

УДК 004.021, 378

**О ПРИМЕНЕНИИ ЯЗЫКА ВЕКТОРНОЙ ГРАФИКИ
ASYMPTOTE ДЛЯ ИЛЛЮСТРАЦИИ
УЧЕБНО-МЕТОДИЧЕСКИХ И НАУЧНЫХ РАБОТ
ФИЗИКО-МАТЕМАТИЧЕСКОЙ НАПРАВЛЕННОСТИ**

П. А. Макаров

Исследована возможность применения языка векторной графики Asymptote для иллюстрации физико-математических учебно-методических и научных работ. Разработан ряд изображений, иллюстрирующих решение задач из разных областей физики и математики. Показано, что язык Asymptote обладает удобным высокоуровневым синтаксисом и достаточно разработанной объектно-ориентированной архитектурой.

Ключевые слова: векторная графика, Asymptote, язык программирования высокого уровня.

1. Введение

Набор макрорасширений L^AT_EX 2_ε [1–4] системы компьютерной вёрстки T_EX [5] фактически стал общепринятым типографским стандартом при подготовке к публикации научных и учебно-методических работ в области физико-математических наук. Так, пакет L^AT_EX 2_ε оказывается чрезвычайно многофункциональным и удобным инструментом при выполнении широкого спектра задач, включающих:

- 1) подготовку отчётов о выполненной работе;
- 2) публикацию методических материалов и учебных пособий;
- 3) написание курсовых, дипломных проектов и научных статей;
- 4) оформление стендовых докладов и презентаций.

Круг проблем, решаемых с помощью системы L^AT_EX 2_ε, не ограничивается приведённым набором. Этот список представляет собой лишь перечисление некоторых классов наиболее актуальных задач.

Опыт свидетельствует о принципиальной важности подготовки качественных графических материалов, иллюстрирующих и дополняющих содержание текста. В зависимости от направленности работы это могут быть и принципиальные электрические схемы, и блок-схемы экспериментальных установок, и чертежи, поясняющие суть физических процессов и явлений, а также графики, иллюстрирующие функциональные зависимости величин.

Для решения столь разнообразных графических задач какого-либо одного общепринятого стандарта нет. Более того, существует точка зрения, что для разных типов графических работ следует использовать только специализированные инструменты. Нисколько не оспаривая эту идею, покажем, что в настоящее время имеется средство, которое можно считать достаточно универсальным для большого набора разнообразных графических работ физико-математической направленности.

В настоящее время для решения обозначенных выше задач существуют программы инструментов различных типов, однако, на наш взгляд, наиболее удобным является язык высокого уровня Asymptote [6]. Язык векторной графики и интерпретатор Asymptote изначально разрабатывались для создания PostScript-графики и её использования в $\text{\LaTeX} 2_{\epsilon}$ документах, однако на настоящий момент с его помощью можно создавать рисунки и других графических форматов, а также использовать независимо от $\text{\LaTeX} 2_{\epsilon}$ [7].

Таким образом, цель данной работы — исследование возможности применения языка векторной графики Asymptote для иллюстрации физико-математических учебно-методических и научных работ.

2. Краткий обзор средств векторной графики

На текущий момент существует множество различных типов графических файлов, однако всё это многообразие можно разбить на две группы форматов:

- 1) растровые;
- 2) векторные.

Преимущества и недостатки обеих групп форматов хорошо известны и подробно отражены в литературе [2; 3; 8–12]. В связи с этим не будем проводить подробное сравнение растровых и векторных форматов. Заметим только, что принципиально невозможно выбрать универсальный тип графики, подходящий для всех случаев. Поэтому различные и растровые (*.jpg, *.png, *.tiff, *.bmp и многие другие), и векторные (*.ps, *.eps, *.svg, *.pdf и т. п.) типы файлов находят своё применение в различных прикладных задачах. В рамках данной работы мы сосредото-

точимся на создании именно векторной графики.

Рассмотрим основные средства с помощью которых можно проиллюстрировать \LaTeX -документы высококачественными векторными изображениями.

2.1. PostScript

PostScript — это язык описания страниц, используемый в издательско-типографских системах. Программные или аппаратные интерпретаторы PostScript для печати документов в настоящее время присутствуют практически на всех компьютерных системах [12].

Фактически, PostScript — это полнофункциональный, полный по Тьюрингу, интерпретируемый, стековый язык программирования. Синтаксис языка использует обратную польскую нотацию, что делает ненужным использование скобок. Большинство операторов берут операнды из стека и помещают результат вычислений в стек [12; 13].

На русском языке очень мало книг о PostScript-графике и её использовании в \LaTeX -документах. В качестве таковых можно отметить перевод [9] классической книги [8], а в остальном приходится использовать оригинальные тексты технической документации [12; 14] и некоторые книги [13; 15].

2.2. PGF и TikZ

Изначально средства для создания рисунков в \LaTeX -документах были довольно ограничены, но усилиями многих людей количество таких инструментов существенно увеличилось. Благодаря идее интегрировать \LaTeX и PostScript были созданы такие графические пакеты, как, например, хурис, PSTricks, PGF/TikZ, программа METAPOST и некоторые другие [16].

