

# Физический рисунок

v0.1. Актуальная версия [здесь](#)

## 1 Общие положения

### 1.1 Начало работы

Для начала надо [отсюда](#) скачать `design.sty` в директорию вашего документа для использования наших предопределенных команд и стилей

Рисунки выполняются в окружении `tikzpicture`, которое можно встраивать внутрь `figure`:

```
% in preamble:
\usepackage{design}

% in document:
\begin{figure}[H]
  \centering
  \begin{tikzpicture}
    \draw (0,0) -- (2,0);
  \end{tikzpicture}
\end{figure}
```

Этот кусок кода рисует отрезок соединяющий точки с координатами  $(0, 0)$  и  $(2 \text{ cm}, 0)$ . Стандартной единицей длины в `tikz` являются сантиметры, но можно, например, в качестве координаты точки использовать  $(2\text{pt}, 0)$ .

### 1.2 Стрелочки и подписи

Для стрелочек используются аргумент `[>=latex']`, который можно указывать непосредственно при отрисовке каждого объекта, но проще его вынести в аргумент всего изображения:

```
\begin{tikzpicture}[>=latex']
  \draw[->,>=latex'] (0,0) -- ++(2,0);
  \draw (0,0) -- ++ (2/2,0) node [below] {Правильная стрелочка};
\end{tikzpicture}
```

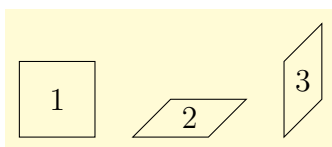
Также здесь продемонстрирован пример подписи объекта с использованием команды `node`. Запись `++(2,0)` означает сдвиг относительно предыдущей координаты на 2 см вдоль горизонтальной оси.

→  
Правильная стрелочка

→  
Неправильная стрелочка

### 1.3 Преобразования координат

Наболее общие линейное преобразование объекта можно сделать с помощью матрицы преобразования.



```
\draw(0,0) rectangle (1,1) node[pos=0.5] {$1$};
\begin{scope}[cm={1,0,.5,.5,(1.5,0)}]
  \draw(0,0) rectangle (1,1) node[pos=0.5] {$2$};
\end{scope}
\begin{scope}[cm={.5,.5,0,1,(3.5,0)}]
  \draw(0,0) rectangle (1,1) node[pos=0.5] {$3$};
\end{scope}
```

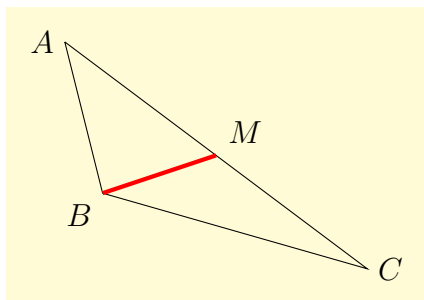
Матрицы преобразований, используемые выше:

$$M_2 = \begin{pmatrix} 1 & 0 \\ 0.5 & 0.5 \end{pmatrix} \quad M_3 = \begin{pmatrix} 0.5 & 0.5 \\ 0 & 1 \end{pmatrix},$$

а сдвиги  $\Delta_2 = (2, 0)$  и  $\Delta_3 = (4.5, 0)$ .

## 1.4 Запоминание точки

При создании рисунков бывает удобно запомнить некоторую точку, которая часто будет использоваться в дальнейшем. Это позволит при необходимости быстро и легко поменять координаты в одном месте кода, а не изменять у каждого элемента по-отдельности, в примере ниже медиана  $BM$  отрисовывается автоматически при изменении координат  $(A)$ ,  $(B)$ ,  $(C)$

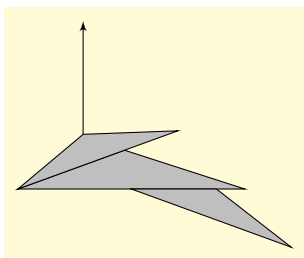


```
\coordinate (A) at (1,5);
\draw (A) node[left] {$A$} -- (1.5,3) coordinate (B) node[below left] {$B$}
-- (5, 2) coordinate (C) node[right] {$C$} -- (A);

% Медиана BM
\draw[red, ultra thick] (B) -- ($(A)!0.5!(C)$) node[above right, black] {$M$};
```

