

АЗБУКА ХАЛТУРЩИКА-АРМАТУРЩИКА

разработка встраиваемых систем

основы бытовой автоматике,
систем управления и сбора данных

- © ruOpenWrt
- © HackSpace «Чебураторный завод»
- © Консорциум хоббитов России

Оглавление

Введение	9
I Основы электроники	10
1 Линейные схемы на пассивных элементах, основы электротехники	12
2 Симуляция и расчет схем в ngSPICE	13
3 KiCAD	14
3.1 Отрисовка схем в KiCAD	14
3.2 Библиотеки элементов	14
3.3 Передача схемы в ngSPICE	14
4 Простейшие полупроводниковые элементы	15
4.1 Оптоэлектроника	15



4.2	Схемы на биполярных транзисорах	15
4.3	Схемы на на полевых транзисорах	15
5	Операционные усилители	16
6	Источники питания	17
6.1	Батарейное питание	17
6.2	Линейные стабилизаторы	17
6.3	Импульсные преобразователи на ШИМ-контроллерах	17
6.4	Цепи защиты и гашения кондуктивных помех	17
7	Цифровая электроника	18
8	Компьютерные интерфейсы	19
8.1	Поколение 90х: COM, LPT, ISA	20
8.1.1	Резервный программатор AVR “пять проводков”	20
8.2	Сеть CAN	20
8.3	Интерфейсные модули USB	20
8.3.1	Универсальный высокоскоростной конвертер FTDI FT2232H	20
8.3.2	JTAG-адаптер	20
8.3.3	Отладочный модуль CAN	20
8.4	Интерфейсные модули Ethernet	20
9	ПЛИС	21
10	Датчики	22

11 Электропривод и исполнительные устройства	23
II Основы конструирования РЭС	24
12 Пакеты моделирования на основе OpenFOAM	25
13 Обеспечение теплового режима	26
14 Электромагнитная совместимость	27
14.1 Кондуктивные помехи	27
14.2 Компонентные модели и оптимизация кабельной сети	27
III Технология РЭС	28
15 Инструменты и оборудование	29
15.1 JTAG-адаптер	29
15.2 Отладочные платы	29
15.2.1 Arduino /Atmel Mega AVR8/	30
15.2.2 Cortex-Mx	30
15.2.3 CubieBoard /Cortex-A8 AllWinner A10/	31
15.2.4 Raspberry Pi /ARM11 BCM3032/	31
15.2.5 BlackSwift /MIPS/	31
15.2.6 VoCore /MIPS/	31
15.3 Монтажный инструмент	31

15.4	Измерительное оборудование	31
15.4.1	Тестер	31
15.4.2	Осциллограф	31
15.4.3	Логический анализатор	31
15.4.4	Генератор сигналов	31
15.4.5	Рыльцемер	31
16	Трассировка плат и подготовка производства в KiCAD	32
16.1	Технология ЛУТ (Лазерный УТюг)	32
16.2	Технология фоторезиста	32
16.3	Формат Gerber и подготовка промышленного производства	32
17	FreeCAD	33
17.1	Чертеж	34
17.2	Эскиз	34
17.3	Деталь	34
17.4	Сборка	34
17.5	Автогенерация конструкторской документации	34
17.6	Скрипты и пользовательские расширения	34
18	Эксплуатация станочного оборудования	35
19	Основы ЧПУ и цифрового производства	36
19.1	CAM-пакеты для FreeCAD	36

IV	Основы теории систем автоматического управления	37
20	Математический аппарат	38
20.1	Передачная функция	38
20.2	Устойчивость САУ	38
20.3	Сети Петри	38
20.4	Автоматы Маркова	38
21	Релейное управление	39
22	Пропорциональные САУ	40
23	ПИДn-регуляторы	41
V	Разработка ПО для встраиваемых систем	42
24	Вспомогательные скрипты на языке Python	43
24.1	Установка под Windows	45
24.2	Запуск	46
24.3	Дополнительные материалы	51
25	Make: управление сборкой проектов	52
26	VCS: системы контроля версий	53
26.1	CVS	53

