage source |   | V/V_CCVS_SubCir.n15 |
|--|-----------|-----------------------------------|---|---------------------|
| Model  | Parameter | Units                             | Description                                 |                     |
| SubCir   | File      |                                   | File name of subcircuit schematic.          |                     |
|  | Pin1      |                                   | Name of subcircuit label connected to pin 1 |                     |
|  | Pin2      |                                   | Name of subcircuit label connected to pin 2 |                     |
|  | Pin3      |                                   | Name of subcircuit label connected to pin 3 |                     |
|  | Pin4      |                                   | Name of subcircuit label connected to pin 4 |                     |
|  | Cmd       |                                   | Subcircuit start-up command string          |                     |
|  | IC        |                                   | Subcircuit Initial conditions string        |                     |
| <p><b>Subcircuit.</b> See <i>Working with Subcircuits</i> chapter for details.</p> |           |                                   |   |                     |

|          |                  |        |                  |        |   |
|----------|------------------|--------|------------------|--------|---|
| <b>V</b> | <b>Voltmeter</b> |        | <b>Voltmeter</b> |        |    |
|          |                  | Models |                  | Traces |  |

|                  |   |                             |
|------------------|---|-----------------------------|
| <b>V</b>         | Voltmeter                               | V/V_Voltmeter_Voltmeter.n15 |
| Model            | No parameters                           |                             |
| <b>Voltmeter</b> | <b>Voltmeter.</b> I = 0 (open circuit). |                             |

|   |                |        |                |        |   |
|---|----------------|--------|----------------|--------|---|
| <b>W</b>  | <b>Winding</b> | Models | <b>Winding</b> | Traces |  |
|  |                |        |                |        |   |

| <b>W</b>  | Winding   | W/W_Winding_Winding.nl5 |                  |  |
|---|-----------|-------------------------|------------------|--|
| Model   | Parameter | Units                   | Description      |  |
| <b>Winding</b>  | n         | turns                   | Number of turns. |  |
| <p><b>Winding.</b> The Winding is actually an ideal transformer, with 1 turn second winding, one end of each is grounded, and another end is shown as a “core” pin of the winding:</p> <div style="text-align: center;">  </div> <p style="text-align: center;"><b>Equivalent schematic</b></p> <p>To make an ideal transformer, connect cores of two or more windings by wire. Core magnetizing can be modeled by setting linear or non-linear inductor from core to ground:</p> <div style="display: flex; justify-content: space-around; align-items: center;"> <div style="text-align: center;">  <p><b>Ideal transformers</b></p> </div> <div style="text-align: center;">  <p><b>Transformer with magnetizing inductor</b></p> </div> </div> |           |                         |                  |  |

|   |   |        |                               |        |
|---|---|--------|-------------------------------|--------|
| <b>W</b>  | <b>Transformer</b>  | Models | <b>Transformer<br/>SubCir</b> | Traces |
|  |   |        |                               |        |
| Views   |  |        |                               |        |


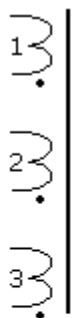
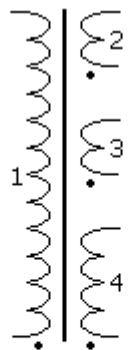
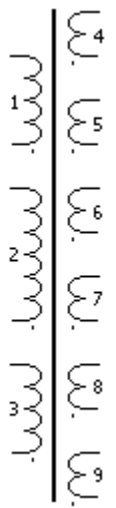
|   |             |                                 |  |  |
|---|-------------|---------------------------------|--|--|
| <b>W</b>  | Transformer | W/W_Transformer_Transformer.n15 |  |  |
| Model   | Parameter   | Units                           | Description                            |  |
| <b>Transformer</b>  | n1          | turns                           | Number of turns in the first winding.  |  |
|   | n2          | turns                           | Number of turns in the second winding. |  |
| <b>Ideal transformer with 2 windings.</b> Coupling coefficient = 1. |             |                                 |  |  |

|   |             |       |   |  |
|---|-------------|-------|---|--|
| <b>W</b>  | Transformer |       |   |  |
| Model   | Parameter   | Units | Description                                 |  |
| <b>SubCir</b>   | File        |       | File name of subcircuit schematic.          |  |
|   | Pin1        |       | Name of subcircuit label connected to pin 1 |  |
|   | Pin2        |       | Name of subcircuit label connected to pin 2 |  |
|   | Pin3        |       | Name of subcircuit label connected to pin 3 |  |
|   | Pin4        |       | Name of subcircuit label connected to pin 4 |  |
|   | Cmd         |       | Subcircuit start-up command string          |  |
|   | IC          |       | Subcircuit Initial conditions string        |  |
| <b>Subcircuit.</b> See <i>Working with Subcircuits</i> chapter for details. |             |       |   |  |

|          |                                 |        |                               |        |
|----------|---------------------------------|--------|-------------------------------|--------|
| <b>W</b> | <b>Differential transformer</b> | Models | <b>Transformer<br/>SubCir</b> | Traces |
|          |                                 |        |                               |        |
| Views    |                                 |        |                               |        |

|   |                          |                                    |  |  |
|---|--------------------------|------------------------------------|--|--|
| <b>W</b>  | Differential transformer | W/W_DifTransformer_Transformer.n15 |  |  |
| Model   | Parameter                | Units                              | Description  |  |
| <b>Transformer</b>  | n1                       | turns                              | Number of turns in the first winding.                |  |
|   | n2                       | turns                              | Number of turns in the second and the third winding. |  |
| <p><b>Ideal differential transformer with 3 windings.</b> Coupling coefficient = 1. Second and third windings have the same number of turns “n2”, and connected to form a differential transformer.</p> |                          |                                    |  |  |

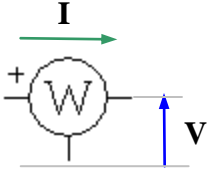
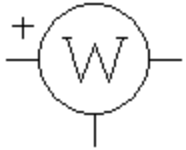
|  |                          |       |   |  |
|--|--------------------------|-------|---|--|
| <b>W</b>   | Differential transformer |       |   |  |
| Model  | Parameter                | Units | Description                                 |  |
| <b>SubCir</b>  | File                     |       | File name of subcircuit schematic.          |  |
|  | Pin1                     |       | Name of subcircuit label connected to pin 1 |  |
|  | Pin2                     |       | Name of subcircuit label connected to pin 2 |  |
|  | Pin3                     |       | Name of subcircuit label connected to pin 3 |  |
|  | Pin4                     |       | Name of subcircuit label connected to pin 4 |  |
|  | Pin5                     |       | Name of subcircuit label connected to pin 5 |  |
|  | Cmd                      |       | Subcircuit start-up command string          |  |
|  | IC                       |       | Subcircuit Initial conditions string        |  |
| <p><b>Subcircuit.</b> See <i>Working with Subcircuits</i> chapter for details.</p> |                          |       |   |  |

|   |   |        |                             |        |  |
|---|---|--------|-----------------------------|--------|--|
| W   | <h3>Custom transformer</h3>   | Models | <h3>Transformer SubCir</h3> | Traces |  |
|  |   |        |                             |        |  |
| Views   | <p>This is a customized component. A component can be edited in the <b>Edit Component</b> dialog box. See <i>Editing customized component</i> chapter for instructions on editing a component.</p> <p>This component may have:</p> <ul style="list-style-type: none"> <li>- height from 2 to 32,</li> <li>- up to 9 windings (total) on both sides,</li> <li>- arbitrary length of a winding.</li> </ul> <p>Examples of Custom transformer component:</p> <div style="display: flex; justify-content: space-around; align-items: center;"> <div style="text-align: center;">  </div> <div style="text-align: center;">  </div> <div style="text-align: center;">  </div> </div> |        |                             |        |  |

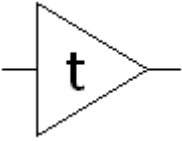
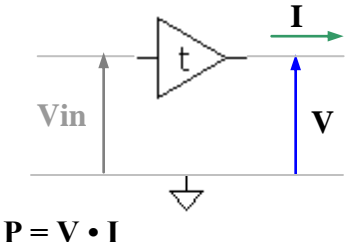


| W           | Custom transformer  |       |   | W/W_CustomTransformer_Transformer.n15 |
|-------------|---|-------|---|---------------------------------------|
| Model       | Parameter   | Units | Description                                     |                                       |
| Transformer | n1  | turns | Number of turns in the first winding.           |                                       |
|             | ...   | ...   | ...   |                                       |
|             | nN  | turns | Number of turns in the N <sup>th</sup> winding. |                                       |
|             | <b>Ideal transformer with N windings.</b> Coupling coefficient = 1. |       |   |                                       |

| W      | Custom transformer  |       |   |  |
|--------|---|-------|---|--|
| Model  | Parameter   | Units | Description                                 |  |
| SubCir | File  |       | File name of subcircuit schematic.          |  |
|        | Pin1  |       | Name of subcircuit label connected to pin 1 |  |
|        | ...   |       | ...   |  |
|        | PinN  |       | Name of subcircuit label connected to pin N |  |
|        | Cmd   |       | Subcircuit start-up command string          |  |
|        | IC  |       | Subcircuit Initial conditions string        |  |
|        | <b>Subcircuit.</b> See <i>Working with Subcircuits</i> chapter for details. |       |   |  |

|   |           |        |           |        |  |
|---|-----------|--------|-----------|--------|--|
| <b>W</b>  | Wattmeter | Models | Wattmeter | Traces |  <p><b><math>P = V \cdot I</math></b></p> |
|  |           |        |           |        |  |

|                  |  |                            |
|------------------|--|----------------------------|
| <b>W</b>         | Wattmeter  | WW_Wattmeter_Wattmeter.nl5 |
| Model            | No parameters  |                            |
| <b>Wattmeter</b> | <p><b>Wattmeter.</b> Short circuit between current ports, open circuit between voltage ports. Can be used to measure power in grounded or non-grounded load.</p> |                            |

|   |              |        |                         |  |   |
|---|--------------|--------|-------------------------|--|---|
| <b>X</b>  | <b>Delay</b> |        | <b>Delay<br/>SubCir</b> |  | <b>Traces</b>   |
|  |              | Models |                         |  |  |