Пакет PGF (Portable Graphics Format — переносимый графический формат) является инструментом для создания встраиваемой в \TeX -документ графики [10; 11].

При использовании PGF графика программируется так же, как и сам \TeX -документ. Преимуществами данного подхода являются: быстрое создание простых графиков, точное позиционирование, использование макроопределений, очень высокое качество печати. Однако наследуются и все недостатки \TeX 'а: код не дает представления о конечном результате, необходимо тратить время на изучение используемых пакетов, отсутствует наглядность, требуется полная повторная компиляция даже при минимальных изменениях [10].

Работа с пакетом PGF может происходить на нескольких уровнях абстракции (см. подробнее руководство [11]):

- Системный уровень. Обеспечивает полную абстракцию того, что передается драйверу. Драйвер — программа, аналогичная `dvips` или `dvipdfm`, которая принимает `*.dvi` файл и производит `*.ps` или `*.pdf` файл.

- Основной уровень. Состоит из ядра, которое содержит несколько взаимозависимых пакетов, загружаемых в одном блоке, и дополнительных модулей, расширяющих ядро большим числом команд специального назначения.

- Уровень внешнего интерфейса.

Существует несколько разных внешних интерфейсов для PGF:

- `TikZ` — позволяет обращаться ко всем особенностям PGF. Синтаксис `TikZ` — смесь синтаксиса программ `METAFONT`, `PSTricks` и некоторых идей Т. Тантау.

- `pgfpict2e` — переопределяет стандартное окружение `picture` и его команды, используя основной уровень PGF.

Пользователю PGF, основная задача которого — создание необходимых рисунков в \TeX -документе, в основном достаточно команд внешнего интерфейса.

2.3. METAPOST

`METAPOST` — это язык программирования, основанный на языке Дональда Кнута `METAFONT`, производящий PostScript-программы. Заимствования из `METAFONT` включают базовые средства для создания и манипулирования картинками, в частности типы чисел, координатные пары, кубические сплайны, аффинные трансформации, текстовые строки и булевы величины. Дополнительные средства делают возможными соединение текста и графики, а также доступ к специальным возможностям PostScript, таким как вырезка, затенение, пунктирные линии [17].

Ещё одним важным свойством, заимствованным у `METAFONT`, является способность решать заданные неявно линейные уравнения, что позволяет писать многие программы в значительной мере в декларативном стиле.

Язык `METAPOST` особенно хорошо приспособлен для генерации картинок для технических документов, где некоторые свойства рисунка могут контролироваться математическими или геометрическими ограничениями, которые наилучшим образом выражаются в символьной форме. Другими словами, `METAPOST` не занимает место средств для ручного рисования или даже интерактивных графических редакторов. Это настоящий язык программирования для генерации графики и особенно иллюстраций для \TeX -документов [17].

На русском языке `METAPOST` достаточно подробно описан в переводе документации [17] и серии журнальных статей [18]. Кроме того, полезным может оказаться изучение языка `METAFONT` по книге Дональда Кнута [19].

2.4. *Asymptote*

Язык векторной графики *Asymptote* создан А. Хаммерлиндлом, Д. Боуменом и Т. Принсом в 2004 году под идейным влиянием программы `METAPOST`. *Asymptote* фактически представляет собой язык программирования высокого уровня, являющийся транслятором к языку `PostScript` и, по задумке авторов, должен обеспечивать некий «стандарт» для вёрстки математических иллюстраций так же, как `TeX` является стандартом при вёрстке сложных формул и уравнений [16]. *Asymptote* может использоваться для создания анимации, трёхмерной графики, диаграмм Фейнмана, бинарных деревьев и многого другого.

Asymptote ориентирован на пользователей, владеющих математикой, так как в нём используются, например, аффинные преобразования, комплексные переменные и т. д. Его будет легче осваивать людям, имеющим хотя бы минимальный опыт программирования и использования `LaTeX`, поскольку текстовые метки, формулы и уравнения могут набираться с его помощью — это обеспечивает соответствующее качество рисунков [16]. Однако язык можно использовать отдельно от системы `LaTeX`. По умолчанию *Asymptote* производит выходные файлы формата `eps`, но существуют и другие возможности, например `pdf`, `png` и т. д.

Asymptote построен по модульному принципу. Основные возможности языка содержатся в модуле `plain`. Полезными на практике могут быть также модули `geometry`, `graph` и `unicode`.

Язык *Asymptote* может использоваться по-разному:

- 1) в режиме непосредственного исполнения команд интерпретатором;
- 2) в скриптовом режиме при обработке предварительно написанных исходных файлов;
- 3) в качестве окружения `LaTeX`-документов.

Первый вариант удобен только для предварительной оценки возможностей языка, в реальной работе применять его нет смысла. Некоторые авторы [20] отмечают, что третий способ (встраивание программы на *Asymptote* в `LaTeX`'овский текст) также неэффективен, так как требует трех проходов компиляции исходного текста документа. Таким образом, наиболее эффективным методом является второй вариант, в котором предполагается отдельная компиляция рисунков и последующая вставка их в текст.