26.2 Subversion	53
26.3 Git	53
26.3.1 GitHub	53
27 Основы Си и C₊	54
27.0.2 Установка MinGW (win32)	54
27.1 Особенности C ₊ в embedded	54
28 LLVM и разработка собственных компиляторов	55
28.1 Лексический и синтаксический анализ	55
28.2 Применение flex/bison для разбора текстовых форматов данных	55
28.3 Компилятор Паскаля	55
29 Сборка кросс-компилятора GNU toolchain	56
 VI Микроконтроллеры Cortex-Mx	 57
30 Отладочные платы	58
30.1 STM32DISCOVERY /Cortex-M3 STM32F103/	58
30.2 STM32F4DISCOVERY /Cortex-M4 STM32F407/	58

VII	Периферия	59
VIII	Встраиваемый emLinux	60
31	cross	61
32	BuildRoot	62
33	Особенности OpenWrt	63
34	Библиотека SDL	64
34.1	Реализация microGUI	64
35	Приложения для X Window	65
36	Программирование сетевых приложений	66
37	Сборка кросс-компилятора GNU мальтийским крестом	67
IX	IDE ECLIPSE	68
X	Подготовка публикаций в \LaTeX	69
37.1	Установка MikTeX под  Windows	72
37.2	Структура документа	72

37.2.1	Заголовочный файл или блок	72
37.2.2	Стили документа	72
37.2.3	Пакеты	72
37.2.4	Автор и название	72
37.2.5	Верстка титульных страниц	72
37.2.6	Оглавление	72
37.3	Верстка слайдов	72
37.4	Список литературы и цитирование	72
37.5	Команды секционирования: часть, глава, раздел,.. . . .	75
37.6	Таблицы	75
37.7	Формулы	75
37.8	Перекрестные ссылки и гиперссылки	75
37.9	Листинги скриптов и текстовых данных	75
37.10	Подготовка иллюстраций	75
37.10.1	Графики GNUPLOT	75
37.10.2	Схемы и графы в GraphViz	75

Список литературы	75
--------------------------	-----------

Введение

Первоначально этот материал задумывался как комплект документации к платам BlackSwift и VoCore, но постепенно превратился в толстенный учебник для студентов ВУЗов и научных работников по специализациям, связанным с применением цифровой электроники и компьютерной техники.

Большой упор был сделан на использование открытого некоммерческого программного обеспечения, с целью удешевления учебного процесса, уменьшения себестоимости ваших проектов¹, и стимулирования вашего участия в развитии этих программных пакетов.

Лицензия на эту книгу пока не выбрана, так что она пока просто пишется в духе OpenSource: любой может использовать ее часть, изменять или дополнять, до тех пор, пока не накладываются какие-либо административные, финансовые или юридические ограничения на распространение и развитие оригинальной версии или ее открытых форков.

Приглашаем всех желающих участвовать в развитии этого учебного пособия на форум [ruOpenWrt](#), нам нужна обратная связь по качеству материала, результаты тестирования на вас или ваших студентах, дополнения и замечания.

¹ вряд ли ли у вас окажется лишняя пачка килобаксов на покупку пары коммерческих САПР, по крайней мере пока ваш стартап не взлетит в Тор\$100К

Часть I

Основы электроники

Здесь идет список ссылок на онлайн лекции в edX, Coursera, и т.п.

Глава 1

Линейные схемы на пассивных элементах, основы электротехники

Глава 2

Симуляция и расчет схем в ngSPICE

Глава 3

KiCAD

3.1 Отрисовка схем в KiCAD

3.2 Библиотеки элементов

3.3 Передача схемы в ngSPICE

Глава 4

Простейшие полупроводниковые элементы

4.1 Оптоэлектроника

4.2 Схемы на биполярных транзисорах

4.3 Схемы на на полевых транзисорах

Глава 5

Операционные усилители

Глава 6

Источники питания

6.1 Батарейное питание

6.2 Линейные стабилизаторы

6.3 Импульсные преобразователи на ШИМ-контроллерах

6.4 Цепи защиты и гашения кондуктивных помех

Глава 7

Цифровая электроника

Глава 8

Компьютерные интерфейсы

8.1 Поколение 90х: COM, LPT, ISA

8.1.1 Резервный программатор AVR “пять проводков”