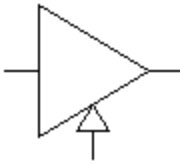
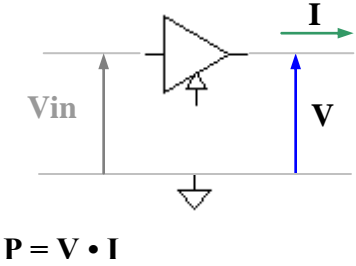
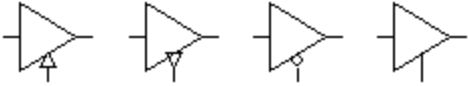
|   |              |       |                                    |  |                     |
|---|--------------|-------|------------------------------------|--|---------------------|
| <b>X</b>  | <b>Delay</b> |       |                                    |  | X/X_Delay_Delay.n15 |
| Model   | Parameter    | Units | Description                        |  |                     |
| <b>Delay</b>  | t0           | s     | Delay.                             |  |                     |
|   | IC           | V     | Initial condition: output voltage. |  |                     |
| <p><b>Delay.</b> Output voltage is equal to input voltage, delayed by delay time “t0”:</p> $V(t) = V_{in}(t - t_0), \text{ where } t \text{ is current time.}$ <p>When calculating DC operating point, output is set to specified output voltage “IC”, or, if “IC” is blank, to input voltage. Then output voltage is not changing until delay time “t0”.</p> <p>The model allocates memory for storing delayed data only when needed, and frees it immediately when possible. At transient start, an approximate amount of needed memory is estimated based on calculation step, and, if it exceeds a limit specified in preferences (Transient page), the warning message is displayed.</p> |              |       |                                    |  |                     |

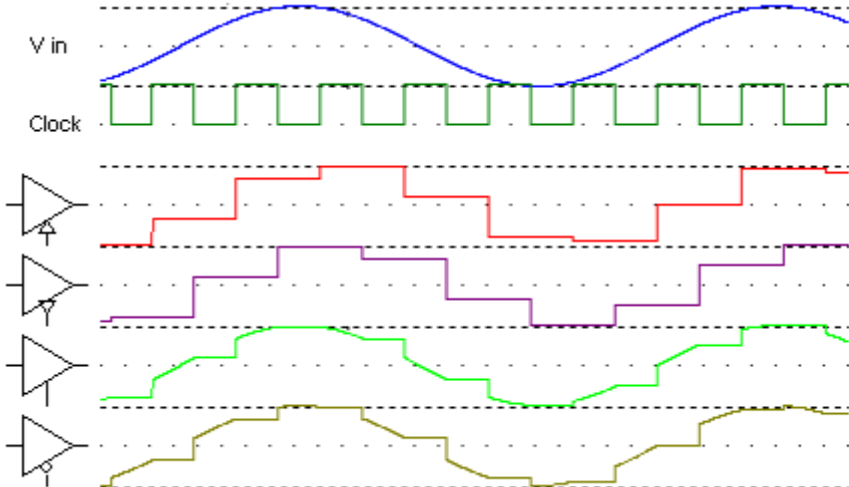
|  |              |       |   |  |  |
|--|--------------|-------|---|--|--|
| <b>X</b>   | <b>Delay</b> |       |   |  |  |
| Model  | Parameter    | Units | Description                                 |  |  |
| <b>SubCir</b>  | File         |       | File name of subcircuit schematic.          |  |  |
|  | Pin1         |       | Name of subcircuit label connected to pin 1 |  |  |
|  | Pin2         |       | Name of subcircuit label connected to pin 2 |  |  |
|  | Cmd          |       | Subcircuit start-up command string          |  |  |
|  | IC           |       | Subcircuit Initial conditions string        |  |  |
| <p><b>Subcircuit.</b> See <i>Working with Subcircuits</i> chapter for details.</p> |              |       |   |  |  |

|          |                   |        |                   |        |  |
|----------|-------------------|--------|-------------------|--------|--|
| <b>X</b> | Transmission line | Models | <b>Line Lossy</b> | Traces |  |
|          |                   |        |                   |        |  |

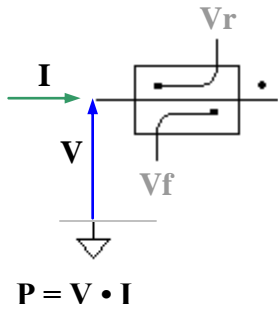
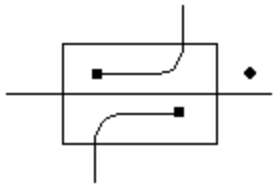
| <b>X</b>   | Transmission line | X/X_Line_Line.n15 |                             |  |
|--|-------------------|-------------------|-----------------------------|--|
| Model  | Parameter         | Units             | Description                 |  |
| <b>Line</b>  | t0                | s                 | Delay.                      |  |
|  | z0                | Ohm               | Characteristic impedance.   |  |
|  | VIC               | V                 | Initial condition: voltage. |  |
|  | IIC               | A                 | Initial condition: current. |  |
| <p><b>Lossless transmission line.</b> The voltage and current in the line are represented as a superposition of forward and reflected waves, with V/I ratio in each wave equal to line characteristic impedance “z0”. V and I values of each wave are calculated based on boundary (input and output) conditions. The line functionality can also be described by the following equations:</p> $V_{in}(t) = z_0 * ( I_{in}(t) - I_{out}(t - t_0) )$ $V_{out}(t) = z_0 * ( I_{out}(t) - I_{in}(t - t_0) )$ <p>where t is current time.</p> <p>Input and output are galvanically isolated: no current is flowing between input and output, and any voltage difference between input and output may exist.</p> <p>When calculating DC operating point initial forward and reflected voltage and current are calculated based on the following conditions:</p> <p>if “VIC” and “IIC” are blank . . . . . : Vin = Vout, Iin = -Iout.<br/>             if “VIC” is specified and “IIC” is blank . . : Vin = Vout = “VIC”.<br/>             if “VIC” is blank and “IIC” is specified . . : Iin = “IIC”, Iout = -“IIC”.<br/>             if “VIC” and “IIC” are specified . . . . . : Vin = Vout = “VIC”, Iin = “IIC”, Iout = -“IIC”.</p> <p>The model allocates memory for storing forward and reflected wave data only when needed, and frees it immediately when possible. At transient start, an approximate amount of needed memory is estimated based on calculation step, and, if it exceeds a limit specified in preferences (Transient page), the warning message is displayed.</p> |                   |                   |                             |  |

| X  | Transmission line |          |  | X/X_Line_Lossy.nl5 |
|--|-------------------|----------|--|--------------------|
| Model  | Parameter         | Units    | Description                                |                    |
| Lossy  | t0                | s        | Delay.                                     |                    |
|  | z0                | Ohm      | Characteristic impedance.                  |                    |
|  | R                 | Ohm/ns   | Series resistance per ns.                  |                    |
|  | fr                | MHz      | Skin losses cutoff (3 dB) frequency.       |                    |
|  | G                 | 1/Ohm/ns | Shunt conductance per ns.                  |                    |
|  | fG                | MHz      | Dielectric losses cutoff (3 dB) frequency. |                    |
|  | VIC               | V        | Initial condition: voltage.                |                    |
|  | IIC               | A        | Initial condition: current.                |                    |
| <p><b>Lossy transmission line.</b> Lossy line modeling is similar to lossless transmission line, with addition of losses due to series resistance, skin effect, shunt conductance, and dielectric losses.</p> <p>Constant series resistance is defined by “r” parameter. Skin losses are modeled by a number of RL chains, providing series impedance increase as a square root of frequency. The number of chains is automatically optimized based on calculation step value; however, the maximum impedance increase due to skin effect is limited to 40 dB (100 times). “fr” parameter defines a frequency where effective series impedance is approximately 3 dB higher than “r”. Skin losses are calculated only if “r” &gt; 0 and “fr is not infinite.</p> <p>Constant shunt conductance is defined by “G” parameter. Dielectric losses are modeled by a shunt capacitance, providing shunt admittance increase proportional to frequency. “fG” parameter defines a frequency where effective shunt admittance is approximately 3 dB higher than “G”. Dielectric losses are calculated only if “G” &gt; 0 and “fG is not infinite.</p> <p>Input and output are galvanically isolated: no current is flowing between input and output, and any voltage difference between input and output may exist.</p> <p>When calculating DC operating point initial forward and reflected voltage and current are calculated based on the following conditions:</p> <p>if “VIC” and “IIC” are blank . . . . . : Vin = Vout, lin = -Iout.<br/>                     if “VIC” is specified and “IIC” is blank . . : Vin = Vout = “VIC”.<br/>                     if “VIC” is blank and “IIC” is specified . . : lin = “IIC”, Iout = -“IIC”.<br/>                     if “VIC” and “IIC” are specified . . . . . : Vin = Vout = “VIC”, lin = “IIC”, Iout = -“IIC”.</p> <p>The model allocates all the required memory immediately at transient start. The amount of memory is proportional to line delay and inverse proportional to calculation step. If the memory required exceeds a limit specified in preferences (Transient page), the warning message is displayed.</p> |                   |          |  |                    |

|              |   |               |                      |  |   |
|--------------|---|---------------|----------------------|--|---|
| <b>X</b>     | <b>Sample/Hold</b>  |               | <b>SH<br/>SubCir</b> |  | <b>Traces</b>   |
|              |  | <b>Models</b> |                      |  | <b>Traces</b>   |
|              |   |               |                      |  |  |
| <b>Views</b> |  |               |                      | Views are used to specify type and polarity of control signal. |   |

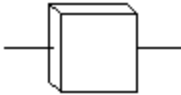
|  |                    |                              |                                    |  |
|--|--------------------|------------------------------|------------------------------------|--|
| <b>X</b>   | <b>Sample/Hold</b> | <b>X/X_SampleHold_SH.n15</b> |                                    |  |
| <b>Model</b>   | <b>Parameter</b>   | <b>Units</b>                 | <b>Description</b>                 |  |
| <b>SH</b>  | <b>IC</b>          | <b>V</b>                     | Initial condition: output voltage. |  |
| <p><b>Sample/hold, track/hold.</b> Depending on view, the model is functioning as a sample/hold, or as a track/hold. In sample/hold mode, input voltage is sampled at rising or falling edge of a logical clock signal. In track/hold mode, output voltage tracks input voltage while clock signal is above the logical threshold, and holds it while clock signal is below the logical threshold (clock signal can be inverted).</p> <p>To see and set logical threshold go to <b>Transient   Settings</b>, or <b>AC   Settings</b> then click <b>Advanced</b> button.</p> <p>A waveforms example for different modes:</p> <div style="text-align: center;">  </div> <p>When calculating DC operating point output is set to specified output voltage "IC".</p> |                    |                              |                                    |  |

| <b>X</b> Sample/Hold  |           |       |   |
|---|-----------|-------|---|
| Model   | Parameter | Units | Description                                 |
| SubCir  | File      |       | File name of subcircuit schematic.          |
|   | Pin1      |       | Name of subcircuit label connected to pin 1 |
|   | Pin2      |       | Name of subcircuit label connected to pin 2 |
|   | Pin3      |       | Name of subcircuit label connected to pin 3 |
|   | Cmd       |       | Subcircuit start-up command string          |
|   | IC        |       | Subcircuit Initial conditions string        |
| <b>Subcircuit.</b> See <i>Working with Subcircuits</i> chapter for details. |           |       |   |

