

# Модуль экспорта IDF

**19 октября 2016 г.**

# Содержание

**1 Знакомство с модулем экспорта IDFv3** **2**

**2 Определение моделей компонентов, используемых модулем экспорта** **2**

**3 Создание файла контура компонента** **4**

**4 Рекомендации по созданию контуров** **6**

4.1 Именование корпусов . . . . . 6

4.2 Комментарии . . . . . 6

4.3 Записи Наименование контура и Наименование компонента . . . . . 6

4.4 Ориентация и положение контактов . . . . . 6

4.5 Советы по размерам . . . . . 8

**5 Инструменты IDF-контуров компонентов** **8**

5.1 idfcyl . . . . . 8

5.2 idfrect . . . . . 9

5.3 dxf2idf . . . . . 10

**6 idf2vrml** **11**

*Руководство пользователя*

**Авторские права**

Авторские права на данный документ © 2014-2015 принадлежит его разработчикам (соавторам), перечисленным ниже. Вы можете распространять и/или изменять его в соответствии с правилами лицензии GNU General Public License (http://www.gnu.org/licenses/gpl.html), версии 3 или более поздней, или лицензии типа Creative Commons Attribution License(http://creativecommons.org/licenses/by/3.0/), версии 3.0 или более поздней.

Все торговые знаки этого руководства принадлежат его владельцам.

**Соавторы**

Cirilo Bernardo

**Перевод**

KiCad Russian Team + alex9

**Обратная связь**

Просьба оставлять все комментарии и замечания на следующих ресурсах:

- О документации KiCad: https://github.com/KiCad/kicad-doc/issues
- О программном обеспечении KiCad: https://bugs.launchpad.net/kicad
- О переводе программного обеспечения KiCad: https://github.com/KiCad/kicad-i18n/issues

**Дата публикации**

5 мая 2016 года.

# 1 Знакомство с модулем экспорта IDFv3

Модуль экспорта IDF создаёт IDFv3-совместимые [1] файлы платы (.emn) и библиотеки (.emp) для передачи физических размеров в машиностроительную САПР. Модуль в настоящее время экспортирует контур платы с вырезами, все контактные и монтажные сквозные отверстия, включая отверстия с пазами, и контуры компонентов. Это основной набор данных, необходимых для взаимодействия с разработчиками механических конструкций. Все остальные объекты, описанные в спецификации IDFv3, в настоящее время не экспортируются.

# 2 Определение моделей компонентов, используемых модулем экспорта

Модуль экспорта IDF использует атрибуты файла 3D-модели, которые изначально использовались программами 3D-просмотра. Поскольку программы 3D-просмотра, IDF и возможные будущие модули экспорта для машиностроительных САПР используют различающиеся типы файловых форматов, то допустимо использование атрибутов файла 3D-модели для определения моделей в различных модулях экспорта.

Находясь в редакторе посадочных мест или Pcbnew, откройте диалог ”Свойства посадочного места”, перейдите на вкладку ”Настройки 3D” (см. рис. 1), нажмите на кнопку ”Добавить 3D форму” и выберите фильтр ”Файлы компонентов IDFv3 (* .idf) ” (см. рис. 2). Выберите нужный файл контура компонента и введите необходимые значения для смещения и вращения. Обратите внимание, что модулем экспорта IDF используются только значения смещения и вращение по оси Z; все остальные значения игнорируются. Смещения должны быть указаны с использованием выходных единиц измерения IDF-файла платы (мм или мил) и в системе координат IDF, то есть правосторонней системе координат: +Z направлено в сторону наблюдателя, +X – вправо и +Y – вверх. Вращение должно быть в градусах и положительный поворот соответствует вращению против часовой стрелки, как это описано в спецификации IDFv3. Несколько контуров компонентов могут быть объединены с соответствующими смещениями для создания простых сборок, таких как DIP-корпус в панели. [ **BUG:** в обсуждениях было принято решение о том, что единица смещения по оси Z должна быть дюйм, что согласуется с единицами смещения VRML-модели. Может быть также полезным не игнорировать значения (X, Y) смещения. Такое поведение в будущем изменится.]

После того как модели были определены, для всех нужных компонентов, находясь в программе Pcbnew, выберите меню **Файл**, затем **Экспорт** и, наконец, **IDFv3**. Появится диалоговое окно (см . рис. 3), которое позволяет задать имя файла и выходные единицы измерения IDF (мм или мил). Полученные IDF-файлы можно просматривать в свободной машиностроительной САПР FreeCAD или преобразовать в VRML с помощью инструмента idf2vrml и просматривать с помощью любой подходящей программы просмотра VRML.