## 1.5 Повороты координат

Если значительная часть картинки должна быть сдвинута и повернута, то можно использовать окружение `scope`, и использовать аргумент `[shift={(x,y)}, rotate=z]`, где  $x, y$  – положение центра координат `scope` на холста, а  $z$  – угол в градусах, на который будет повернуты координаты внутри `scope` относительно координат `scope`.



```
\draw[fill=black!25] (0,0) -- (2,0) -- (20:1) -- cycle
;
\begin{scope}[rotate=20]
\draw[fill=black!25] (0,0) -- (1.5,0) -- (20:.75) --
cycle;
\path (20:.75) coordinate (a);
\end{scope}
\begin{scope}[shift={(1,0)}, rotate=-20]
\draw[fill=black!25] (0,0) -- (1.5,0) -- (20:.75) --
cycle;
\end{scope}

\draw[->] (a) --- (0,1);
```

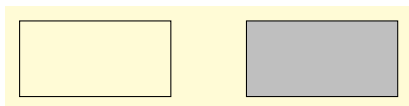
В виде  $(20:1)$  задается точка, удаленная на расстояние 1 см от начала отсчета, такая, что ее радиус-вектор составляет угол  $20^\circ$  с направлением оси  $x$  по часовой стрелке. Также корректно использовать  $++(20:1)$  – сдвиг от текущего положения на 1 см в направлении составляющем  $20^\circ$  с осью  $x$ .

## 1.6 Общий принцип

Рекомендуется использовать такой принцип работы с жирными линиями: какие-то виртуально существующие линии рисовать толщиной `ultrathin`, а физически существующие объекта линиями стандартной толщины либо `thick`. Исключением являются оси систем координат и векторы — они рисуются линиями стандартной толщины.

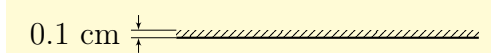
## 2 Механика

### 2.1 Подвижные тела

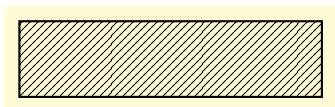


```
\draw (0,0) rectangle (2,1);  
\draw[fill=black!25] (3,0) rectangle (5,1);
```

### 2.2 Недвижимость



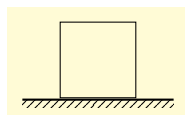
```
\draw[thick] (0,0) -- (4,0);  
\draw[ground] (0,0) rectangle (4,0.1);
```



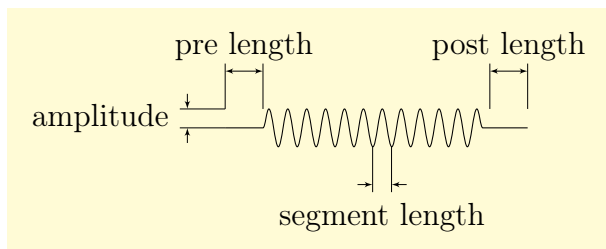
```
\draw[thick] (0,0) rectangle (4,1);  
\draw[ground] (0,0) rectangle (4,1);
```

### 2.3 Подвижное тело на неподвижности

Чтобы подчеркнуть подвижность тела, можно добавить расстояние 0.025 cm, чтобы оно «парило» над поверхностью:



### 2.4 Пружины

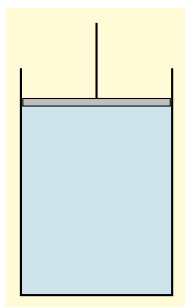


```
\draw[decorate,decoration={snake,  
pre length=0.5cm, post length=0.5cm,  
segment length=0.5cm,  
amplitude=0.25cm}]  
(0,0) -- (4,0);
```

### 2.5 Жидкости, сосуды и поршни

1. Сосуды рисуются `thick` линиями.
2. Для отрисовки воды используется аргумент `water`, которые автоматически заливает фигуру голубым полупрозрачным цветом. Также есть предопределенные цвета для других заливок: `gas`, `glass`