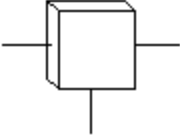
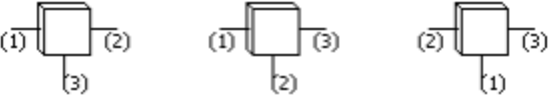
|   |                            |        |                |        |   |
|---|----------------------------|--------|----------------|--------|---|
| <b>X</b>  | <b>Directional coupler</b> | Models | <b>Coupler</b> | Traces |  <p><math>P = V \cdot I</math></p> |
|  |                            |        |                |        |   |

|   |                     |                            |                          |  |
|---|---------------------|----------------------------|--------------------------|--|
| <b>X</b>  | Directional coupler | X/X_DirCoupler_Coupler.n15 |                          |  |
| Model   | Parameter           | Units                      | Description              |  |
| <b>Coupler</b>  | z0                  | Ohm                        | Characteristic impedance |  |
|   | CF                  | dB                         | Coupling factor          |  |
| <p><b>Directional coupler</b> is a short circuit (no insertion loss) with two output ports: forward (<b>Vf</b>) and reflected (<b>Vr</b>). Output ports are voltage sources with zero output impedance and coupling factor <b>CF</b>. The output voltages are calculated as follows:</p> $V_f = K * (V + I * z_0) / 2$ $V_r = K * (V - I * z_0) / 2$ <p>where <math>K = 10^{-CF/20}</math>.</p> <p>All voltages are referenced to ground.</p> |                     |                            |                          |  |

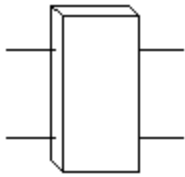
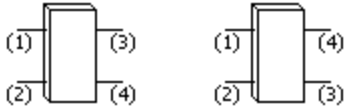


|   |                |        |               |        |  |
|---|----------------|--------|---------------|--------|--|
| <b>X</b>  | <b>Block-2</b> | Models | <b>SubCir</b> | Traces |  |
|  |                |        |               |        |  |

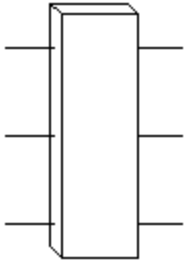
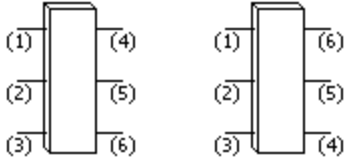
| <b>X</b>  | Block-2   | X/X_Block-2_SubCir.n15 |   |  |
|---|-----------|------------------------|---|--|
| Model   | Parameter | Units                  | Description                                 |  |
| <b>SubCir</b>   | File      |                        | File name of subcircuit schematic.          |  |
|   | Pin1      |                        | Name of subcircuit label connected to pin 1 |  |
|   | Pin2      |                        | Name of subcircuit label connected to pin 2 |  |
|   | Cmd       |                        | Subcircuit start-up command string          |  |
|   | IC        |                        | Subcircuit Initial conditions string        |  |
| <b>Subcircuit.</b> See <i>Working with Subcircuits</i> chapter for details. |           |                        |   |  |

|          |   |        |               |        |  |
|----------|---|--------|---------------|--------|--|
| <b>X</b> | <b>Block-3</b>  |        | <b>SubCir</b> |        |  |
|          |  | Models |               | Traces |  |
| Views    |  |        |               |        |  |

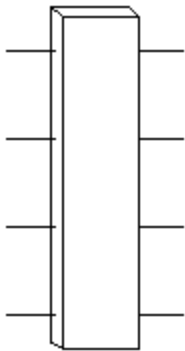
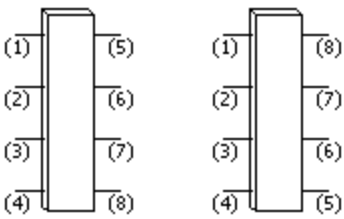
|  |           |                        |   |  |
|--|-----------|------------------------|---|--|
| <b>X</b>   | Block-3   | X/X_Block-3_SubCir.n15 |   |  |
| Model  | Parameter | Units                  | Description                                 |  |
| <b>SubCir</b>  | File      |                        | File name of subcircuit schematic.          |  |
|  | Pin1      |                        | Name of subcircuit label connected to pin 1 |  |
|  | Pin2      |                        | Name of subcircuit label connected to pin 2 |  |
|  | Pin3      |                        | Name of subcircuit label connected to pin 3 |  |
|  | Cmd       |                        | Subcircuit start-up command string          |  |
|  | IC        |                        | Subcircuit Initial conditions string        |  |
| <p><b>Subcircuit.</b> See <i>Working with Subcircuits</i> chapter for details.</p> |           |                        |   |  |

|   |   |        |        |        |
|---|---|--------|--------|--------|
| <b>X</b>  | <b>Block-4</b>  | Models | SubCir | Traces |
|  |   |        |        |        |
| Views   |  |        |        |        |

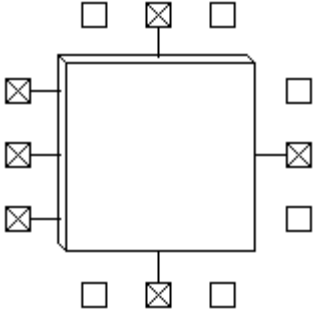
|   |           |                        |   |  |
|---|-----------|------------------------|---|--|
| <b>X</b>  | Block-4   | X/X_Block-4_SubCir.n15 |   |  |
| Model   | Parameter | Units                  | Description                                 |  |
| <b>SubCir</b>   | File      |                        | File name of subcircuit schematic.          |  |
|   | Pin1      |                        | Name of subcircuit label connected to pin 1 |  |
|   | Pin2      |                        | Name of subcircuit label connected to pin 2 |  |
|   | Pin3      |                        | Name of subcircuit label connected to pin 3 |  |
|   | Pin4      |                        | Name of subcircuit label connected to pin 4 |  |
|   | Cmd       |                        | Subcircuit start-up command string          |  |
|   | IC        |                        | Subcircuit Initial conditions string        |  |
| <b>Subcircuit.</b> See <i>Working with Subcircuits</i> chapter for details. |           |                        |   |  |

|   |   |        |        |        |
|---|---|--------|--------|--------|
| <b>X</b>  | <b>Block-6</b>  | Models | SubCir | Traces |
|  |   |        |        |        |
| Views   |  |        |        |        |

| <b>X</b>  | Block-6   | X/X_Block-6_SubCir.n15 |   |  |
|---|-----------|------------------------|---|--|
| Model   | Parameter | Units                  | Description                                 |  |
| <b>SubCir</b>   | File      |                        | File name of subcircuit schematic.          |  |
|   | Pin1      |                        | Name of subcircuit label connected to pin 1 |  |
|   | Pin2      |                        | Name of subcircuit label connected to pin 2 |  |
|   | Pin3      |                        | Name of subcircuit label connected to pin 3 |  |
|   | Pin4      |                        | Name of subcircuit label connected to pin 4 |  |
|   | Pin5      |                        | Name of subcircuit label connected to pin 5 |  |
|   | Pin6      |                        | Name of subcircuit label connected to pin 6 |  |
|   | Cmd       |                        | Subcircuit start-up command string          |  |
|   | IC        |                        | Subcircuit Initial conditions string        |  |
| <b>Subcircuit.</b> See <i>Working with Subcircuits</i> chapter for details. |           |                        |   |  |

|   |   |        |        |        |
|---|---|--------|--------|--------|
| <b>X</b>  | <b>Block-8</b>  | SubCir | Models | Traces |
|  |   |        |        |        |
| Views   |  |        |        |        |

| <b>X</b>  | Block-8   | X/X_Block-8_SubCir.n15 |   |  |
|---|-----------|------------------------|---|--|
| Model   | Parameter | Units                  | Description                                 |  |
| <b>SubCir</b>   | File      |                        | File name of subcircuit schematic.          |  |
|   | Pin1      |                        | Name of subcircuit label connected to pin 1 |  |
|   | Pin2      |                        | Name of subcircuit label connected to pin 2 |  |
|   | Pin3      |                        | Name of subcircuit label connected to pin 3 |  |
|   | Pin4      |                        | Name of subcircuit label connected to pin 4 |  |
|   | Pin5      |                        | Name of subcircuit label connected to pin 5 |  |
|   | Pin6      |                        | Name of subcircuit label connected to pin 6 |  |
|   | Pin7      |                        | Name of subcircuit label connected to pin 7 |  |
|   | Pin8      |                        | Name of subcircuit label connected to pin 8 |  |
|   | Cmd       |                        | Subcircuit start-up command string          |  |
|   | IC        |                        | Subcircuit Initial conditions string        |  |
| <b>Subcircuit.</b> See <i>Working with Subcircuits</i> chapter for details. |           |                        |   |  |

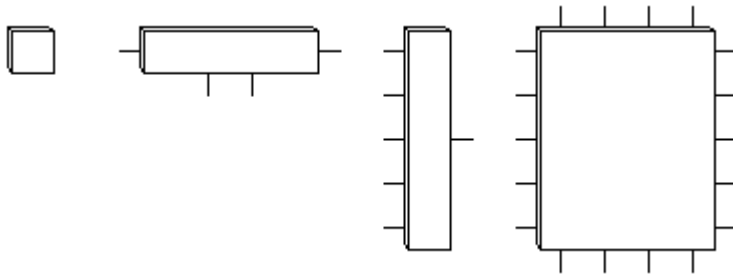
|   |                     |        |               |        |
|---|---------------------|--------|---------------|--------|
| <b>X</b>  | <b>Custom block</b> |        | <b>SubCir</b> |        |
|  |                     | Models |               | Traces |

This is a customized component. A component can be edited in the **Edit Component** dialog box. See *Editing customized component* chapter for instructions on editing a component.

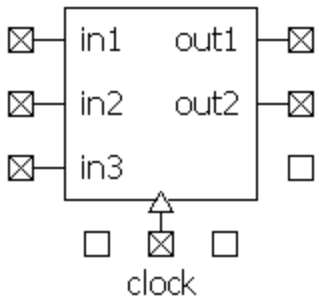
This component may have:

- arbitrary size up to 32(width) X 32(height),
- up to 32 pins on each side

Examples of Custom block component:



|               |   |                            |   |  |
|---------------|---|----------------------------|---|--|
| <b>X</b>      | <b>Custom block</b>   | X/X_CustomBlock_SubCir.nl5 |   |  |
| Model         | Parameter   | Units                      | Description                                 |  |
| <b>SubCir</b> | File  |                            | File name of subcircuit schematic.          |  |
|               | Pin1  |                            | Name of subcircuit label connected to pin 1 |  |
|               | ...   |                            | ...   |  |
|               | PinN  |                            | Name of subcircuit label connected to pin N |  |
|               | Cmd   |                            | Subcircuit start-up command string          |  |
|               | IC  |                            | Subcircuit Initial conditions string        |  |
|               | <b>Subcircuit.</b> See <i>Working with Subcircuits</i> chapter for details. |                            |   |  |

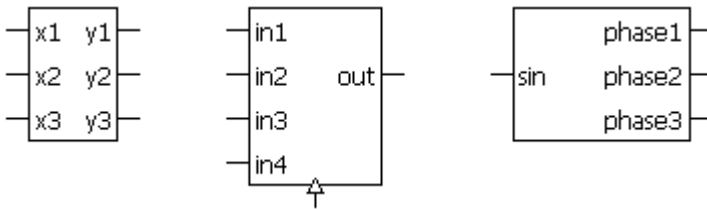
|   |             |        |                  |        |
|---|-------------|--------|------------------|--------|
| <b>X</b>  | <b>Code</b> |        | <b>C<br/>DLL</b> |        |
|  |             | Models |                  | Traces |

This is a customized component. A component can be edited in the **Edit Component** dialog box. See *Editing customized component* chapter for instructions on editing a component.