[1] http://www.simplifiedsolutionsinc.com/images/idf_v30_spec.pdf

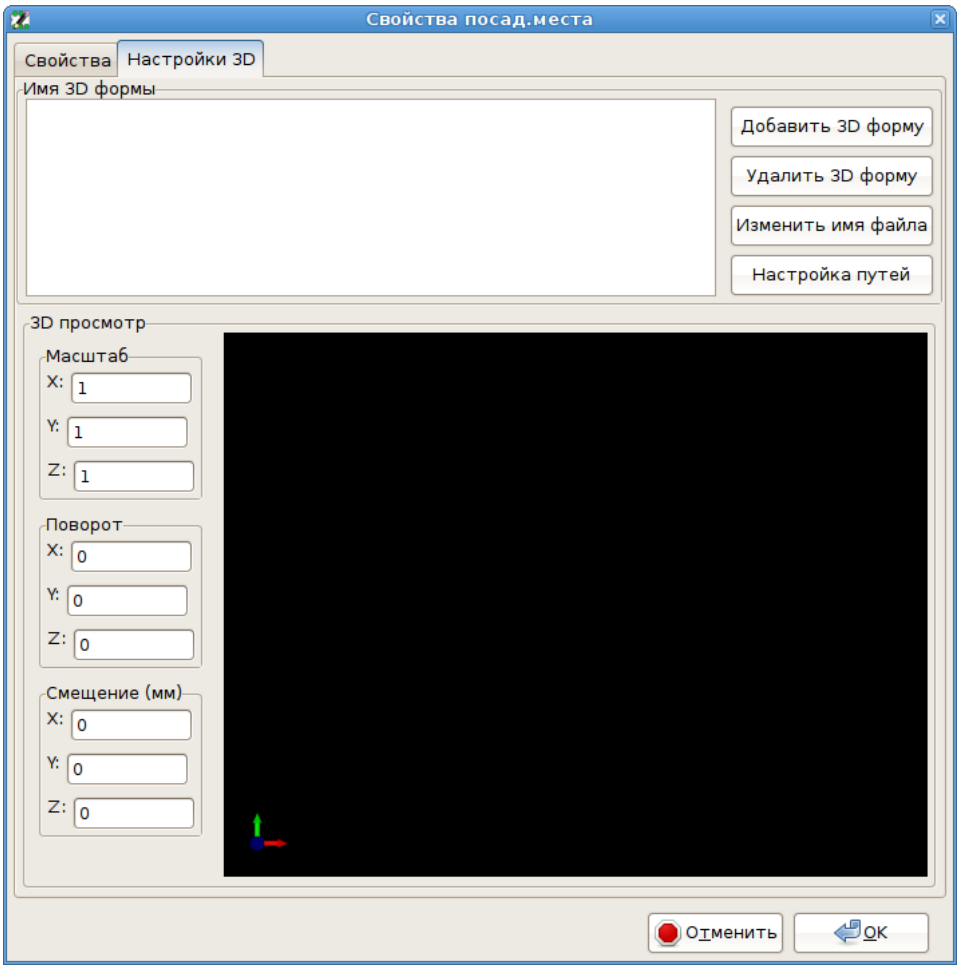


Рис. 1: Свойства посадочных мест, настройки 3D

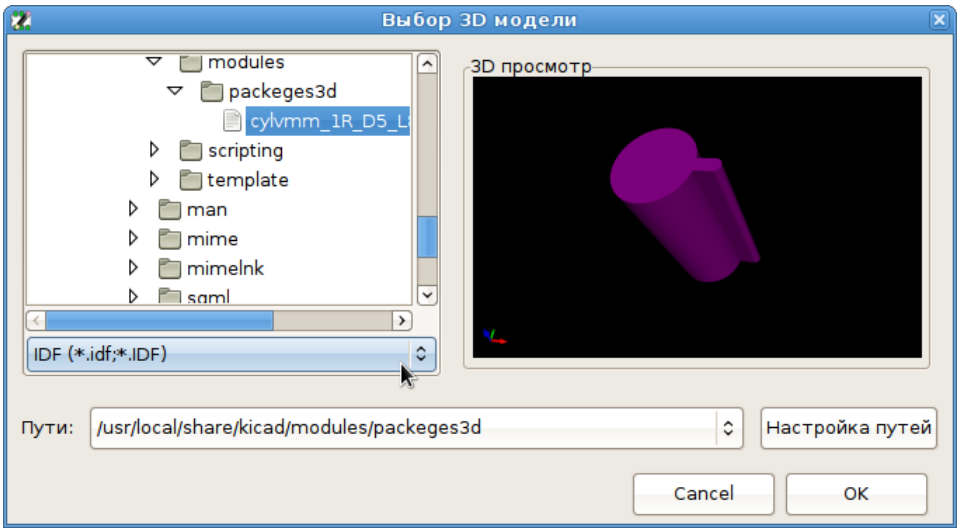


Рис. 2: Выбор IDF-контура компонента

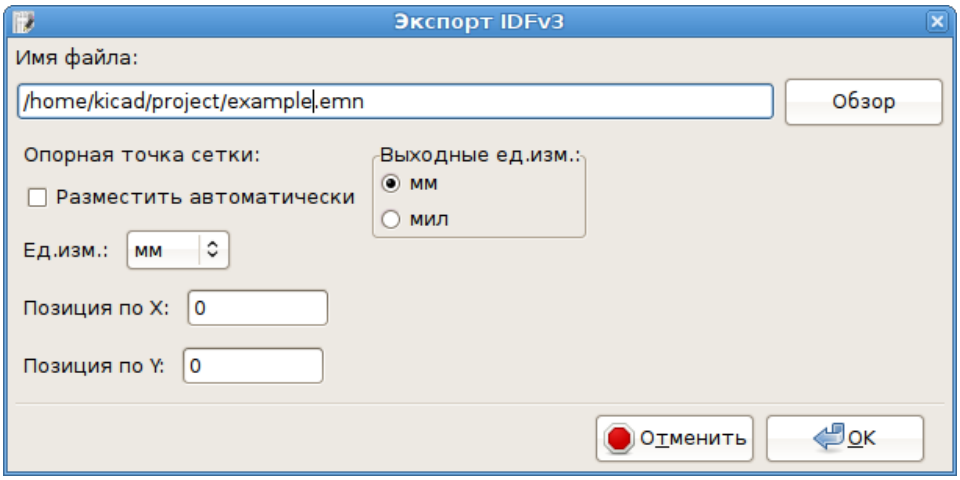


Рис. 3: Настройки для вывода IDF

# 3 Создание файла контура компонента

Файл контура компонента (* .idf) состоит из одной .ELECTRICAL или .MECHANICAL секции, как описано в спецификации. Секции может предшествовать любое количество строк комментария; комментарий копируются модулем экспорта в файл библиотеки и может использоваться для отслеживания метаданных, таких как ссылки на документы, используемые для определения контуров и габаритных размеров компонентов.

Секция контура компонента содержит поля, которые являются строками, целыми числами или числами с плавающей точкой (десятичным разделителем должна быть именно точка, а не запятая). Строка представляет собой комбинацию символов, которые могут включать в себя пробелы; если строка содержит пробелы, то она должна быть заключена в кавычки. Кавычки не должны использоваться внутри строки. Числа с плавающей точкой могут быть представлены в виде десятичной дроби или в экспоненциальном виде, но применение десятичных дробей является предпочтительным для человеческого восприятия. Файл IDF должен состоять только из 7-битных символов ASCII; использование 8-битных символов приведет к непредсказуемому поведению.

IDF-файл состоит из секций, которые состоят из записей, состоящих из полей. В IDF-файлах контуров может присутствовать только один тип секции и это должен быть либо .ELECTRICAL, либо .MECHANICAL. Запись представляет собой одну строку текста и может содержать одно или несколько полей. Поля представляют собой последовательности символов, разделенных одним или несколькими пробелами. Все поля записи должны располагаться в одной строке. Записи и поля могут быть любой длины.

Заголовком секции (.ELECTRICAL или .MECHANICAL) считается первая запись (Запись 1) секции. За Записью 1 должна следовать Запись 2, которая имеет четыре поля:

1. Наименование контура: строка, которая в сочетании с Полем 2 должна сформировать уникальный идентификатор для контура компонента. Для стандартных корпусов лучше использовать их наименование, например, ”SOT-23”. Для уникальных корпусов – наименование компонента, заданное производителем (part number).