3. Поршни выделяются серым `black!25` цветом.
4. Зазор между поршнем и стенками сосуда `0.025 cm`



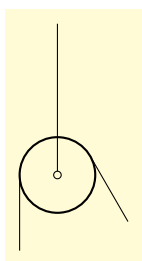
```
\fill[water] (0,-.5) rectangle ++(2,-2.5);

\draw[fill=black!25] (0.025,-0.5) rectangle ++(2-0.05,0.1);
\draw[thick] (1,-0.4) -- ++(0,1);

\draw[thick] (0,0) --++(0,-3) --++(2,0) --++(0,3);
```

## 2.6 Блоки и нити

Центр блока обозначается белым кругом радиуса `0.05 cm`. Нити рисуются линиями стандартной толщины, для блоков используется `thick`.



```
\draw[thick] (0,0) circle (0.5);
\draw (0,0) circle (0.05);

\draw (0,0.05) -- (0,2); %нить, на которой подвешен блок

\path[rotate=30] (0.5,0) coordinate (a);
\draw[shift=(a),rotate=30] (0,0) -- (0,-1); %правая нить

\draw (-0.5,0) -- ++(0,-1); %левая нить
```

## 2.7 Гирька

Если в Вашем рисунке присутствует гирька, то Вы можете воспользоваться этим паттерном:

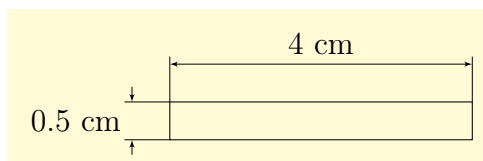


```
\draw[fill=black!25](-.2,0) -- (-.2,.3) .. controls (0,.4) and (-.1,.4)..
(-.1,.5) arc (180:0:.1) .. controls (.1,.4) and (0,.4) .. (.2,.3) --
(.2,0) -- cycle;
```

# 3 Выносные размеры

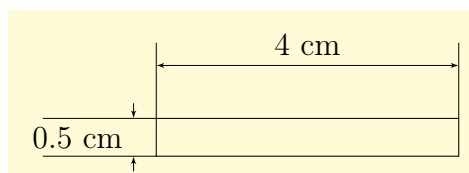
## 3.1 Линейные размеры

В стандартных случаях, как например подпись длины прямоугольника, можно использовать команды `\dist` и `\sdist`:



```
\dist{(0, 0.5)}{(4, 0.5)}{90}{above}{4 cm}

\sdist{(0,0)}{(0,0.5)}{180}{left}{0.5 cm}
```



```
\dist[1][0.7]{(0,0.5)}{(4,0.5)}{90}{above}{4cm}
}

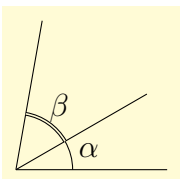
\sdist[1.5][0.3]{(0,0)}{(0,0.5)}{180}{left}
}{0.5 cm}
```

Команды `\dist` и `\sdist` имеют 5 обязательных (указываются в фигурных скобках `{}`) и 2 опциональных аргумента (указываются в квадратных скобках `[]` **перед обязательными**).

- Первый опциональный аргумент – это длина выносной линии в сантиметрах (`default = 0.6 cm`)
- Второй – расстояние от стрелок до точек, куда подносится выносной размер (`default = 0.5 cm`)
- Первые два обязательных аргумента – это точки, к которым мы хотим поднести наш выносной размер (указываются обе координаты в круглых скобках `(x, y)`)
- Третий обязательный аргумент – это угол в градусах, на который повернуты габаритные линии
- Четвертый обязательный аргумент – это расположение подписи, относительно центра линии, соединяющей стрелочки (`above, below, right, left`)
- Пятый обязательный аргумент – это текст подписи