8.2 Сеть CAN

8.3 Интерфейсные модули USB

8.3.1 Универсальный высокоскоростной конвертер FTDI FT2232H

8.3.2 JTAG-адаптер

8.3.3 Отладочный модуль CAN

Глава 9

ПЛИС

Глава 10

Датчики

Глава 11

Электропривод и исполнительные устройства

Часть II

Основы конструирования РЭС

Глава 12

Пакеты моделирования на основе OpenFOAM

Глава 13

Обеспечение теплового режима

Глава 14

Электромагнитная совместимость

14.1 Кондуктивные помехи

14.2 Компонентные модели и оптимизация кабельной сети

Часть III

Технология РЭС

Глава 15

Инструменты и оборудование

15.1 JTAG-адаптер

15.2 Отладочные платы

Прежде чем начать работать с отдельными МК, устанавливая их на плату собственной разработки, для быстрого старта используют *отладочные платы*¹

¹ development board, demo board

15.2.1 Arduino /Atmel Mega AVR8/

15.2.2 Cortex-Mx

См. [30](#)

15.2.3 CubieBoard /Cortex-A8 AllWinner A10/

15.2.4 Raspberry Pi /ARM11 BCM3032/

15.2.5 BlackSwift /MIPS/

15.2.6 VoCore /MIPS/

15.3 Монтажный инструмент

15.4 Измерительное оборудование

15.4.1 Тестер

15.4.2 Осциллограф

15.4.3 Логический анализатор

15.4.4 Генератор сигналов

15.4.5 Рыльцемер

Глава 16

Трассировка плат и подготовка производства в KiCAD

16.1 Технология ЛУТ (Лазерный УТюг)

16.2 Технология фоторезиста

16.3 Формат Gerber и подготовка промышленного производства

Глава 17

FreeCAD

17.1 Чертеж

17.2 Эскиз

17.3 Деталь

17.4 Сборка

17.5 Автогенерация конструкторской документации

17.6 Скрипты и пользовательские расширения

Глава 18

Эксплуатация станочного оборудования

Глава 19

Основы ЧПУ и цифрового производства

19.1 САМ-пакеты для FreeCAD

Часть IV

Основы теории систем автоматического управления

Глава 20

Математический аппарат

20.1 Передаточная функция

20.2 Устойчивость САУ

20.3 Сети Петри

20.4 Автоматы Маркова

Глава 21

Релейное управление

Глава 22

Пропорциональные САУ

Глава 23

ПИДn-регуляторы

Часть V

Разработка ПО для встраиваемых систем

Глава 24

Вспомогательные скрипты на языке Python



Название языка произошло вовсе не от вида пресмыкающихся. Автор назвал язык в честь популярного британского комедийного телешоу 1970-х «Летающий цирк Монти Пайтона». Впрочем, всё равно название языка чаще ассоциируют именно со змеёй, нежели с передачей — пиктограммы файлов в KDE или в Microsoft Windows и даже эмблема на сайте <http://www.python.org> (до выхода версии 2.5) изображают змеиные головы.

Python¹ — высокоуровневый язык программирования общего назначения, ориентированный на повышение производительности разработчика и читаемости кода.

Python удобно применять для написания различных вспомогательных скриптов. Часто его используют при разработке сложных программных систем для написания первых версий. В процессе работы над большими программами часто перерабатываются большие объемы кода, поэтому для ускорения разработки требуется максимально высокоуровневый язык. После того как архитектура программы стабилизируется, узким местом становится производительность, и программу переписывают на более низкоуровневом компилируемом языке, чаще всего C_+^+ .