2. Наименование компонента: не смотря на очевидное указание использовать наименование компонента (например, BS107), в этой строке лучше сохранить вспомогательное описание корпуса. Например, если значение Поля 1 – ”TO-

92 ", то Поле 2 можно использовать для описания расположения контактных площадок или ориентации конкретно этого файла контура TO-92.

3. Определение единиц измерения: здесь должно быть либо MM, либо THOU. Относится только к этому компоненту.
4. Высота компонента: число с плавающей точкой, представляющее номинальную высоту компонента (от поверхности установки) с использованием единиц измерения, указанных в Поле 3.

После Записи 2 должны следовать несколько Записей 3, которые определяют контур компонента. Запись 3 состоит из четырех полей:

1. Направление обхода: 0 (точки контура указаны в направлении против часовой стрелки) или 1 (по часовой стрелке).
2. Значение координаты X: число с плавающей точкой
3. Значение координаты Y: число с плавающей точкой
4. Угол сегмента: число с плавающей точкой. Если значение равно 0, то рисуется отрезок прямой от предыдущей точки до этой. Если значение не равно 0 , то рисуется дуга от предыдущей точки до этой; отрицательное значение указывает на направление по часовой стрелке, положительное – против часовой стрелки. Если значение равно 360, то рисуется окружность с центром в предыдущей точке, а эта точка располагается на окружности (никогда не указывайте окружность, используя значение -360: по меньшей мере один крупный пакет машиностроительной САПР в такой ситуации ведет себя не очень хорошо).

Допускается описание только одного замкнутого контура. Т.о. не представляется возможным отобразить вырез внутри контура. Координаты последней точки должны быть таким же, как первой (если контур не представляет собой окружность).

Пример IDF-файла 1:

```
# a simple cylinder - this could represent an electrolytic capacitor
.ELECTRICAL
    "cylinder" "5mm OD, 5mm height" MM 5
    0 0 0 0
    0 2.5 0 360
.END_ELECTRICAL
```

Пример IDF-файла 2:

```
# an upside-down T
# a comment added for the sake of adding comments
.ELECTRICAL
    "Capital T" "5x8x10mm, upside down" MM 10
    0 -0.5 8 0
    0 -0.5 0.5 0
    0 -2.5 0.5 0
    0 -2.5 -0.5 180
    0 2.5 -0.5 0
    0 2.5 0.5 180
    0 0.5 0.5 0
    0 0.5 8 0
    0 -0.5 8 180
.END_ELECTRICAL
```

# 4 Рекомендации по созданию контуров

При создании контуров и, особенно, при использовании результатов работы совместно с другими разработчиками, последовательность в проектировании и именовании файлов помогает людям находить файлы быстрее и размещать компоненты с минимальными конфликтами.

## 4.1 Именование корпусов

Попробуйте сделать некоторые сведения о контуре доступными в имени файла, чтобы дать пользователю общее представление о том, чем является контур. Например, цилиндрические корпуса с аксиальными выводами могут представлять как некоторые типы конденсаторов, так и резисторов. Так что имеет смысл определить контур как горизонтальное или вертикальное устройство с аксиальными выводами и добавить некоторую дополнительную информацию о размерах, являющихся наиболее важными: диаметр, длина и шаг. Если устройство имеет уникальный контур, то наименование компонента, заданное производителем (part number), и префикс, чтобы указать класс устройства, будут достаточны.

## 4.2 Комментарии

Используйте комментарии в IDF-файле, чтобы предоставить пользователям больше информации о контуре. Например, приведите ссылку на источник, используемый для получения информации о геометрии корпуса.

## 4.3 Записи Наименование контура и Наименование компонента

Тщательно продумайте о значениях, которые хотите указать в записях Наименование контура и Наименование компонента. Взятые вместе, эти строки выступают в качестве уникального идентификатора для системы машиностроительной САПР. Значения строк могут иметь какое-то значение для пользователя, но в этом нет необходимости: значения в первую очередь предназначены для САПР. В идеале, выбор значений будет уникальным в пределах любой большой коллекции контуров. Правильный выбор значений также приведет к уменьшению числа конфликтов, особенно в сложных платах.