This component may have:

- arbitrary size up to 32(width) X 32(height),
- up to 32 inputs on the left side,
- up to 32 outputs on the right side,
- one or no clock pins on the bottom side.
- custom input and output names.

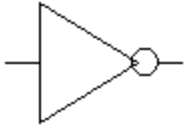
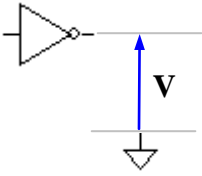
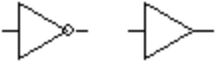
Examples of Code component:



| <b>X</b> | Code  |       |                      |  | X/X_Code_C_Ex1.nl5<br>X/X_Code_C_Ex2.nl5<br>X/X_Code_C_Ex3.nl5 |
|----------|---|-------|----------------------|--|--|
| Model    | Parameter   | Units | Description          |  |  |
| <b>C</b> | Init  |       | Initialization code. |  |  |
|          | Main  |       | Main code.           |  |  |
|          | IC  |       | Initial conditions.  |  |  |
|          | <p><b>C-code block.</b> The model contains code written on simplified C language. The code will be interpreted by NL5 during transient simulation.</p> <p>Initialization code “Init” is executed once at the beginning of simulation at t=0. Initialization code is optional. Leave “Init” parameter blank if initialization code is not used.</p> <p>“Main” is the main code. If <b>clock</b> pin does not exist, the code is executed on every calculation step. If <b>clock</b> pin exists, the code is executed only on rising edge of logical clock signal.</p> <p>“IC” may contain the code assigning initial values to output variables and global variables defined in the initialization code. If not empty, “IC” code will be executed after initialization code.</p> <p>See <i>Creating C-code</i> chapter for details of the model functionality and instructions on creating the code.</p> |       |                      |  |  |

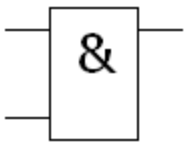
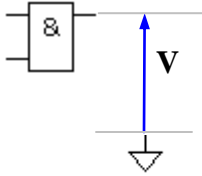
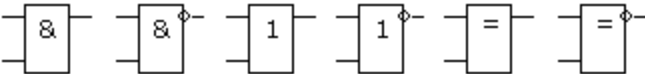
| <b>X</b>   | Code   |       |                               |  | X/X_Code_DLL_Ex1.nl5<br>X/X_Code_DLL_Ex2.nl5<br>X/X_Code_DLL_Ex3.nl5 |
|------------|--|-------|-------------------------------|--|--|
| Model      | Parameter  | Units | Description                   |  |  |
| <b>DLL</b> | DLL  |       | DLL file name                 |  |  |
|            | Init   |       | Initialization function name. |  |  |
|            | Main   |       | Main function name.           |  |  |
|            | <p><b>DLL block.</b> Component’s code is written in C, compiled, and placed in the DLL file. DLL functions will be called by NL5 during transient simulation.</p> <p>“DLL” parameter is a DLL file name, with full path to the file. If the file is located in the same directory as schematic file, the path can be omitted. File extension “dll” can be omitted.</p> <p>“Init” is the name of initialization function. Initialization function is executed once at the beginning of simulation at t=0. Initialization function is optional. Leave “Init” parameter blank if initialization function is not used.</p> <p>“Main” is the name of main function. If <b>clock</b> pin does not exist, the function is executed on every calculation step. If <b>clock</b> pin exists, the function is executed only on rising edge of logical clock signal.</p> <p>See <i>Creating DLL code</i> chapter for details of the model functionality and instructions on creating code and DLL.</p> |       |                               |  |  |



|          |   |        |                    |        |   |   |
|----------|---|--------|--------------------|--------|---|---|
| <b>Y</b> | <b>Logic-1</b>  | Models | <b>Logic Delay</b> | Traces |  |  |
| Views    |  |        |                    |        | Views are used to specify inverted or non-inverted output.                        |   |

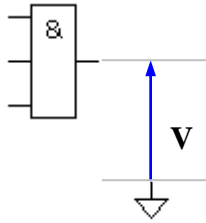
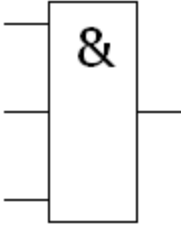
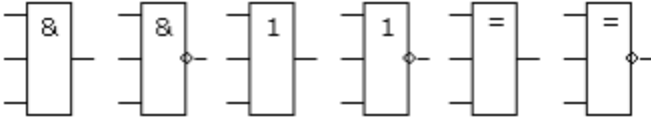
|  |           |                       |                              |  |
|--|-----------|-----------------------|------------------------------|--|
| <b>Y</b>   | Logic-1   | Y/Y_Logic-1_Logic.nl5 |                              |  |
| Model  | Parameter | Units                 | Description                  |  |
| <b>Logic</b>   | IC        |                       | Initial condition: Low/High. |  |
| <p><b>Logical component with one input.</b> Output type (inverted or non-inverted) depends on selected view.</p> <p>Output voltage may have only logical levels (Low/High). Input voltage is considered Low if it is below logical threshold, or High if it is above logical threshold. To see and set logical levels and threshold go to <b>Transient   Settings</b>, or <b>AC   Settings</b>, then click <b>Advanced</b> button.</p> <p>When calculating DC operating point output is set to specified level "IC". When calculating transient, output voltage is always delayed by one calculation step.</p> |           |                       |                              |  |

|  |           |                       |                              |  |
|--|-----------|-----------------------|------------------------------|--|
| <b>Y</b>   | Logic-1   | Y/Y_Logic-1_Delay.nl5 |                              |  |
| Model  | Parameter | Units                 | Description                  |  |
| <b>Delay</b>   | Delay     | s                     | Output delay.                |  |
|  | IC        |                       | Initial condition: Low/High. |  |
| <p><b>Logical component with one input and delay.</b> Output type (inverted or non-inverted) depends on selected view.</p> <p>Output voltage may have only logical levels (Low/High). Input voltage is considered Low if it is below logical threshold, or High if it is above logical threshold. To see and set logical levels and threshold go to <b>Transient   Settings</b>, or <b>AC   Settings</b>, then click <b>Advanced</b> button.</p> <p>The output is delayed by "Delay" time. Input pulses shorter than "Delay" will not pass through and will not affect output.</p> <p>When calculating DC operating point output is set to specified level "IC".</p> |           |                       |                              |  |

|          |   |  |                    |  |   |        |   |        |   |  |
|----------|---|--|--------------------|--|---|--------|---|--------|---|--|
| <b>Y</b> | <b>Logic-2</b>  |  | <b>Logic Delay</b> |  |  | Models |   | Traces |  |  |
| Views    |  |  |                    |  |   |        | Views are used to specify logical function and inverted or non-inverted output. |        |   |  |

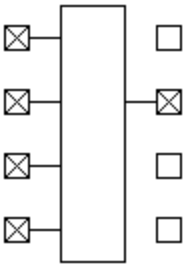
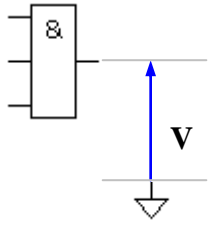
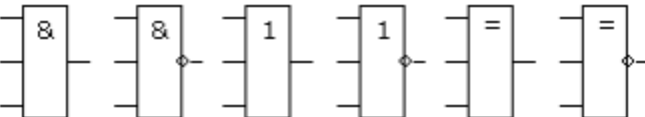
|  |           |                       |                              |  |  |  |  |  |
|--|-----------|-----------------------|------------------------------|--|--|--|--|--|
| <b>Y</b>   | Logic-2   | Y/Y_Logic-2_Logic.n15 |                              |  |  |  |  |  |
| Model  | Parameter | Units                 | Description                  |  |  |  |  |  |
| <b>Logic</b>   | IC        |                       | Initial condition: Low/High. |  |  |  |  |  |
| <p><b>Logical component with two inputs.</b> Logic function (AND, OR, XOR) and output type (inverted or non-inverted) depend on selected view.</p> <p>Output voltage may have only logical levels (Low/High). Input voltage is considered Low if it is below logical threshold, or High if it is above logical threshold. To see and set logical levels and threshold go to <b>Transient   Settings</b>, or <b>AC   Settings</b>, then click <b>Advanced</b> button.</p> <p>When calculating DC operating point output is set to specified level "IC". When calculating transient, output voltage is always delayed by one calculation step.</p> |           |                       |                              |  |  |  |  |  |

|  |           |                       |                              |  |  |  |  |  |
|--|-----------|-----------------------|------------------------------|--|--|--|--|--|
| <b>Y</b>   | Logic-2   | Y/Y_Logic-2_Delay.n15 |                              |  |  |  |  |  |
| Model  | Parameter | Units                 | Description                  |  |  |  |  |  |
| <b>Delay</b>   | Delay     | s                     | Output delay.                |  |  |  |  |  |
|  | IC        |                       | Initial condition: Low/High. |  |  |  |  |  |
| <p><b>Logical component with two inputs and delay.</b> Logic function (AND, OR, XOR) and output type (inverted or non-inverted) depend on selected view.</p> <p>Output voltage may have only logical levels (Low/High). Input voltage is considered Low if it is below logical threshold, or High if it is above logical threshold. To see and set logical levels and threshold go to <b>Transient   Settings</b>, or <b>AC   Settings</b>, then click <b>Advanced</b> button.</p> <p>The output is delayed by "Delay" time. Input pulses shorter than "Delay" will not pass through and will not affect output.</p> <p>When calculating DC operating point output is set to specified level "IC".</p> |           |                       |                              |  |  |  |  |  |

|   |         |   |             |   |
|---|---------|---|-------------|---|
| Y | Logic-3 |   | Logic Delay |  |
|   | Views   |  |             |   |
|   |         |  |             |   |

|       |  |       |                              |                       |
|-------|--|-------|------------------------------|-----------------------|
| Y     | Logic-3  |       |                              | Y/Y_Logic-3_Logic.n15 |
| Model | Parameter  | Units | Description                  |                       |
| Logic | IC   |       | Initial condition: Low/High. |                       |
|       | <p><b>Logical component with three inputs.</b> Logic function (AND, OR, XOR) and output type (inverted or non-inverted) depend on selected view.</p> <p>Output voltage may have only logical levels (Low/High). Input voltage is considered Low if it is below logical threshold, or High if it is above logical threshold. To see and set logical levels and threshold go to <b>Transient   Settings</b>, or <b>AC   Settings</b>, then click <b>Advanced</b> button.</p> <p>When calculating DC operating point output is set to specified level "IC". When calculating transient, output voltage is always delayed by one calculation step.</p> |       |                              |                       |