Основной проблемой проекта *Asymptote* является отсутствие достаточно полной и доступной документации. Это, конечно, не является непреодолимым препятствием, так как имеется свободно доступное официальное руководство [6], но оно довольно обзорно и вряд ли может рассматриваться как полная документация к языку. Ситуация несколько скрашивается тем, что в сети Интернет имеется некоторое количество сайтов с галереями примеров использования *Asymptote*. Однако и здесь имеется тонкость: дело в том, что большая часть этих примеров прокомментирована не на английском, а на французском, чешском и некоторых других языках.

В то же время в части документации есть и положительные моменты. Так, имеется перевод на английский [21] вполне полного руководства по очень полезному модулю `geometry.asy`. В 2015 году вышел перевод этого источника и на русский язык [7]. Также за последние три года появились относительно оригинальные русскоязычные пособия Крячкова [16] и Волченко [20].

2.5. Выводы

Таким образом, проведённый обзор литературы показывает, что в настоящее время существует достаточно большое количество качественных инструментов для создания векторной графики в \LaTeX -документах.

Вместе с тем *Asymptote* использует наиболее высокий уровень абстракций из всех рассмотренных языков, а его синтаксис максимально приближен к синтаксису языков общего назначения типа C, C++, Java. *Asymptote* обладает модульной структурой и ориентирован на создание действительно сложной технической графики. Всё это позволяет рассматривать язык *Asymptote* как универсальный инструмент в ежедневной работе физика и математика.

3. Примеры применения *Asymptote*

Рассмотрим несколько примеров практического использования языка *Asymptote* для создания рисунков, иллюстрирующих задачи из различных областей физики и математики.

3.1. Логарифмическая спираль

Достаточно часто при решении задач о плоском движении удобным оказывается использовать полярную систему координат.

Постановка задачи

Пусть точка участвует одновременно в двух взаимно перпендикулярных затухающих колебаниях:

$$\begin{aligned}x(t) &= Ae^{-ht} \cos(kt + \varepsilon), \\y(t) &= Ae^{-ht} \sin(kt + \varepsilon),\end{aligned}\tag{1}$$

где $A, h, k > 0$ и ε — некоторые вещественные константы.

Найти уравнение её движения в полярных координатах и определить траекторию.

Решение

Очевидно, что параметрическое уравнение траектории (1) в полярных координатах принимает следующий вид:

$$r(\varphi) = A \exp\left[\frac{h}{k}(\varepsilon - \varphi)\right].\tag{2}$$

Решение (2) — это уравнение логарифмической спирали, встречающейся ещё в работах Декарта и Бернулли. Пример траектории (2), построенный для значений $A = 5$ м, $h = 0.1$ с⁻¹, $\varepsilon = \pi/3$, $k = \pi/4$ с⁻¹, приведён на рис. 1.

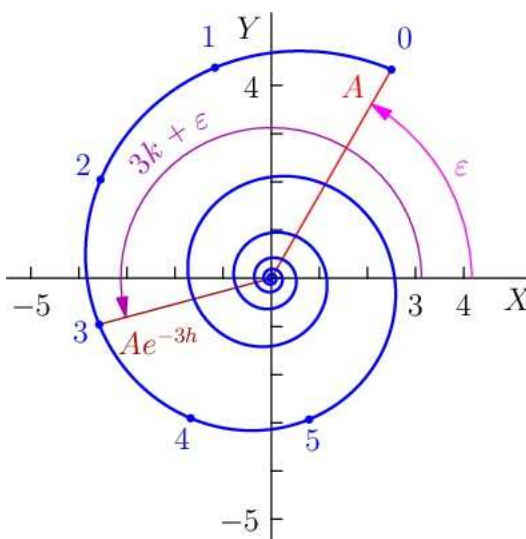


Рис. 1. Логарифмическая спираль

Рис. 1 наглядно демонстрирует физический смысл параметров задачи A , h , k и ε . В частности, величина A — есть начальное расстояние от точки до начала координат O ; ε — это начальный угол, отсчитываемый от оси OX ; значение $1/h$ определяет время, через которое расстояние между точкой и началом координат уменьшается в e раз; k — есть угловая скорость вращения точки. Кроме того, на рис. 1 указаны положения точки и соответствующие моменты времени за первые пять секунд движения.

Для построения рис. 1 с помощью *Asymptote* в первую очередь требуется подключить все необходимые для этого модули и выполнить основные настройки. Соответствующий исходный текст приведён в листинге 1.

Листинг 1. Подключаемые модули и настройка параметров

```

1 import settings;
2 import geometry;
3 import graph;
4 import labelpath;
5 size(7cm, 0);
6 unitsize(3cm);
7 settings.tex = "latex";
8 settings.outformat = "pdf";
9 usepackage("mathabx");
10 ngraph = 1000;
```

После этого необходимо определить и проинициализировать переменные, а также описать используемые функции. Соответствующий фрагмент текста программы приведён в листинге 2.