## 3.2 Угловые размеры

Для указания угловых размеров обычно удобно использовать следующие параметры: радиус дуги 0.75 cm, расстояние до подписи 1.0 cm.



```
\draw (0,0) -- (2,0);
\draw[rotate=30] (0,0) -- (2,0);
\draw[rotate=80] (0,0) -- (2,0);

%дуга 0-30
\draw(0.75,0) arc (0:30:0.75);
\path[rotate=30/2] (1,0) node {$\alpha$};

%двойная дуга 30-80
\draw[rotate=30,double=yellow!20](0.75,0) arc (0:50:0.75);
\path[rotate=30+50/2] (1,0) node {$\beta$};
```

## 4 Оптика

### 4.1 Источники света

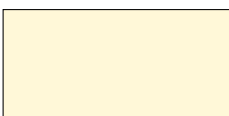
Команда `\lightsource` рисует источник света и в качестве аргумента принимает его координаты. Подписи к источникам света удаляются на 0.2 cm.



```
\lightsource{(0,0)};
\draw (0,0) node [above=0.2cm] {$S$};
```

### 4.2 Стекланные объекты

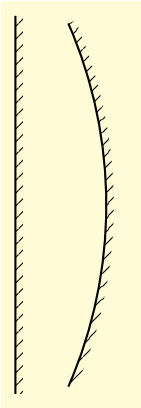
Аналогично с водой, аргумент `glass` приводит к автоматической полупрозрачной заливки объекта.



```
\draw[glass] (0,0) rectangle (3,1.5);
```

### 4.3 Зеркала и экраны

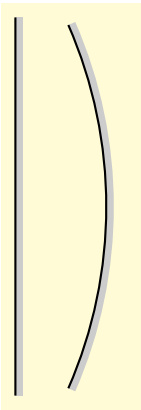
Отрисовка экранов и зеркал похожа на отрисовку недвижимости, только надо использовать аргумент `[screen]`. Линия экрана рисуется толщиной `thick`



```
\draw[thick] (0,0) -- (0, -5);
\fill[screen] (0,0) rectangle (0.1, -5);

\draw[thick] (0.7,-0.1) arc (23.6:-23.6:6);
\fill[screen] (0.7,-0.1) ++(23.6:0.1) arc(23.6:-23.6:6.1) --
  ++(-23.6:-0.1) arc (-23.6:23.6:6);
```

Чтобы на рисунках можно было визуально различать зеркала и экраны рекомендуется использовать для зеркал шаблон, представленный ниже. **Важно** делать заливку до отрисовки линии, иначе часть толщины линии будет перекрыта

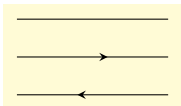


```
\fill[mirror] (0,0) rectangle (0.1, -5);
\draw[thick] (0,0) -- (0, -5);

\fill[mirror] (0.7,-0.1) ++(23.6:0.1) arc(23.6:-23.6:6.1) --
  ++(-23.6:-0.1) arc (-23.6:23.6:6);
\draw[thick] (0.7,-0.1) arc (23.6:-23.6:6);
```

### 4.4 Лучи

Лучи рисуются линиями стандартной толщины.



```
\draw (0,0) -- ++ (2,0)
\draw[directed] (0,-0.5) -- ++ (2,0)
\draw[reverse directed] (0,-1) -- ++ (2,0)
```

## 5 Электрические схемы

Для отрисовки электрических схем используется пакет `circuitikz`, у которого есть понятная [документация](#)

Рекомендуется использовать аргументы `american resistors` и `american inductors` у всего рисунка

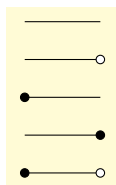
```
% in preamble:
\usepackage{circuitikz}
% in document:
\begin{tikzpicture}[american resistors, american inductors]
\end{tikzpicture}
```

Или можно задать это для всего документа в преамбуле:

```
% in preamble:
\usepackage{circuitikz}
\ctikzset{resistor = european, inductor = american}
```