| Y     | Logic-3  |       | Y/Y_Logic-3_Delay.nl5        |
|-------|--|-------|------------------------------|
| Model | Parameter  | Units | Description                  |
| Delay | Delay  | s     | Output delay.                |
|       | IC   |       | Initial condition: Low/High. |
|       | <p><b>Logical component with three inputs and delay.</b> Logic function (AND, OR, XOR) and output type (inverted or non-inverted) depend on selected view.</p> <p>Output voltage may have only logical levels (Low/High). Input voltage is considered Low if it is below logical threshold, or High if it is above logical threshold. To see and set logical levels and threshold go to <b>Transient   Settings</b>, or <b>AC   Settings</b>, then click <b>Advanced</b> button.</p> <p>The output is delayed by “Delay” time. Input pulses shorter than “Delay” will not pass through and will not affect output.</p> <p>When calculating DC operating point output is set to specified level “IC”.</p> |       |                              |

|   |   |        |                    |  |
|---|---|--------|--------------------|--|
| <b>Y</b>  | <b>Custom logic</b>   |        | <b>Logic Delay</b> |  |
|   |  | Models |                    |     |
| Views   |  |        |                    | <p>Views are used to specify logical function and inverted or non-inverted output.</p> |
| <p>This is a customized component. A component can be edited in the <b>Edit Component</b> dialog box. See <i>Editing customized component</i> chapter for instructions on editing a component.</p> <p>This component may have:</p> <ul style="list-style-type: none"> <li>- arbitrary size up to 32(width) X 32(height),</li> <li>- up to 32 inputs on the left side,</li> <li>- one output on the right side.</li> </ul> |   |        |                    |  |

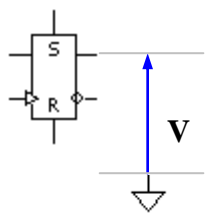
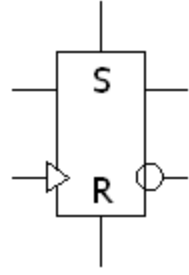
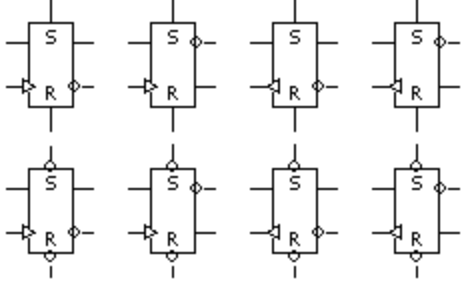
|          |              |                            |       |   |
|----------|--------------|----------------------------|-------|---|
| <b>Y</b> | Custom logic | Y/Y_Custom_Logic_Logic.n15 |       |   |
|          | Model        | Parameter                  | Units | Description   |
|          | <b>Logic</b> | IC                         |       | <p>Initial condition: Low/High.</p> <p><b>Custom logical component.</b> Logic function (AND, OR, XOR) and output type (inverted or non-inverted) depend on selected view.</p> <p>Output voltage may have only logical levels (Low/High). Input voltage is considered Low if it is below logical threshold, or High if it is above logical threshold. To see and set logical levels and threshold go to <b>Transient   Settings</b>, or <b>AC   Settings</b>, then click <b>Advanced</b> button.</p> <p>When calculating DC operating point output is set to specified level "IC". When calculating transient, output voltage is always delayed by one calculation step.</p> |

| Y  | Custom logic |       | Y/Y_Custom_Logic_Delay.n15   |
|--|--------------|-------|------------------------------|
| Model  | Parameter    | Units | Description                  |
| Delay  | Delay        | s     | Output delay.                |
|  | IC           |       | Initial condition: Low/High. |
| <p><b>Custom logical component with delay.</b> Logic function (AND, OR, XOR) and output type (inverted or non-inverted) depend on selected view.</p> <p>Output voltage may have only logical levels (Low/High). Input voltage is considered Low if it is below logical threshold, or High if it is above logical threshold. To see and set logical levels and threshold go to <b>Transient   Settings</b>, or <b>AC   Settings</b>, then click <b>Advanced</b> button.</p> <p>The output is delayed by “Delay” time. Input pulses shorter than “Delay” will not pass through and will not affect output.</p> <p>When calculating DC operating point output is set to specified level “IC”.</p> |              |       |                              |

|          |                  |        |                    |   |  |
|----------|------------------|--------|--------------------|---|--|
| <b>Y</b> | <b>D-trigger</b> |        | <b>Logic Delay</b> |   |  |
|          |                  | Models |                    | Traces  |  |
| Views    |                  |        |                    | Views are used to specify clock polarity and location of output pins. |  |

|          |           |                         |  |  |
|----------|-----------|-------------------------|--|--|
| <b>Y</b> | D-trigger | Y/Y_D_Trigger_Logic.n15 |  |  |
| Model    | Parameter | Units                   | Description  |  |
| Logic    | IC        |                         | <p>Initial condition: Low/High.</p> <p><b>D-trigger.</b> Clock polarity (rising or falling edge) depends on selected view.</p> <p>Output voltage may have only logical levels (Low/High). Input voltage is considered Low if it is below logical threshold, or High if it is above logical threshold. To see and set logical levels and threshold go to <b>Transient   Settings</b>, or <b>AC   Settings</b>, then click <b>Advanced</b> button.</p> <p>When calculating DC operating point output is set to specified level "IC". When calculating transient, output voltage is always delayed by one calculation step.</p> |  |

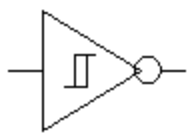


|          |           |                         |  |  |
|----------|-----------|-------------------------|--|--|
| <b>Y</b> | D-trigger | Y/Y_D_Trigger_Delay.n15 |  |  |
| Model    | Parameter | Units                   | Description  |  |
| Delay    | Delay     | s                       | <p>Output delay</p> <p>IC</p> <p>Initial condition: Low/High.</p> <p><b>D-trigger with delay.</b> Clock polarity (rising or falling edge) depends on selected view.</p> <p>Output voltage may have only logical levels (Low/High). Input voltage is considered Low if it is below logical threshold, or High if it is above logical threshold. To see and set logical levels and threshold go to <b>Transient   Settings</b>, or <b>AC   Settings</b>, then click <b>Advanced</b> button.</p> <p>Trigger outputs are delayed by "Delay" time. Output pulses with duration shorter than "Delay" will not show up.</p> <p>When calculating DC operating point output is set to specified level "IC".</p> |  |

|       |   |  |   |        |
|-------|---|--|---|--------|
| Y     | RS-trigger  | Logic Delay  |  |        |
|       |  |  |   | Models |
| Views |  | Views are used to specify clock and S/R inputs polarity and location of output pins. |   |        |

|  |            |                          |                              |  |
|--|------------|--------------------------|------------------------------|--|
| Y  | RS-trigger | Y/Y_RS_Trigger_Logic.nl5 |                              |  |
| Model  | Parameter  | Units                    | Description                  |  |
| Logic  | IC         |                          | Initial condition: Low/High. |  |
| <p><b>RS-trigger.</b> Clock polarity (rising or falling edge) and R/S polarity (inverted or non-inverted) depend on selected view.</p> <p>Output voltage may have only logical levels (Low/High). Input voltage is considered Low if it is below logical threshold, or High if it is above logical threshold. To see and set logical levels and threshold go to <b>Transient   Settings</b>, or <b>AC   Settings</b>, then click <b>Advanced</b> button.</p> <p>When calculating DC operating point output is set to specified level "IC". When calculating transient, output voltage is always delayed by one calculation step.</p> |            |                          |                              |  |

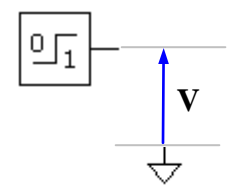


| Y     | RS-trigger   |       |                              | Y/Y_RS_Trigger_Delay.n15 |
|-------|--|-------|------------------------------|--------------------------|
| Model | Parameter  | Units | Description                  |                          |
| Delay | Delay  | s     | Output delay.                |                          |
|       | IC   |       | Initial condition: Low/High. |                          |
|       | <p><b>RS-trigger with delay.</b> Clock polarity (rising or falling edge) and R/S polarity (inverted or non-inverted) depend on selected view.</p> <p>Output voltage may have only logical levels (Low/High). Input voltage is considered Low if it is below logical threshold, or High if it is above logical threshold. To see and set logical levels and threshold go to <b>Transient   Settings</b>, or <b>AC   Settings</b>, then click <b>Advanced</b> button.</p> <p>Trigger outputs are delayed by “Delay” time. Output pulses with duration shorter than “Delay” will not show up.</p> <p>When calculating DC operating point output is set to specified level “IC”.</p> |       |                              |                          |

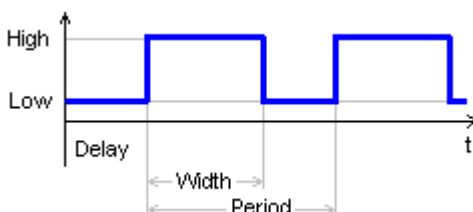
|          |   |  |        |
|----------|---|--|--------|
| <b>Y</b> | Schmitt trigger   |  |        |
|          |    | Models   | Traces |
| Views    |   | Views are used to specify inverted or non-inverted output. |        |

|  |                 |       |                              |                               |
|--|-----------------|-------|------------------------------|-------------------------------|
| <b>Y</b>   | Schmitt trigger |       |                              | Y/Y_Schmitt_Trigger_Logic.nl5 |
| Model  | Parameter       | Units | Description                  |                               |
| Logic  | Hysteresis      | V     | Hysteresis.                  |                               |
|  | IC              |       | Initial condition: Low/High. |                               |
| <p><b>Schmitt trigger.</b> Output type (inverted or non-inverted) depends on selected view.</p> <p>Output voltage may have only logical levels (Low/High). To see and set logical levels and threshold go to <b>Transient   Settings</b>, or <b>AC   Settings</b>, then click <b>Advanced</b> button.</p> <p>Output is set to Low or High level following rules (inverted output):</p> <p style="margin-left: 40px;"> <math>V_{in} &gt; \text{Threshold} + \text{Hysteresis}/2 \dots : V = \text{Low}</math><br/> <math>V_{in} &lt; \text{Threshold} - \text{Hysteresis}/2 \dots : V = \text{High}</math><br/>                 Otherwise ..... : <math>V = \text{previous state}</math> </p> <p>When calculating DC operating point output is set to specified level "IC". When calculating transient, output voltage is always delayed by one calculation step.</p> |                 |       |                              |                               |