Листинг 2. Определение переменных и функций

```

1 real A = 5;
2 real epsilon = pi/3;
3 real k = pi/4;
4 real h = 1/10;
5 real r(real t) { return A*exp(-h*t); }
6 real phi(real t) { return k*t + epsilon; }
7 real x(real t) { return r(t)*cos(phi(t)); }
8 real y(real t) { return r(t)*sin(phi(t)); }
```

На следующем этапе подготовки иллюстрации изобразим координатную систему и непосредственно кривую (2). Также отметим точками на кривой положения материальной точки за первые пять секунд движения. Все эти действия описываются фрагментом текста листинга 3.

Листинг 3. Изображение системы координат и кривой

```

1 coordsys Osys = cartesianssystem((0,0),i=(1,0),j=(0,1));
2 xaxis("$X$", xmin=-1.1A, xmax=1.1A);
3 yaxis("$Y$", ymin=-1.1A, ymax=1.1A);
4 for(int i=-5; i < 5; ++i) {
5   xtick(i, size=4);
6   ytick(i, size=4);
7 }
8 labelx(-5); labelx(3); labelx(4);
9 labely(-5); labely(4);
10 draw(graph(x, y, 0, 45), blue+linewidth(1));
11 for(int i=0; i <= 5; ++i) {
12   dot((x(i),y(i)), blue);
13   label("$"+string(i)+"$", (1.105x(i),1.15y(i)), blue);
14 }

```

Завершая подготовку рисунка, изобразим дополнительные метки, наглядно поясняющие физический смысл величин A , ε , k и h . Данный фрагмент исходного текста составляет листинг 4.

Листинг 4. Отображение дополнительных меток

```

1 draw(origin(Osys)--(x(0),y(0)),red);
2 label("$A$", C2, NW, red);
3 point C0 = (A/1.2,0);
4 point C1 = (A/1.2*cos(epsilon/2), A/1.2*sin(epsilon/2));
5 point C2 = (A/1.2*cos(epsilon), A/1.2*sin(epsilon));
6 draw(C0..C1..C2, magenta, Arrow);
7 label("$\varepsilon$", 1.1C1, magenta);
8 draw(origin(Osys)--(x(3),y(3)),0.7red);
9 label("$Ae^{-3h}$", (x(3),y(3)), S+2E, 0.7red);
10 point D0 = (A/1.6,0);
11 point D1 = (A/1.6*cos(phi(3)/2), A/1.6*sin(phi(3)/2));
12 point D2 = (A/1.6*cos(phi(3)), A/1.6*sin(phi(3)));
13 draw(D0..D1..D2, 0.7magenta, Arrow);
14 point D = relpoint(D2..D1..D0, 0.7);
15 labelpath(L="$3k+\varepsilon$", 1.1D2..1.1D1..1.1D,
    Centered, 0.7magenta);

```

Подводя итоги, можно сказать, что средства *Asymptote* в целом соответствуют процедурной парадигме, существенно дополненной элементами объектно-ориентированного программирования. Это оказывается достаточно удобным, так как позволяет писать программы на разных уровнях абстракции.

3.2. Теорема о сложении скоростей

В качестве следующей иллюстрации применения языка Asymptote рассмотрим следующую задачу.

Постановка задачи

Моторная лодка A переплывает реку шириной ℓ из точки B с постоянной по величине скоростью v (относительно воды), всё время направленной в точку O . Считая лодку материальной точкой, найти уравнение её траектории в системе OXY , связанной с берегом реки (см. рис. 2), если скорость течения воды в реке $u = \text{const}$.

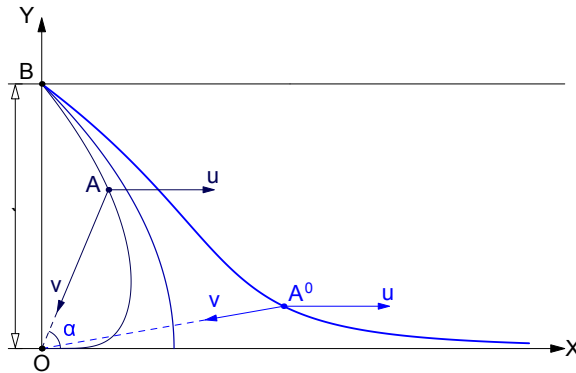


Рис. 2. Иллюстрация к задаче о моторной лодке

Решение

Не будем останавливаться здесь на подробном решении задачи, которое можно найти, например, в § 4 второй главы пособия [22]. Приведём лишь окончательный результат, который имеет вид уравнения траектории, заданной в неявном виде:

$$x(y) = \frac{y^{1-\beta}}{2\ell^\beta} (\ell^{2\beta} - y^{2\beta}), \quad (3)$$

где параметр $\beta = u/v$ есть отношение скоростей реки и лодки.

Три кривые из семейства траекторий (3) для значений параметра $\beta \in \{1.25, 1, 0.75\}$ уже приведены на рис. 2. При построении кривой, соответствующей значению $\beta = 1.25$, было учтено, что $\lim_{y \rightarrow 0} x = \infty$.