## 5.1 Общая логика, провода

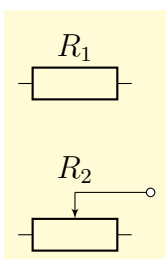
Общая логика пакета `circuitikz` строится вокруг декорирования пути между двумя точками. Для этого используются дополнительные аргументы у команды `to[]`. Для работы с проводами используется аргумент `short`.



```
\draw (0,0) to[short,-] ++(1,0);
\draw (0,-1) to[short,-o] ++(1,0);
\draw (0,-2) to[short,*-] ++(1,0);
\draw (0,-3) to[short,-*] ++(1,0);
\draw (0,-4) to[short,*-o] ++(1,0);
```

## 5.2 Резисторы

Резистору соответствует аргумент `R`. Потенциометру соответствует аргумент `pR`.



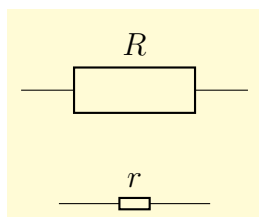
```
\draw (0,0)
to[R=$R_1$] ++ (1.5,0);

\draw (0,-1.56)
to[pR=$R_2$,name=pR] ++ (1.5,0);

\draw(pR.wiper) to[short,-o] ++(1,0);
```

Для работы с третьим выходом потенциометра даем уникальное название `name` потенциометру, а далее обращаемся к этой координате, как к `name.wiper` (в примере выше это `pR.wiper`).

Параметр `/tikz/circuitikz/bipoles/length=` изменит размер элементов с двумя выводами

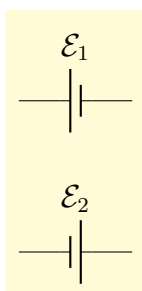


```
\draw (0,0)
to [R=$R$, /tikz/circuitikz/bipoles/length=2cm] ++(3,0);

\draw (0.5,-1.5)
to [R=$r$, /tikz/circuitikz/bipoles/length=0.5cm] ++(2,0);
```

## 5.3 Источник напряжения и источник тока

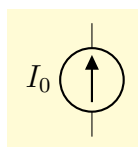
Источнику постоянного напряжения соответствует аргумент `battery1`.



```
\draw (0,0)
to[battery1,l=$\mathcal{E}_1$] ++ (1.5,0);

\draw (0,-2)
to[battery1,l=$\mathcal{E}_2$,invert] ++ (1.5,0);
```

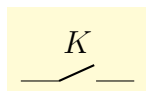
Чтобы развернуть батарею используйте аргумент `invert`. Наименования батареек следует добавлять с помощью дополнительного аргумента `l=label`.



```
\draw (0,0) to[american current source, l=$I_0$] (0,1.5);
% or
\draw (0,0) to[american, isource, l=$I_0$] (0,1.5);
```

## 5.4 Ключи

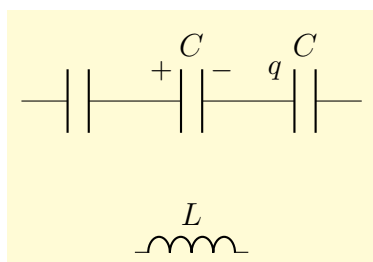
Обычному ключу соответствует аргумент `pos`.



```
\draw (0,0)
to[nos=$K$] ++ (1.5,0);
```

## 5.5 Конденсаторы, катушки

Конденсатору соответствует аргумент `C`. Катушке соответствует аргумент `L`.



```
\draw (-2.25,0)
to[C] ++ (1.5,0)
to[C=$C$] ++ (1.5,0)
to[C=$C$] ++ (1.5,0);

\draw(0,0) ++ (-.4,.4) node {\small $+$};
\draw(0,0) ++ (.4,.4) node {\small $-$};

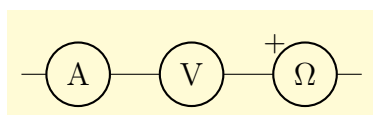
\draw(1.5,0) ++ (-.4,.4) node {\small $q$};

\draw (-.75,-2)
to[L=$L$] ++ (1.5,0);
```

Иногда в задаче нужно указать знаки либо величины зарядов на пластинах конденсатора. В таком случае используются удаленные от центра конденсатора на 0.4 cm метки, сделанные шрифтом `small`.