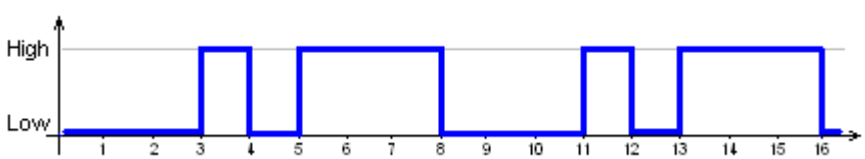
| Y   |            | Schmitt trigger |                              | Y/Y_Schmitt_Trigger_Delay.nl5 |
|---|------------|-----------------|------------------------------|-------------------------------|
| Model   | Parameter  | Units           | Description                  |                               |
| Delay   | Hysteresis | V               | Hysteresis.                  |                               |
|   | Delay      | s               | Output delay                 |                               |
|   | IC         |                 | Initial condition: Low/High. |                               |
| <p><b>Schmitt trigger with delay.</b> Output type (inverted or non-inverted) depends on selected view.</p> <p>Output voltage may have only logical levels (Low/High). To see and set logical levels and threshold go to <b>Transient   Settings</b>, or <b>AC   Settings</b>, then click <b>Advanced</b> button.</p> <p>Output is set to Low or High level following rules (inverted output):</p> <p style="padding-left: 40px;"> <math>V_{in} &gt; \text{Threshold} + \text{Hysteresis}/2 \dots : V = \text{Low}</math><br/> <math>V_{in} &lt; \text{Threshold} - \text{Hysteresis}/2 \dots : V = \text{High}</math><br/>                     Otherwise ..... : V = previous state                 </p> <p>Trigger outputs are delayed by “Delay” time. Output pulses with duration shorter than “Delay” will not show up.</p> <p>When calculating DC operating point output is set to specified level “IC”.</p> |            |                 |                              |                               |

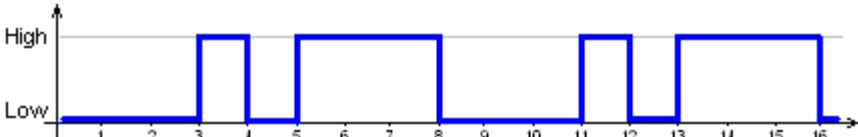
|   |                 |  |   |
|---|-----------------|--|---|
| Y   | Logic generator | Models<br>V<br>Pulse<br>Step<br>List<br>File | Traces<br> |
|  |                 |  |   |

|  |                 |                           |                           |  |
|--|-----------------|---------------------------|---------------------------|--|
| Y  | Logic generator | Y/Y_Logic_Generator_V.nl5 |                           |  |
| Model  | Parameter       | Units                     | Description               |  |
| V  | Out             |                           | Logical output: Low/High. |  |
| <p><b>Logical output.</b> Generates constant Low or High logical output.</p> <p>To see and set logical levels go to <b>Transient   Settings</b>, or <b>AC   Settings</b>, then click <b>Advanced</b> button.</p> |                 |                           |                           |  |

|   |                 |                               |                                  |  |
|---|-----------------|-------------------------------|----------------------------------|--|
| Y   | Logic generator | Y/Y_Logic_Generator_Pulse.nl5 |                                  |  |
| Model   | Parameter       | Units                         | Description                      |  |
| Pulse   | Period          | s                             | Period.                          |  |
|   | Width           | s                             | Pulse width.                     |  |
|   | Delay           | s                             | Delay before first pulse starts. |  |
|   | Active          |                               | Active output state: Low/High.   |  |
| <p><b>Logical pulses.</b> Pulses start at “Delay” time. Output level is “Active” during “Width” time.</p> <p>To see and set logical levels go to <b>Transient   Settings</b>, or <b>AC   Settings</b>, then click <b>Advanced</b> button.</p> <p>The following pulses will be generated if “Active” = On:</p>  |                 |                               |                                  |  |

|             |   |       |                                |                              |
|-------------|---|-------|--------------------------------|------------------------------|
| <b>Y</b>    | Logic generator   |       |                                | Y/Y_Logic_Generator_Step.nl5 |
| Model       | Parameter   | Units | Description                    |                              |
| <b>Step</b> | Delay   | s     | Delay before active state.     |                              |
|             | Active  |       | Active output state: Low/High. |                              |
|             | <p><b>Logical step.</b> Output level is non-“Active” before “Delay” time, turns to “Active” level after “Delay” time.</p> <p>To see and set logical levels go to <b>Transient   Settings</b>, or <b>AC   Settings</b>, then click <b>Advanced</b> button.</p> |       |                                |                              |

|             |  |       |                           |                              |
|-------------|--|-------|---------------------------|------------------------------|
| <b>Y</b>    | Logic generator  |       |                           | Y/Y_Logic_Generator_List.nl5 |
| Model       | Parameter  | Units | Description               |                              |
| <b>List</b> | List   |       | Comma-separated string.   |                              |
|             | Cycle  |       | Cycling (repeat): No/Yes. |                              |
|             | Delay  | s     | Delay.                    |                              |
|             | <p><b>Logic list.</b> Logical output sequence is defined in the “List” parameter in the csv (“comma-separated values”) format, as follows:</p> <p style="text-align: center;"><math>t_0, s_0, t_1, s_1, \dots, t_n, s_n</math></p> <p>where all <math>t</math> and <math>s</math> can be numerical values or expressions. <math>s_0 \dots s_n</math> defines output logical level: positive number corresponds to High, zero or negative number - Low. If <math>t &lt; t_0</math>, output level is <math>s_0</math>. At <math>t_0</math> output level is <math>s_0</math>. At <math>t_1</math> output level is <math>s_1</math>, and so on. At <math>t &gt; t_n</math>, output remains at <math>s_n</math> level if “Cycle” parameter is set to “No”, otherwise the sequence defined in <math>t_0 \dots t_n</math> interval is repeated continuously. In addition, the whole signal is delayed by “Delay” time.</p> <p>To see and set logical levels go to <b>Transient   Settings</b>, or <b>AC   Settings</b>, then click <b>Advanced</b> button.</p> <p>Example:<br/> <code>List = 0,0,3,1,4,0,5,1,8,0</code></p> <p>The following logical output will be generated if “Cycle” = Yes, “Delay” = 0:</p>  <p>See <i>Working with List model</i> chapter for more details.</p> |       |                           |                              |

| Y  | Logic generator |       |                           | Y/Y_Logic_Generator_File.n15 |
|--|-----------------|-------|---------------------------|------------------------------|
| Model  | Parameter       | Units | Description               |                              |
| <b>File</b>  | File            |       | File name.                |                              |
|  | Cycle           |       | Cycling (repeat): No/Yes. |                              |
|  | Delay           | s     | Delay.                    |                              |
| <p><b>Logic file.</b> Logical output sequence is defined in the text file. “File” parameter is a file name, with full path to the file. If the file is located in the same directory as schematic file, the path can be omitted. Logical output sequence is defined in the following format:</p> <pre>&lt;if first line does not start with a number, it is ignored&gt; t0,s0 t1,s1 ..... tn,sn</pre> <p>where all t and s can be numerical values or expressions. s0...sn defines output logical level: positive number corresponds to High, zero or negative number - Low. If <math>t &lt; t_0</math>, output level is s0. At <math>t_0</math> output level is s0. At <math>t_1</math> output level is s1, and so on. At <math>t &gt; t_n</math>, output remains at sn level if “Cycle” parameter is set to “No”, otherwise the sequence defined in <math>t_0...t_n</math> interval is repeated continuously. In addition, the whole signal is delayed by “Delay” time.</p> <p>To see and set logical levels go to <b>Transient   Settings</b>, or <b>AC   Settings</b>, then click <b>Advanced</b> button.</p> <p>Example:</p> <pre>0,0 3,1 4,0 5,1 8,0</pre> <p>The following logical output will be generated if “Cycle” = Yes, “Delay” = 0:</p>  |                 |       |                           |                              |

## 2. Operators

**P** column shows operator precedence: 1- least, 8 - most.

**C** column indicates if operator supports complex numbers.

| Operator | P | C | Description   | Example                       |
|----------|---|---|---|-------------------------------|
| $x=y$    | 1 | √ | Assignment  |                               |
| $x+=y$   | 1 | √ | Assignment by addition,<br>$x=x+y$  |                               |
| $x-=y$   | 1 | √ | Assignment by subtraction,<br>$x=x-y$   |                               |
| $x*=y$   | 1 | √ | Assignment by multiplication,<br>$x=x*y$  |                               |
| $x/=y$   | 1 | √ | Assignment by division,<br>$x=x/y$  |                               |
| $a?x:y$  | 2 | √ | $x$ if $a > 0$ ,<br>$y$ otherwise<br>Condition $a$ is calculated first,<br>then only $x$ or $y$ is calculated,<br>according to condition. | $3>2?1:0=1$<br>$3==2?1:0=0$   |
| $x\&\&y$ | 3 |   | 1 if $x>0$ and $y>0$ ,<br>0 otherwise   | $5\&\&8=1$<br>$5\&\&0=0$      |
| $x  y$   | 3 |   | 1 if $x>0$ or $y>0$ ,<br>0 otherwise  | $5  (-5)=1$<br>$(-5)  (-8)=0$ |
| $x<y$    | 4 |   | 1 if $x<y$ ,<br>0 otherwise   | $5<8=1$<br>$8<5=0$            |
| $x<=y$   | 4 |   | 1 if $x<=y$ ,<br>0 otherwise  | $5<=5=1$<br>$5<=2=0$          |
| $x>y$    | 4 |   | 1 if $x>y$ ,<br>0 otherwise   | $5>8=0$<br>$8>5=1$            |
| $x>=y$   | 4 |   | 1 if $x>=y$ ,<br>0 otherwise  | $5>=5=1$<br>$5>=8=0$          |
| $x==y$   | 4 | √ | 1 if $x$ equal $y$ ,<br>0 otherwise   | $5==8=0$<br>$5+j1==5+j1=1$    |
| $x!=y$   | 4 | √ | 1 if $x$ non-equal $y$ ,<br>0 otherwise   | $5!=8=1$<br>$5+j1!=5+j1=0$    |
| $x+y$    | 5 | √ | $x+y$   | $2+3=5$<br>$1+1j+2+2j=3+3j$   |
| $x-y$    | 5 | √ | $x-y$   | $3-2=1$<br>$3+3j-2-2j=1+1j$   |
| $x*y$    | 6 | √ | $x*y$   | $2*3=6$<br>$(1+2j)*(1-2j)=5$  |

| Operator | P | C | Description                          | Example                                     |
|----------|---|---|--------------------------------------|---|
| $x/y$    | 6 | √ | $x/y$                                | $4/2=2$<br>$(2+2j)/(1-1j)=0+2j$             |
| $x\%y$   | 6 |   | x modulo y, the remainder of x/y     | $5\%2=1$                                    |
| $x^y$    | 7 | √ | $x^y$ : x to the power of y          | $2^3=8$<br>$(-4)^{0.5}=0+2j$<br>$1j^3=0-1j$ |
| $++x$    | 8 | √ | Prefix increment: $x=x+1$ before use |   |
| $x++$    | 8 | √ | Postfix increment: $x=x+1$ after use |   |
| $--x$    | 8 | √ | Prefix decrement: $x=x-1$ before use |   |
| $x--$    | 8 | √ | Postfix decrement: $x=x-1$ after use |   |