Аналогично примеру 3.1 приведём текст программы, создающей рис. 2, разбив его на несколько отдельных листингов. В частности, подключение необходимых модулей и настройка основных параметров производится в фрагменте программы согласно листингу 5.

Листинг 5. Подключаемые модули и настройка параметров

```
1 import graph; import geometry;
2 settings.tex="pdflatex"; settings.outformat="pdf";
3 unitsize(1mm);
4 texpreamble("\usepackage{amsmath}");
```

Определение и инициализация переменных, а также описание функции приведено в листинге 6.

Листинг 6. Определение переменных и функций

```
1 real l = 50;
2 real[] v = {25,20,15};
3 real u = 20;
4 real b;
5 pen[] p;
6 pen vpen = linewidth(0.4);
7 pair f(real y) {
8   return (y^(1-b)*(l^(2b) - y^(2b))/2/l^b, y);
9 }
```

Три траектории движения лодки, координатная система, а также основные обозначения на чертеже изображаются с помощью фрагмента программы, показанного в листинге 7.

Листинг 7. Изображение трёх кривых и системы координат

```
1 for(int i = 0; i < v.length; ++i) {
2   b = u/v[i];
3   p.push((i+1)/v.length*blue);
4   real finish = 0;
5   if (b>1) finish = 1;
6   path trajectory = graph(f, finish, l, n=1000);
7   draw(trajectory, p[i]+linewidth((i+1)/v.length));
8 }
9 xaxis(Label("$X$", EndPoint, 2E), Arrow);
10 yaxis(Label("$Y$", EndPoint, NW), 0, 1.25*l, Arrow);
11 point B = (0, 1); dot(B); label("$B$", B, NW);
12 point O = (0, 0); dot(O); label("$O$", O, S);
13 coordsys Osys = cartesianssystem(0, i=(1,0), j=(0,1));
14 line river = parallel(B, (1,0)); draw(river);
```

```
15 distance(rotate(-90)*"$\ell$", 0, B, -5mm);
```

Для большей информативности рисунка изобразим векторы скорости реки и лодки в разных положениях A и A' . Текст программы, необходимый для этих целей, содержится в листинге 8.

Листинг 8. Отображение дополнительной информации

```
1 b = u/v[0];
2 point A=f(30); dot(A,p[0]); label("$A$", A, WNW, p[0]);
3 coordsys Asys = cartesianssystem(A, i=(1,0), j=(0,1));
4 line A0 = line(A,0);
5 real a = angle(A0);
6 vector U = point(Asys, (u,0));
7 show(Label("$\boldsymbol{u}$", align = N, EndPoint), U,
      p[0]+vpen);
8 vector V = point(Asys, (v[0]*cos(a),v[0]*sin(a)));
9 show(Label("$\boldsymbol{v}$", align = 3N, EndPoint), V,
      p[0]+vpen);
10 draw(0--point(V), p[0]+vpen+dashed);
11 markangle("$\alpha$", Ox(0sys), A0, radius=10, p[0]);
12 b = u/v[2];
13 point C=f(8); dot(C,p[2]); label("$A'$", C, NE, p[2]);
14 coordsys Csys = cartesianssystem(C, i=(1,0), j=(0,1));
15 line C0 = line(C,0);
16 real b = angle(C0);
17 U = point(Csys, (u,0));
18 show(Label("$\boldsymbol{u}$", align = N, EndPoint), U,
      p[2]+vpen);
19 V = point(Csys, (v[2]*cos(b),v[2]*sin(b)));
20 show(Label("$\boldsymbol{v}$", align=2N+E, EndPoint), V,
      p[2]+vpen);
21 draw(0--point(V), p[2]+vpen+dashed);
```

Как видно из данного примера, комбинация средств модулей `graph.asy` и `geometry.asy` оказывается очень удобной для изображений кривых в разных системах координат.

3.3. Блок-схема экспериментальной установки

В качестве третьего примера использования языка `Asymptote` на рис. 3 приведена упрощенная блок-схема экспериментальной установки, основанной на ЭПР-спектрометре РЭ 1306.

Не будем в данном случае приводить полный исходный текст программы в силу его объективной очевидности и громоздкости. Рассмотр-