## 4.4 Ориентация и положение контактов

Для выводных компонентов не существует общепринятых схем для определения ориентации выводов и центра компонента в 3D-модели. Для обеспечения согласованности, если у компонента есть только 2 вывода, то они должны располагаться горизонтально (см. рис. 4) вдоль оси X; для 3-х выводов так же пытайтесь 2 из них расположить горизонтально на оси X. У поляризованных устройств, такие как электролитические или танталовые конденсаторы, положительный вывод должен быть на Контакте 1, а у диода на Контакте 1 должны быть катод; это необходимо для поддержания совместимости УГО с ориентацией, определенной для компонентов поверхностного монтажа; однако необходимо отметить, что на многих существующих в KiCad УГО и посадочных местах на Контакте 1 расположен анод.

**Замечание**

В последней редакции посадочных мест KiCad на GitHub анод теперь расположен на Контакте 2 как для выводных компонентов, так и для компонентов поверхностного монтажа.

Для DIP-корпусов центр контура должен быть в центре прямоугольника, описывающего расположение контактов, и Вывод 1 желательно располагать в верхнем левом углу; это обеспечит определенную согласованность со стандартизованной ориентацией компонентов поверхностного монтажа; тем не менее, такая модель будет повернута на -90 градусов по отношению к большинству существующих посадочных мест компонентов KiCad и моделей VRML. Для таких элементов, как горизонтально расположенный конденсатор с радиальными выводами или горизонтальный корпус TO-220, предпочтительно размещать контакты в ряд по оси X и с корпусом устройства, направленным вверх (см. рис. 4). Неполярные вертикально расположенные компоненты с радиальными выводами должны иметь контакт с правой стороны. Для полярных вертикально расположенных компонентов с радиальными выводами контакт располагают в зависимости от того, где расположен Вывод 1: если он находится на нижнем конце, то контакт располагают справа, а если – на верхнем конце, то – слева.

**Замечание**

В текущей версии модулей посадочных мест KiCad у выводных компонентов контакты расположены вдоль оси Y, а не оси X, и в начале координат находится Вывод 1, а не центр корпуса. Ориентируйте и располагайте контур компонента с учетом конкретного посадочного места; это позволит избежать необходимости указывать ненулевой поворот для IDF-контура этого компонента. Так как модуль экспорта IDF в настоящее время игнорирует значения смещения (X, Y), то крайне важно использовать правильную привязку в IDF-контуре компонента.

Для компонентов поверхностного монтажа ориентация, центр корпуса и контур определяются различными стандартами. Используйте стандарт, подходящий для работы. Также имейте в виду, что многие компоненты не соответствуют никаким стандартам; в таких случаях этот компонент, вероятно, лучше всего идентифицировать в имени файла контура с помощью наименования компонента, заданного производителем (part number). В общем случае, контур компонента поверхностного монтажа представляет собой прямоугольник, охватывающий корпус и выводы; корпус ориентирован таким образом, что Вывод 1 расположен как можно ближе к верхнему левому углу, а верхний левый угол обычно скошен для визуального определения.

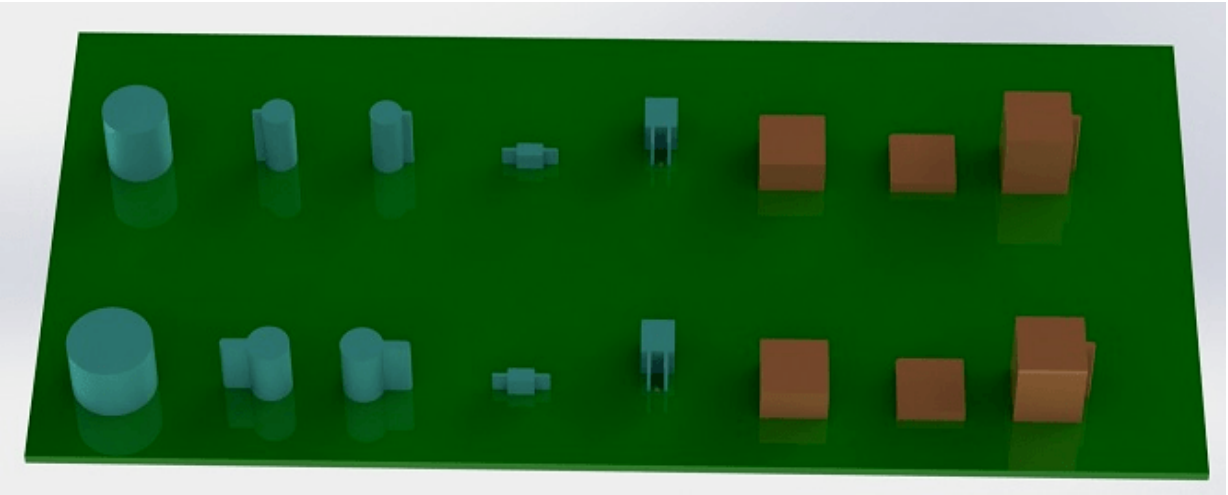

Рис. 4: Пример контуров, сгенерированных программами idfcyl и idfrect, открытый в SolidWorks.