### 3. Functions

C column indicates if function supports complex numbers.

| Function          | C | Description  | Example  |
|-------------------|---|--|--|
| sin(x)            |   | sin(x), x in degrees.  | sin(90)=1  |
| cos(x)            |   | cos(x), x in degrees.  | cos(45)= 707.106e-3  |
| tan(x)<br>tg(x)   |   | tan(x), x in degrees.  | tan(45)=1  |
| sqrt(x)           | √ | $\sqrt{x}$ : square root   | sqrt(4)=2<br>sqrt(-4)=0+2j<br>sqrt(2j)=1+1j                      |
| sqr(x)            |   | "Signed" square root:<br>$\sqrt{x}$ if $x \geq 0$ ,<br>$-\sqrt{-x}$ if $x < 0$ | sqr(4)=2<br>sqr(-4)=-2   |
| sq(x)             | √ | $x^2$ : square   | sq(2)=4<br>sq(1+1j)=0+2j   |
| pow(x,y)          | √ | $x^y$ : x to the power of y  | pow(10,2)=100<br>pow(-4,0.5)=0+2j<br>pow(1j,3)=0-1j              |
| pwr(x,y)          |   | "Signed" power:<br>$x^y$ if $x \geq 0$ ,<br>$-(-x)^y$ if $x < 0$               | pwr(10,2)=100<br>pwr(-10,2)=-100                                 |
| exp(x)            | √ | $e^x$ : exponent   | exp(3)=20.0855<br>exp(PI*0.5j)=0+1j                              |
| ln(x)<br>log(x)   | √ | ln(x) : logarithm x to base e  | ln(100)=4.60517<br>ln(-1)=0+3.14159j<br>ln(-1j)=0-1.57079j       |
| lg(x)<br>log10(x) | √ | log <sub>10</sub> (x) : logarithm x to base 10                                 | log10(100)=2<br>lg(-100)=2+1.36437j<br>lg(1j)=0+682.188e-3j      |
| lb(x)<br>log2(x)  | √ | log <sub>2</sub> (x) : logarithm x to base 2 ("binary" logarithm)              | log2(8)=3<br>lb(-8)=3+4.53236j<br>lb(1j)=0+2.26618j              |
| log(x,y)          | √ | log <sub>y</sub> (x) : logarithm x to base y.<br>Complex base y not allowed.   | log(PI,PI)=1<br>log(-10,10)=1+1.36437j<br>log(1j,10)=0+682.1e-3j |
| asin(x)           |   | arcsin(x), -90...+90 degrees   | asin(1)=90   |
| acos(x)           |   | arccos(x), 0...+180 degrees  | acos(.5)=60  |
| atan(x)           |   | arctan(x), -90...+90 degrees   | atan(1)=45   |

| Function                               | C | Description   | Example  |
|--|---|---|--|
| atan2(x, y)                            |   | arctan(x/y), -180...180 degrees                               | atan2(1, -1)=135<br>atan2(1, 1)=45                       |
| abs(x)<br>mag(x)                       |   | Absolute value (magnitude)                                    | abs(1)=1<br>mag(-10)=10                                  |
| abs(x, y, ...)<br>mag(x, y, ...)       |   | $\sqrt{x^2 + y^2 + \dots}$ , number of arguments not limited  | abs(3, 4)=5<br>mag(1, 1, 1, 1)=2                         |
| sign(x)                                |   | Sign x :<br>1 if x>0,<br>0 if x=0,<br>-1 if x<0               | sign(-2)=-1<br>sign(0)=0<br>sign(100)=1                  |
| db(x)                                  |   | 20*log <sub>10</sub> (abs(x)) : x in decibel                  | db(100)=40   |
| db(x, y)                               |   | 20*log <sub>10</sub> (abs(x/y)) : ratio x/y in decibel        | db(1, 10)=-20  |
| min(x, y, ...)                         |   | Minimum, number of arguments not limited                      | min(5, 9, 10, 2)=2                                       |
| max(x, y, ...)                         |   | Maximum, number of arguments not limited                      | max(5, 9, 10, 2)=10                                      |
| int(x)<br>round(x)                     |   | Round x to the nearest integer                                | int(1.4449)=1<br>int(1.5)=2                              |
| int(x, y)<br>round(x, y)               |   | Round x to the nearest multiple of y, x if y<=0.              | int(123, 10)=120<br>int(3.1415, 0.1)=3.1                 |
| abs(c)<br>mag(c)                       | √ | Absolute value (magnitude) of complex number <b>c</b>         | abs(3+4j)=5<br>mag(-3j)=3                                |
| phase(c)                               | √ | Phase of complex number <b>c</b> , -180...180 degrees         | phase(1+1j)=45   |
| re(c)                                  | √ | Real part of complex number <b>c</b>                          | re(2+3j)=2   |
| im(c)                                  | √ | Imaginary part of complex number <b>c</b>                     | im(2-3j)=-3  |
| par(x, y)                              | √ | (x*y)/(x+y) : x in parallel with y                            | par(2, 3)=1.2<br>par(10j, 10)=5+5j                       |
| par(x, y, ...)                         | √ | x, y, ... in parallel, number of arguments not limited        | par(1, 1, 1, 1)=0.25                                     |
| random(x)<br>rand(x)                   |   | Random number in the range 0...x                              | rand(1)=0.2937463  |
| limit(x, min, max)<br>lim(x, min, max) |   | Limit x:<br>min if x<min,<br>max if x>max<br>x if min<=x<=max | lim(0, -1, 1)=0<br>lim(-2, -1, 1)=-1<br>lim(10, -1, 2)=2 |
| islow(x)                               |   | 1 if x < logical threshold<br>0 otherwise                     | islow(1.0)=1<br>islow(55)=0                              |

| <b>Function</b> | <b>C</b> | <b>Description</b>                         | <b>Example</b>                |
|-----------------|----------|--|-------------------------------|
| ishigh(x)       |          | 1 if x >= logical threshold<br>0 otherwise | ishigh(1.0)=0<br>ishigh(55)=1 |

## 4. C language syntax

**Comments.** Use `//` to comment text until the end of the line, or delimiters `/*` and `*/` to comment block of the text. Delimiters `/*` and `*/` can be nested.

```
for( i=0; i<10; ++i ) { // this is a comment
    /* This block is commented out
    x=i*2;
    y=i/10;
    */
    x=i;
}
```

**Variables.** Only **double** type is supported. To declare a new variable:

- use keyword **double**,
- or assign some value to a new name,
- or use a new name in the **for** statement.

```
double x, y, z;
double x=1.0;
i=5;
for( n=0; n<10; ++n );
```

**Arrays.** Only one-dimensional arrays are supported. Index is zero-based. Use keyword **double** to declare a new array.

```
double x[100];
double y[] = { 1, 2, 3, 4, 5 };
```

**if...else.** Conditional statement.

```
if(i<=0) R1=1.0;
else if(i==1) R1=2.0;
else {
    R1=3.0;
    C1=1n;
}
```

**for.** Loop operator.

```
for( i=0; i<10; ++i ) {
    x[i]=2^i;
    y+=x[i];
}
```

**for.** “Foreach” loop operator. The code is executed for all values from the comma-separated list.

```
for( i=1,5,10,50,100 ) {
    y*=i;
}
```

**while.** Loop operator.

```
i=0;
while( i<10 ) {
    x[i]=2^i;
    ++i;
}
```

**do...while.** Loop operator.

```
i=0;
do {
    x[i]=2^i;
    ++i;
}
while( i<10 );
```

**switch.** Selective structure.

```
switch(i) {
    case 1: x=1; break;
    case 2: x=2; break;
    default: x=3; break;
}
```

**continue.** Skip the rest of the code in the current loop.

```
for( i=0; i<10; ++i ) {
    x[i]=2^i;
    if(i==5) continue;
    y+=x[i];
}
```

**break.** Leave current loop or **switch** statement.

```
for( i=0; i<10; ++i ) {
    x[i]=2^i;
    if(i==5) break;
    y+=x[i];
}
```

**return.** Stop execution of the code immediately and exit.

```
for( i=0; i<10; ++i ) {
    x[i]=2^i;
    if(x[i]==0) return;
    y/=x[i];
}
```

## 5. Script commands

*In alphabetical order.*

| Command                                      | Description  |
|--|--|
| <pre>ac [from[,to [,points [,scale]]]]</pre> | <p>Set AC analysis parameters and start AC analysis.<br/> from = start frequency<br/> to = stop frequency<br/> points = number of points<br/> scale =log or lin - logarithmic or linear frequency scale.</p> <p>If called from the script, command will not return until AC analysis is completed. If called from console or HTTP link, returns immediately. Use "ready" command to check for analysis completion.</p> <p>Example:<br/> ac;<br/> ac 1M;<br/> ac 1M, 100M;<br/> ac 1M, 100M, 500;<br/> ac 1M, 100M, 500, lin;</p> |
| <pre>clear</pre>                             | <p>Clear storage.</p> <p>Example:<br/> clear;</p>  |
| <pre>close</pre>                             | <p>Close active document.</p> <p>Example:<br/> close;</p>  |
| <pre>cont [screen[,step]]</pre>              | <p>Continue transient.<br/> screen = screen size<br/> step = calculation step</p> <p>If called from the script, command will not return until transient is completed. If called from console or HTTP link, returns immediately. Use "ready" command to check for transient completion.</p> <p>Example:<br/> cont;<br/> cont 1m;<br/> cont 1m, 10n;</p>   |
| <pre>cursors left,right</pre>                | <p>Show cursors (transient or AC) and set to specified positions.</p> <p>Example:<br/> cursors 1.5, 2.5;</p>   |

| Command   | Description  |
|---|--|
| cursors off   | Hide cursors (transient or AC).<br><br>Example:<br><pre>cursors off;</pre>   |
| display off<br>display on                                 | Hide or show transient and AC windows.<br><br>Example:<br><pre>display off;</pre>  |
| exit  | Close all documents and exit NL5. Can not be called from console command line.<br><br>Example:<br><pre>exit;</pre>   |
| <pre>export [filename[,from[,to[,step]]]]</pre>           | Export traces into csv file. If <i>filename</i> is omitted, name of the file to export is the same as script file name, with "csv" extension. If file path is not specified, export in the script file directory. Extension "csv" can be omitted. Number of points can not exceed "Max number of points" value defined on the Preferences Transient page. If <i>step</i> is omitted, export 101 points.<br><br><pre>from = start of the data interval to = end of the data interval step = step</pre> Example:<br><pre>export; export rctraces; export rctraces,0,100; export rctraces,0,1,1m;</pre> |
| <pre>export [filename [from[,to[,points[,scale]]]]]</pre> | Export AC traces into csv file. If <i>filename</i> is omitted, name of the file to export is the same as script file name, with "csv" extension. If file path is not specified, export in the script file directory. Extension "csv" can be omitted.<br><pre>from = start frequency to = stop frequency points = number of points scale =log or lin - logarithmic or linear frequency scale</pre> Example:<br><pre>export; export actraces; export actraces,1m,1k; export actraces,1m,1k,100; export actraces,1m,1k,100,lin;</pre>   |