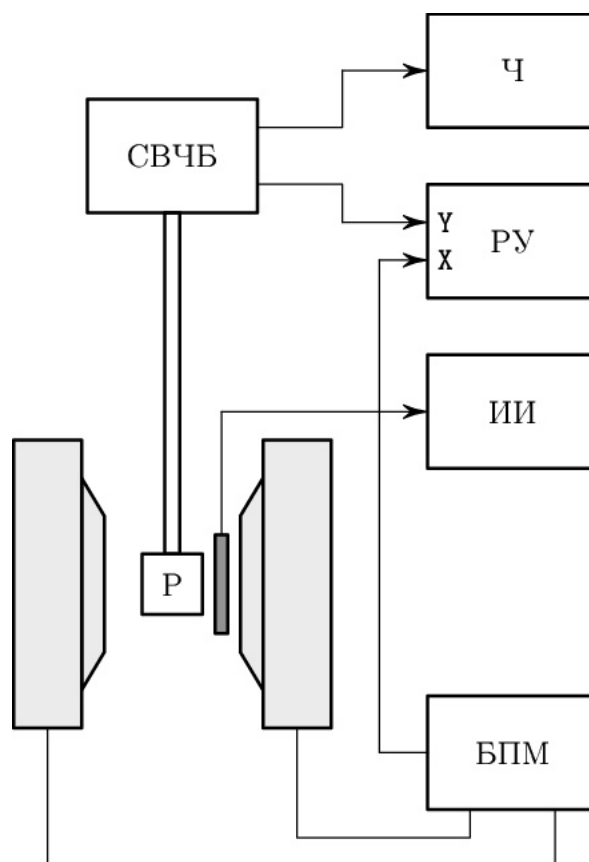


Рис. 3. Блок-схема установки на основе радиоспектрометра

рим только фрагмент кода, приведённый в листинге 9, с описанием объекта СВЧБ.

Листинг 9. Описание блока СВЧБ

```

1 point SVCH_LB = (0, 0);
2 point SVCH_LT = SVCH_LB + (0, 2w);
3 point SVCH_RB = SVCH_LB + (3w, 0);
4 point SVCH_RT = SVCH_RB + (0, 2w);
5 point SVCH_MM = ((SVCH_LB + SVCH_RT)/2);
6 path SVCH = SVCH_LB--SVCH_LT--SVCH_RT--SVCH_RB--cycle;
7 draw(SVCH, linewidth(1.1)); label("СВЧБ", SVCH_MM);

```

В приведённом фрагменте переменная w — это объявленная ранее единица длины. Сам блок определяется четырьмя точками — вершинами прямоугольника, и пятой точкой — центром фигуры. Точкой привязки является левый нижний угол фигуры $SVCH_LB$, координаты осталь-

ных точек вычисляются относительно неё.

3.4. Схема простейшего транзисторного усилителя

Специалистам в области радиофизики и электроники часто необходимо изображать принципиальные электрические схемы различной сложности. Поэтому в качестве последнего примера, иллюстрирующего возможности языка Asymptote, рассмотрим схему простейшего транзисторного усилителя, приведённую на рис. 4.

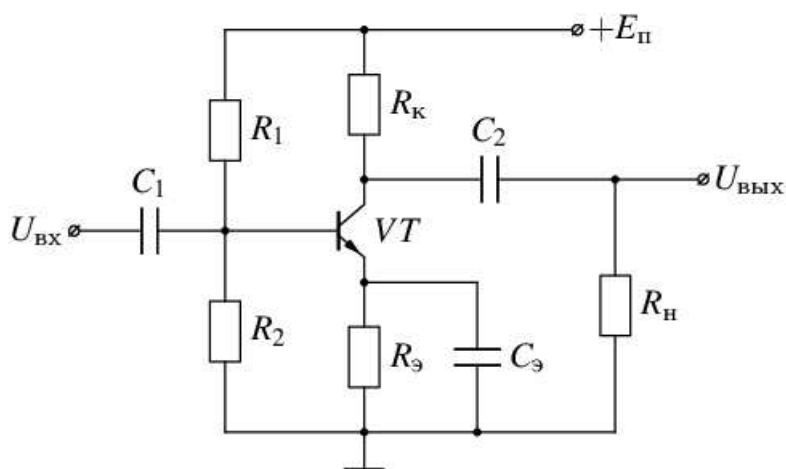


Рис. 4. Схема усилителя на биполярном транзисторе с общим эмиттером

В данном случае также не будем описывать исходный текст программы, формирующей рис. 4. Заметим только, что сам язык Asymptote не предоставляет средств для изображения принципиальных электрических схем, однако архитектура языка очень удобна для разработки модулей, описывающих новые классы объектов и методы работы с ними.

Указанное обстоятельство хорошо известно сообществу программистов Asymptote, поэтому в настоящее время существует модуль `asy-circ.asy`, написанный Кристофом Кассо для разработки компонентов электрических принципиальных схем. Сам модуль и документация к нему свободно распространяются в виде архива с исходными текстами `asy-circ1.0.zip` и доступны на официальном сайте проекта Asymptote.

Принцип работы схемы, изображённой на рис. 4, прекрасно описан в учебной литературе по электронике [23; 24], поэтому на этом также останавливаться не будем.

4. Выводы

Таким образом, в данной работе была исследована возможность применения языка векторной графики *Asymptote* для иллюстрации физико-математических учебно-методических и научных работ. Для этого выполнен обзор существующих технологий векторной графики, используемых в образовательной и научной сферах деятельности. Разработан ряд изображений, иллюстрирующих решение задач из разных областей физики и математики.