Слева направо: (а) вертикальный цилиндр с радиальными выводами, (б) вертикальный цилиндр с аксиальными выводами с контактом слева, (в) вертикальный цилиндр с аксиальными выводами с контактом справа, (г) горизонтальный цилиндр с

аксиальными выводами, (д) горизонтальный цилиндр с радиальными выводами, (е) ровный квадратный контур, (ж) квадратный контур со скошенным левым верхним углом (скос очень маленький), (з) квадратный контур с аксиальными выводами с контактом справа. Верхние контуры были определены в миллиметрах, а нижние – в дюймах.

### 4.5 Советы по размерам

Цель использования выдавленных контуров – дать конструктору некоторое представление о расположении и физическом пространстве, занимаемом каждым компонентом. Обычно конструктор заменяет некоторые грубые контуры на более подробные механические модели, например, чтобы при проверке убедиться, что светодиод, установленный под прямым углом, будет вписываться в отверстие на панели. В большинстве случаев точность наброска не имеет значения, но это хорошая практика создания контуров, передающих больше конструкторской информации. В некоторых случаях пользователь может захотеть установить компонент в корпус с очень маленьким свободным пространством, например, в портативный музыкальный проигрыватель. В такой ситуации, когда большинство выдавленных контуров достаточно хорошо представляют компоненты, конструктор может заменить очень мало моделей при проектировании корпуса. Если контуры не достоверно отражают действительность, то конструктор потратит много времени на замену моделей, чтобы обеспечить хорошую подгонку. В конце концов, отдав мусор нельзя рассчитывать на получение конфетки. И напротив, предоставив наиболее полные данные можно быть уверенным в быстром и хорошем результате.

## 5 Инструменты IDF-контуров компонентов

Доступен ряд инструментов командной строки, чтобы помочь при создании IDF-контуров:

1. **idfcyl:** создает контур цилиндра вертикальной или горизонтальной ориентации и с аксиальными или радиальными выводами
2. **idfrect:** создает прямоугольный контур, который может иметь либо аксиальный вывод, либо скос верхнего левого угла
3. **dxf2idf:** преобразует рисунок в формате DXF в IDF-контур компонента

### 5.1 idfcyl

Когда **idfcyl** вызывается без аргументов, он печатает примечание об использовании и краткую информацию о входных данных:

```
idfcyl: This program generates an outline for a cylindrical component.
    The cylinder may be horizontal or vertical.
    A horizontal cylinder may have wires at one or both ends.
    A vertical cylinder may have at most one wire which may be
    placed on the left or right side.

Input:
    Unit: mm, in (millimeters or inches)
    Orientation: V (vertical)
    Lead type: X, R (axial, radial)
```

```
    Diameter of body
    Length of body
    Board offset
    *   Wire diameter
    *   Pitch
    **  Wire side: L, R (left, right)
    *** Lead length
    File name (must end in *.idf)

    NOTES:
        *   only required for horizontal orientation or
            vertical orientation with axial leads

        **  only required for vertical orientation with axial leads

        *** only required for horizontal orientation with radial leads
```

Примечания могут быть подавлены путем ввода любой произвольного аргумента в командной строке. Пользователь может ввести информацию вручную в командной строке или создавать сценарии для генерации контуров. Следующий скрипт создаёт один контур в виде цилиндра с аксиальными выводами с контактом справа:

```
#!/bin/bash
# Generate a cylindrical IDF outline for test purposes
# vertical 5mm cylinder,  nominal length 8mm + 3mm board offset,
# axial wire on right,  0.8mm wire dia., 3.5mm pitch
idfcyl - 1 > /dev/null <<  _EOF
mm
v
x
5
8
3
0.8
3.5
r
cylvmm_1R_D5_L8_Z3_WD0.8_P3.5.idf
_EOF
```