| Command                        | Description  |
|--------------------------------|--|
| open <i>filename</i>           | <p>Open schematic file <i>filename</i>. Extension "nl5" can be omitted. If file path is not specified, search in the script file directory.</p> <p>Example:<br/> <pre>open "c:Project files/nl5/rc.nl5"; open rc;</pre> </p>   |
| pause                          | <p>Pause transient. Command can be called from console command line only.</p> <p>Example:<br/> <pre>pause;</pre> </p>  |
| ready                          | <p>Check if transient or AC analysis is completed. Returns "0" if analysis is running, returns "1" if completed.</p> <p>Example:<br/> <pre>ready;</pre> </p>   |
| rununtil [ <i>expression</i> ] | <p>Set up "run until" transient mode. If parameter is omitted, turn off "run until" mode and clear "run until" expression. Otherwise turn on "run until" mode and use parameter as "run until" expression.</p> <p>Example:<br/> <pre>rununtil; rununtil V(C1)&lt;0;</pre> </p>   |
| save [ <i>filename</i> ]       | <p>Save schematic to file <i>filename</i>. Extension "nl5" can be omitted. If <i>filename</i> is omitted, save into the same file. If file path is not specified, save in the script file directory.</p> <p>Example:<br/> <pre>save; save rcnew;</pre> </p>  |
| savedata [ <i>filename</i> ]   | <p>Save traces into "nlt" data file. Extension "nlt" can be omitted. If <i>filename</i> is omitted, name of the file to save data is the same as script file name, with "nlt" extension. If file path is not specified, save in the script file directory.</p> <p>Example:<br/> <pre>savedata; savedata rctraces;</pre> </p> |



| Command  | Description   |
|--|---|
| saveic   | <p>Save Initial Conditions.</p> <p>Example:<br/>saveic;</p>   |
| sleep <i>time</i>  | <p>Pause script execution for <i>time</i> ms.</p> <p>Example:<br/>sleep 1000;</p>   |
| stop   | <p>Stop transient. This command can be used to free memory allocated for transient analysis. Transient can not be continued after this command.</p> <p>Example:<br/>stop;</p>   |
| store [ <i>expr</i> ]  | <p>Move run into storage. The parameter is evaluated as an expression, and the result is used as a storage name. If parameter is omitted, a default storage name "RunN" is used.</p> <p>Example:<br/>store;<br/>store R1*C1;</p>  |
| storetext [ <i>text</i> ]                                      | <p>Move run into storage with parameter as a storage name. If parameter is omitted, a default storage name "RunN" is used.</p> <p>Example:<br/>storetext;<br/>storetext This is first run;</p>  |
| <i>tracename</i> [ <i>from</i> [, <i>to</i> [, <i>step</i> ]]] | <p>Request transient trace data as a comma-separated string. Trace <i>tracename</i> should be specified in the Transient Data, however it does not need to be displayed on the graph or in the table. Number of points can not exceed "Max number of points" value defined on the Preferences Transient page. Can be called from HTTP link only.</p> <p>from = start of the data interval<br/>to = end of the data interval<br/>step = step</p> <p>Example:                   // Returns:<br/>V(R1);                   // 101 points of the entire trace<br/>V(R1) 1.23;              // trace value at t=1.23<br/>V(R1) 0,100;             // 101 points in the specified<br/>                          // interval<br/>V(R1) 0,10,0.1;         // specified interval and step</p> |

| Command   | Description   |
|---|---|
| <pre>tracename [from[, to [, points[, scale]]]]</pre> | <p>Request AC trace data as a comma-separated string. Trace <i>tracename</i> should be specified in the AC Data, however it does not need to be displayed on the graph or in the table. Can be called from HTTP link only.</p> <p>from = start frequency<br/> to = stop frequency<br/> points = number of points<br/> scale =log or lin - logarithmic or linear frequency scale</p> <p>Example:                   // Returns:<br/> V(R1);                    // all calculated AC points<br/> V(R1) 1.23;               // trace value at t=1.23<br/> V(R1) 1,100;             // specified interval<br/> V(R1) 1,100,10;         // specified interval and number<br/>                           // of points<br/> V(R1) 1,100,10,lin;     // specified interval, number<br/>                           // of points and frequency scale</p> |
| <pre>tran [start[, screen[, step]]]</pre>             | <p>Set transient parameters and start transient</p> <p>start = start of transient display<br/> screen = screen size<br/> step = calculation step</p> <p>If called from the script, command will not return until transient is completed. If called from console or HTTP link, returns immediately. Use "ready" command to check for transient completion.</p> <p>Example:<br/> tran;<br/> tran 0, 10m;<br/> tran 0, 10m, 1mk;</p>   |

## 6. END USER LICENSE AGREEMENT

This End-User License Agreement ("EULA", "Agreement") is a legal agreement between you ("you", either an individual or a single entity) and Sidelinesoft, LLC ("Sidelinesoft") for the NL5 Circuit Simulator software ("the Software", "the Software Product"), NL5 License ("the Software License"), and accompanying documentation.

### Ownership

The Software, any accompanying documentation, and all intellectual property rights therein are owned by Sidelinesoft. The Software is licensed, not sold. The Software is protected by copyright laws and treaties, as well as laws and treaties related to other forms of intellectual property. The Licensee's license to download, use, copy, or change the Software Product is subject to these rights and to all the terms and conditions of this Agreement.

### Acceptance

YOU ACCEPT AND AGREE TO BE BOUND BY THE TERMS OF THIS AGREEMENT BY DOWNLOADING THE SOFTWARE PRODUCT OR BY INSTALLING, USING, OR COPYING THE SOFTWARE PRODUCT. YOU MUST AGREE TO ALL OF THE TERMS OF THIS AGREEMENT BEFORE YOU WILL BE ALLOWED TO DOWNLOAD THE SOFTWARE PRODUCT. IF YOU DO NOT AGREE TO ALL OF THE TERMS OF THIS AGREEMENT, YOU MUST NOT INSTALL, USE, OR COPY THE SOFTWARE PRODUCT.

### License Grant

Sidelinesoft grants you a right to download, install, and use unlimited copies of the Software Product. Without a Software License, the Software operates as a Demo version, with limited number of components in the schematic, and possibly some functional and performance limitations. Several types of Full-Function Software Licenses can be obtained at Software Product website ([nl5.sidelinesoft.com](http://nl5.sidelinesoft.com)). Terms and conditions of each type of Full-Function Software License are available at the website and are subject to change without notice.

### Restrictions on Reverse Engineering, Decompilation, and Disassembly.

You may not decompile, reverse-engineer, disassemble, or otherwise attempt to derive the source code for the Software Product.

### Restrictions on Alteration

You may not modify the Software Product or create any derivative work of the Software Product or its accompanying documentation without obtaining permission of Sidelinesoft. Derivative works include but are not limited to translations. You may not alter any files or libraries in any portion of the Software Product.

### Consent to Use of Data

Sidelinesoft may ask for your permission to collect and use technical information gathered as part of the product support services provided to you, if any, related to the Software. Sidelinesoft may use this information solely to improve the Software or to provide customized services to you and will not disclose this information in a form that personally identifies you.

### Disclaimer of Warranties and Limitation of Liability

UNLESS OTHERWISE EXPLICITLY AGREED TO IN WRITING BY SIDELINESOFT, SIDELINESOFT MAKES NO OTHER WARRANTIES, EXPRESS OR IMPLIED, IN FACT OR IN

LAW, INCLUDING, BUT NOT LIMITED TO, ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE OTHER THAN AS SET FORTH IN THIS AGREEMENT.

Sidelinesoft makes no warranty that the Software Product will meet your requirements or operate under your specific conditions of use. Sidelinesoft makes no warranty that operation of the Software Product will be secure, error free, or free from interruption. YOU MUST DETERMINE WHETHER THE SOFTWARE PRODUCT SUFFICIENTLY MEETS YOUR REQUIREMENTS FOR SECURITY AND UNINTERRUPTABILITY. YOU BEAR SOLE RESPONSIBILITY AND ALL LIABILITY FOR ANY LOSS INCURRED DUE TO FAILURE OF THE SOFTWARE PRODUCT TO MEET YOUR REQUIREMENTS. UNDER NO CIRCUMSTANCES SHALL SIDELINESOFT BE LIABLE TO YOU OR ANY OTHER PARTY FOR INDIRECT, CONSEQUENTIAL, SPECIAL, INCIDENTAL, PUNITIVE, OR EXEMPLARY DAMAGES OF ANY KIND (INCLUDING LOST REVENUES OR PROFITS OR LOSS OF BUSINESS) RESULTING FROM THIS AGREEMENT, OR FROM THE PERFORMANCE, INSTALLATION, USE OR INABILITY TO USE THE SOFTWARE PRODUCT, WHETHER DUE TO A BREACH OF CONTRACT, BREACH OF WARRANTY, OR THE NEGLIGENCE OF SIDELINESOFT OR ANY OTHER PARTY, EVEN IF SIDELINESOFT IS ADVISED BEFOREHAND OF THE POSSIBILITY OF SUCH DAMAGES. TO THE EXTENT THAT THE APPLICABLE JURISDICTION LIMITS SIDELINESOFT'S ABILITY TO DISCLAIM ANY IMPLIED WARRANTIES, THIS DISCLAIMER SHALL BE EFFECTIVE TO THE MAXIMUM EXTENT PERMITTED.

#### **Limitation of Remedies and Damages**

Your remedy for a breach of this Agreement or of any warranty included in this Agreement is the correction or replacement of the Software Product. Selection of whether to correct or replace shall be solely at the discretion of Sidelinesoft. Any claim must be made within the applicable warranty period. All warranties cover only defects arising under normal use and do not include malfunctions or failure resulting from misuse, abuse, neglect, alteration, improper installation, or a virus. All limited warranties on the Software Product are granted only to you and are non-transferable. You agree to indemnify and hold Sidelinesoft harmless from all claims, judgments, liabilities, expenses, or costs arising from your breach of this Agreement and/or acts or omissions.

#### **Severability**

If any provision of this Agreement shall be held to be invalid or unenforceable, the remainder of this Agreement shall remain in full force and effect. To the extent any express or implied restrictions are not permitted by applicable laws, these express or implied restrictions shall remain in force and effect to the maximum extent permitted by such applicable laws.

#### **Termination**

This Agreement is effective until terminated. Without prejudice to any other rights, Sidelinesoft may terminate this Agreement if you fail to comply with the terms and conditions of this Agreement. In such event, you must destroy all copies of the Software License.

#### **Governing Law, Dispute Resolution**

This Agreement is governed by the laws of the State of Colorado, U.S.A., without regard to its choice of law principles to the contrary.

#### **Contact Information.**

Any inquiries regarding this Agreement or the Software may be addressed to Sidelinesoft at the Software Product website ([nl5.sidelinesoft.com](http://nl5.sidelinesoft.com)).

*The end*