Показано, что язык *Asymptote* обладает удобным высокоуровневым синтаксисом, сходным со многими универсальными языками программирования, а также достаточно разработанной объектно-ориентированной архитектурой. Эти достоинства языка оказываются очень удобными при разработке особенно сложных научных иллюстраций. Высокое качество получаемых изображений определяется сочетанием математически точной формализации задачи и возможностью гибкой настройки формируемых изображений. Поддерживается множество векторных и растровых форматов выходных файлов. Имеется возможность интегрирования языка с системой \LaTeX , но не исключено и независимое использование языка.

Список литературы

1. **Lamport L.** \LaTeX : a document preparation system. 2 ed., Addison-Wesley, 1994. 291 p.
2. **Львовский С. М.** Набор и верстка в системе \LaTeX . 3-е изд., испр. и доп. М.: МЦНМО, 2003. 448 с.
3. **Котельников И. А., Чеботаев П. З.** $\text{\LaTeX} 2_{\epsilon}$ по-русски. 3-е изд., перераб. и доп. Новосибирск: Сибирский хронограф, 2004. 496 с.
4. **Знаменская О. В., Знаменский С. В., Лейнартас Д. Е., Трутнев В. М.** Математическая типография : курс лекций. Красноярск: Изд-во СФУ, 2008. 421 с.
5. **Кнут Д. Э.** Все про \TeX . М.: Изд. дом «Вильямс», 2003. 560 с.
6. **Hammerlindl A., Bowman J., Prince T.** *Asymptote: the Vector Graphics Language*, 2016. 189 p.

7. **Крячков Ю. Г.** Евклидова геометрия на языке векторной графики ASYMPTOTE. Волгоград: Изд-во ВГСПУ. 2015. 88 с.
8. **Goossens M., Raatz S., Mittelbach F.** The L^AT_EX graphics companion: illustrating documents with T_EX and PostScript. Addison-Wesley, 1997. 299 p.
9. **Гуссенс М., Ратц С., Миттельбах Ф.** Путеводитель по пакету L^AT_EX и его графическим расширениям. Иллюстрирование документов при помощи T_EX'a и PostScript'a. М.: Мир: Бином ЛЗ, 2002. 621 с.
10. **Кирютенко Ю. А.** TikZ& PGF. Создание графики в L^AT_EX 2_ε-документах. Ростов н/Д, 2014. 277 с.
11. **Tantau T.** The TikZand PGF Packages. Manual for version 2.10. Institut für Theoretische Informatik Universität zu Lübeck, 2010. 880 p.
12. **Taft E., Chernicoff S., Rose C.** PostScript language reference manual. 3 ed. Adobe Systems Incorporated, 1999. 912 p.
13. **Reid G. C.** Thinking in PostScript. Addison-Wesley Publishing Company, 1990. 239 p.
14. PostScript language. Tutorial and cookbook. Addison-Wesley Publishing Company, 1985. 247 p.
15. **Casselmann B.** Mathematical illustrations: a manual of geometry and PostScript. Cambridge University Press, 2004. 264 p.
16. **Крячков Ю. Г.** Асимптота для начинающих. Создание рисунков на языке векторной графики Asymptote. Волгоград: Изд-во ВГСПУ, 2015. 131 с.
17. **Хобби Дж.** МЕТАPOST. Руководство пользователя / перевод с англ. В. Лидского. 2008. 106 с. URL: <http://mirrors.ibiblio.org/CTAN/info/metapost/doc/russian/npman-ru/npman-ru.pdf> (дата обращения: 20.12.2018).
18. **Балдин Е. М.** Создание иллюстраций в МЕТАPOST // *Linux Format*, № 6–10, 2006.
19. **Кнут Д. Э.** Все про МЕТАFONТ. М.: Вильямс, 2003. 376 с.

20. **Волченко Ю. М.** Научная графика на языке *Asymptote*. 2018. 220 с. URL: <http://www.math.volchenko.com/AsyMan.pdf> (дата обращения: 20.12.2018).
21. **Guibé O., Ivaldi P.** *geometry.asy*. Euclidean geometry with *asymptote*. 2011. 95 p.
22. **Беляев Ю. Н.** Векторный и тензорный анализ. Сыктывкар: Изд-во СыктГУ, 2010. 298 с.
23. **Кабардин О. Ф.** Транзисторная электроника. Спецпрактикум. М.: Просвещение, 1972. 207 с.
24. **Жеребцов И. П.** Основы электроники. 5-е изд., перераб. и доп. Л.: Энергоатомиздат, 1989. 352 с.

Summary

Makarov P. A. On the application of the vector graphics language *Asymptote* for illustrating educational, methodical and scientific works of physics and mathematics

The possibility of using the *Asymptote* vector graphics language to illustrate the physics and mathematics educational and scientific works is explored. A few of images illustrating the solution of problems from various fields of physics and mathematics have been developed. It is shown that the *Asymptote* language has convenient high-level syntax and a fairly developed object-oriented architecture.

Keywords: vector graphics, Asymptote, high-level programming language.