## 5.2 idfrect

Когда **idfrect** вызывается без аргументов, он печатает примечание об использовании и краткую информацию о входных данных:

```
idfrect: This program generates an outline for a rectangular component.
    The component may have a single lead (axial) or a chamfer on the
    upper left corner.
Input:
```

```
    Unit: mm, in (millimeters or inches)
    Width:
    Length:
    Height:
    Chamfer: length of the 45 deg. chamfer
    *  Leaded: Y,N (lead is always to the right)
    ** Wire diameter
    ** Pitch
    File name (must end in *.idf)

    NOTES:
        *   only required if chamfer = 0

        **  only required for leaded components
```

Примечания могут быть подавлены путем ввода любой произвольного аргумента в командной строке. Пользователь может ввести информацию вручную в командной строке или создавать сценарии для генерации контуров. Следующий скрипт создаёт два контура в виде прямоугольника (один – со скошенным углом, другой – с аксиальным выводом):

```
#!/bin/bash
# Generate various rectangular IDF outlines for test purposes
# 10x10, 1mm chamfer, 2mm height
idfrect - 1 > /dev/null <<  _EOF
mm
10
10
2
1
rectMM_10x10x2_C0.5.idf
_EOF
# 10x10x12,  0.8mm lead on 6mm pitch
idfrect - 1 > /dev/null <<  _EOF
mm
10
10
12
0
Y
0.8
6
rectLMM_10x10x12_D0.8_P6.0.idf
_EOF
```

## 5.3 dxf2idf

Файл DXF, используемый для задания контура компонента, может быть подготовлен с помощью свободного программного обеспечения LibreCAD (для лучшей совместимости). Когда **dxf2idf** вызывается без аргументов, он печатает примечание

об использовании и краткую информацию о входных данных:

```
dxf2idf: this program takes line, arc, and circle segments
    from a DXF file and creates an IDF component outline file.

Input:
    DXF filename: the input file, must end in '.dxf'
    Units: mm, in (millimeters or inches)
    Geometry Name: string, as per IDF version 3.0 specification
    Part Name: as per IDF version 3.0 specification of Part Number
    Height: extruded height of the outline
    Comments: all non-empty lines are comments to be added to
        the IDF file. An empty line signifies the end of
        the comment block.
    File name: output filename, must end in '.idf'
```

Примечания могут быть подавлены путем ввода любой произвольного аргумента в командной строке. Пользователь может ввести информацию вручную в командной строке или создавать сценарии для генерации контуров. Следующий скрипт создает контур высотой 5 мм из DXF-файла *test.dxf*:

```
#!/bin/bash
# Generate an IDF outlines from a DXF file
dxf2idf - 1 > /dev/null << _EOF
test.dxf
mm
DXF TEST GEOMETRY
DXF TEST PART
5
This is an IDF test file produced from the outline 'test.dxf'
This is a second IDF comment to demonstrate multiple comments

test_dxf2idf.idf
_EOF
```

# 6 idf2vrml

Инструмент idf2vrml читает набор из одного IDF-файла платы (.emn) и одного IDF-файла библиотеки (.emp) и создает файл VRML, который можно просматривать с помощью программы просмотра VRML. Эта функция полезна для визуализации сборки платы в тех случаях, когда пользователь не имеет доступа к программному обеспечению машиностроительной САПР. Вызов idf2vrml без каких-либо аргументов приведет к отображению подсказки по использованию:

```
>./idf2vrml
Usage: idf2vrml -f input_file.emn -s scale_factor {-k} {-d} {-z} {-m}
flags:
   -k: produce KiCad-friendly VRML output; default is compact VRML
   -d: suppress substitution of default outlines
   -z: suppress rendering of zero-height outlines
```

```
   -m: print object mapping to stdout for debugging purposes
example to produce a model for use by KiCad: idf2vrml -f input.emn -s 0.3937008 -k
>
```

[**BUG:** Инструмент idf2vrml в настоящее время не корректно отображает объекты **OTHER_OUTLINE** в emn-файле, если они установлены на нижнем слое печатной платы; но эта ошибка не будет заметна, если использовать файлы, экспортируемые KiCad, так как механизм для определения такого объекта отсутствует. По сути, это ошибка проявляется в редких случаях при обработке emn-файла стороннего разработчика, который использовал объект на нижней стороне платы.]