## 5.6 Вольтметры, амперметры, омметры

Измерительные приборы создаются с помощью аргумента `rmeter`

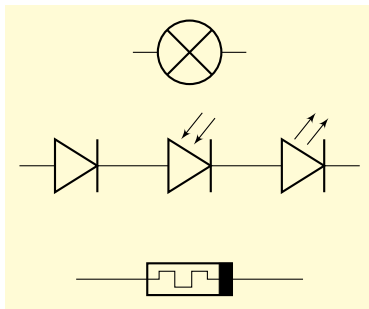


```
\draw (0,0)
to[rmeter,t=$A\,$] (1.5,0)
to[rmeter,t=$V$] (3,0)
to[rmeter,t=$\Omega$] (4.5,0);

\draw(3.75,0) ++ (-.4,.4) node {\small $+$};
```



## 5.7 Другие элеткрические элементы

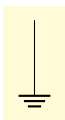


```
\draw (1.5,0)
to[lamp] ++(1.5,0);

\draw (0,-1.5)
to[diode] ++(1.5,0)
to[photodiode] ++(1.5,0)
to[led]++(1.5,0);

\draw (0.75,-3)
to[memristor] ++(3,0);
```

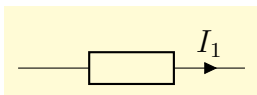
## 5.8 Заземление



```
\draw (0,0)
to (0,-1) node[tlground]{};
```

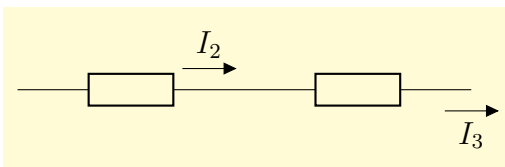
## 5.9 Сила тока в цепях

По умолчанию сила тока в пакете `circuittikz` обозначаются и выглядят так:



```
\draw (0,0) to[R, i^>=$I_1$] (3,0);
```

Такой вариант не рекомендуется использовать. Для привычного обозначения можно использовать `flow` (потоки) из этого пакета:



```
\draw (0,0)
to[R, f^>=$I_2$] ++(3,0)
to[R, f_>=$I_3$, flow/distance=1] ++(3,0);
```

Аргумент `flow/distance=` меняет расположение стрелочки вдоль провода. Настроить расположение стрелки относительно провода(сверху/снизу) и направление можно используя разные символы после `f`: `f=`, `f<=`, `f_`, `f_>=`, `f<^=`, `f<_`, `f>_`

## 6 Лайфхаки

### 6.1 Вертикальные подписи для электрических схем

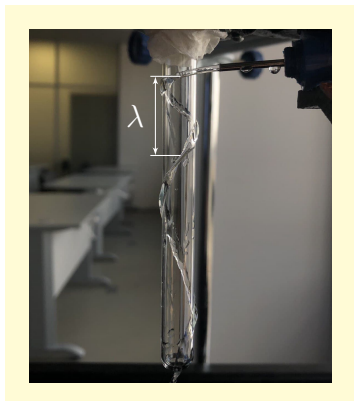
1. Чтобы подписи для повернутых элементов были тоже вертикальными пользуйтесь командой

```
\ctikzset{label/align = straight}
```

в самом начале кода рисунка.

## 6.2 Вставка растрового рисунка внутрь tikz

Если вам, например, понадобилось нарисовать обозначения поверх фотографии, то вы можете вставить изображение внутрь `tikzpicture` с помощью команды `\includegraphics`, как текст `node` выбранная точка будет центром вставленного изображения.



```
\begin{tikzpicture}[example,>=latex',scale=4.0/6.0]
\path (0,0) node {\includegraphics[width=4cm]{pic.jpg}};

\begin{scope}[draw=white]
\dist{(0,2.57)}{(0,1.0)}{180}{left}{}
\path(0,{(2.57+1.0)/2}) ++ (180:.5) node [left,
white] {$\lambda$};
\end{scope}

\end{tikzpicture}
```