References

1. **Lamport L.** *TEX: a document preparation system*, 2 ed., Addison-Wesley, 1994, 291 p.
2. **Lvovski S. M.** *Nabor i verstka v sisteme TEX* (Typesetting in the TEXsystem), 3rd ed, M.: MCNMO, 2003, 448 p.
3. **Kotelnikov I. A., Chebotarev P. Z.** *TEX_{2 ϵ} po-russki* (TEX_{2 ϵ} in Russian), 3rd ed, Novosibirsk: Siberian chronograph, 2004, 496 p.
4. **Znamenskaya O. V., Znamenski S. V., Leinartas D. E., Trutnev V. M.** *Matematicheskaya tipografiya: Kurs lektsiy* (Mathematical typography: The course of lectures), Krasnoyarsk: SFU, 2008, 421 p.

5. **Knut D. E.** *Vse pro T_EX* (All about T_EX), M.: Williams, 2003, 560 p.
6. **Hammerlindl A., Bowman J., Prince T.** *Asymptote: the Vector Graphics Language*, 2016, 189 p.
7. **Kryachkov Yu. G.** *Yevklidova geometriya na yazyke vektornoy grafiki ASYMPTOTE* (Euclidean geometry with asymptote), Volgograd: VGSPU, 2015, 88 p.
8. **Goossens M., Rahtz S., Mittelbach F.** *The L^AT_EX graphics companion: illustrating documents with T_EX and PostScript*, Addison-Wesley, 1997, 299 p.
9. **Goossens M., Rahtz S., Mittelbach F.** *Putevoditel' po paketu L^AT_EX i yego graficheskim rasshireniyam. Illyustrirovaniye dokumentov pri pomoshchi T_EX'a i PostScript'a* (The L^AT_EX graphics companion. Illustrating documents with T_EX and PostScript), M.: Binom, 2002, 621 p.
10. **Kiryutenko Yu. A.** *TikZ& PGF. Sozdaniye grafiki v L^AT_EX 2_ε-dokumentakh* (TikZ& PGF. Creating graphics in L^AT_EX 2_ε-documents), Rostov-on-Don, 2014, 277 p.
11. **Tantau T.** *The TikZand PGF Packages. Manual for version 2.10*, Institut für Theoretische Informatik Universität zu Lübeck, 2010, 880 p.
12. **Taft E., Chernicoff S., Rose C.** *PostScript language reference manual*, 3 ed, Adobe Systems Incorporated, 1999, 912 p.
13. **Reid G. C.** *Thinking in PostScript*, Addison-Wesley Publishing Company, 1990, 239 p.
14. *PostScript language. Tutorial and cookbook*, Addison-Wesley Publishing Company, 1985, 247 p.
15. **Casselmann B.** *Mathematical illustrations: a manual of geometry and PostScript*, Cambridge University Press, 2004, 264 p.
16. **Kryachkov Yu. G.** *Asimptota dlya nachinayushchikh. Sozdaniye risunkov na yazyke vektornoy grafiki Asymptote* (Asymptote for beginners. Creating pictures in the vector graphics language Asymptote), Volgograd: VGSPU, 2015, 131 p.

17. **Hobby J. D.** *METAPOST. Rukovodstvo pol'zovatelya* (METAPOST. User guide), 2008, 106 p. URL: <http://mirrors.ibiblio.org/CTAN/info/metapost/doc/russian/mpman-ru/mpman-ru.pdf> (date of the application: 20.12.2018).
18. **Baldin E. M.** Sozdaniye illyustratsiy v METAPOST (Creating illustrations in METAPOST), *Linux Format*, №6–10, 2006.
19. **Knut D. E.** *Vse pro METAFONT* (All about METAFONT), M.: Williams, 2003, 376 p.
20. **Volchenko Yu. M.** *Nauchnaya grafika na yazyke Asymptote* (Scientific graphics in the Asymptote language), 2018, 220 p. URL: <http://www.math.volchenko.com/AsyMan.pdf> (date of the application: 20.12.2018).
21. **Guibé O., Ivaldi P.** *geometry.asy. Euclidean geometry with asymptote*, 2011, 95 p.
22. **Belyaev Yu. N.** *Vektornyy i tenzornyy analiz* (Vector and tensor analysis), Syktyvkar: Syktyvkar State University, 2010, 298 p.
23. **Kabardin O. F.** *Tranzistornaya elektronika. Spetspraktikum* (Transistor electronics. Special Practice), M.: «Education», 1972, 207 p.
24. **Zherebtsov I. P.** *Osnovy elektroniki* (Fundamentals of Electronics), 5th ed, L.: Energoatomizdat, 1989, 352 p.

Для цитирования: Макаров П. А. О применении языка векторной графики *Asymptote* для иллюстрации учебно-методических и научных работ физико-математической направленности // *Вестник Сыктывкарского университета. Сер. 1: Математика. Механика. Информатика. 2018. Вып. 3 (28). С. 19–37.*

For citation: Makarov P. A. On the application of the vector graphics language *Asymptote* for illustrating educational, methodical and scientific works of physics and mathematics, *Bulletin of Syktyvkar University. Series 1: Mathematics. Mechanics. Informatics*, 2018, 3 (28), pp. 19–37.