Написание программ упрощают:

- **объектно-ориентированное программирование** облегчает разработку программ, позволяет переопределить стандартные операторы для пользовательских типов данных, упрощая синтаксис
- **динамическая типизация** не требуется заранее упределять переменные, они создаются простым присваиванием
- **обработка исключений** для секции кода можно определить обработчик ошибок
- **высокоуровневые структуры данных** — списки, словари (набор элементов ключ:значение), очереди
- богатая стандартная библиотека и множество дополнительных библиотек на все случаи

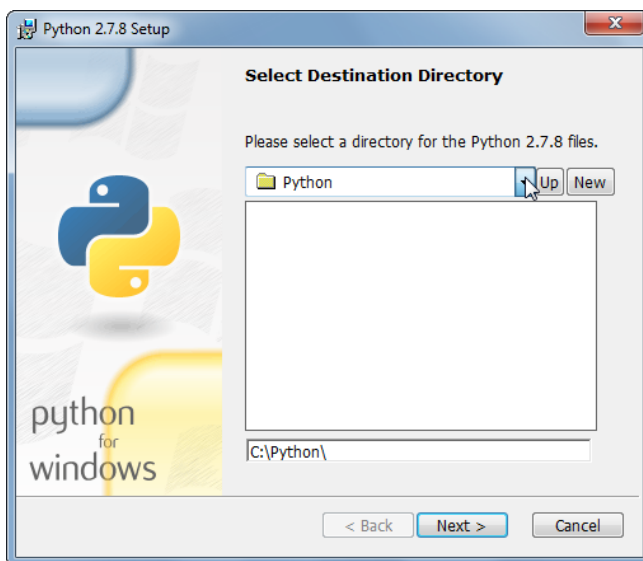
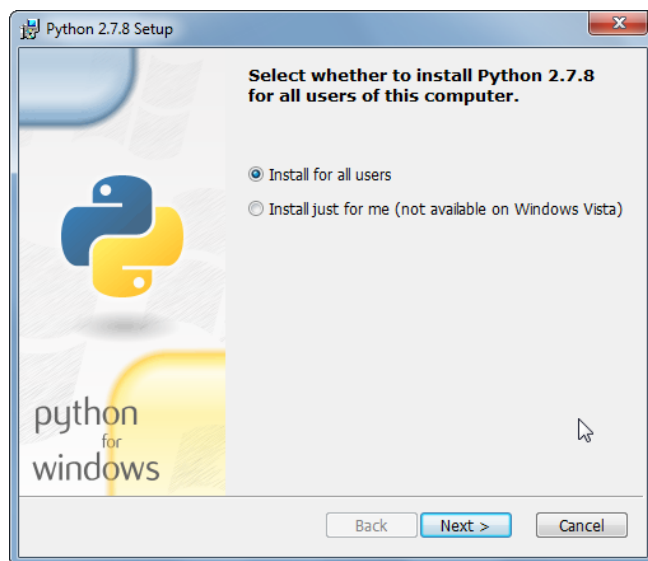
¹ в оригинале читается **пáйтон**, но давно русифицировался как **питóн**

24.1 Установка под Windows

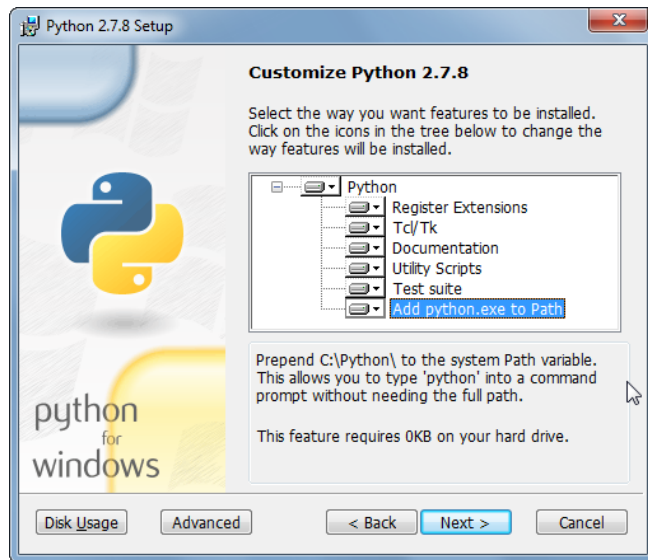
⊞ + R >> <http://www.python.org> >> Downloads >> Python 2.7.8

python-2.7.8.msi >> Setup >> for all users/for me


Destination Directory >> C:/Python/ >> Next

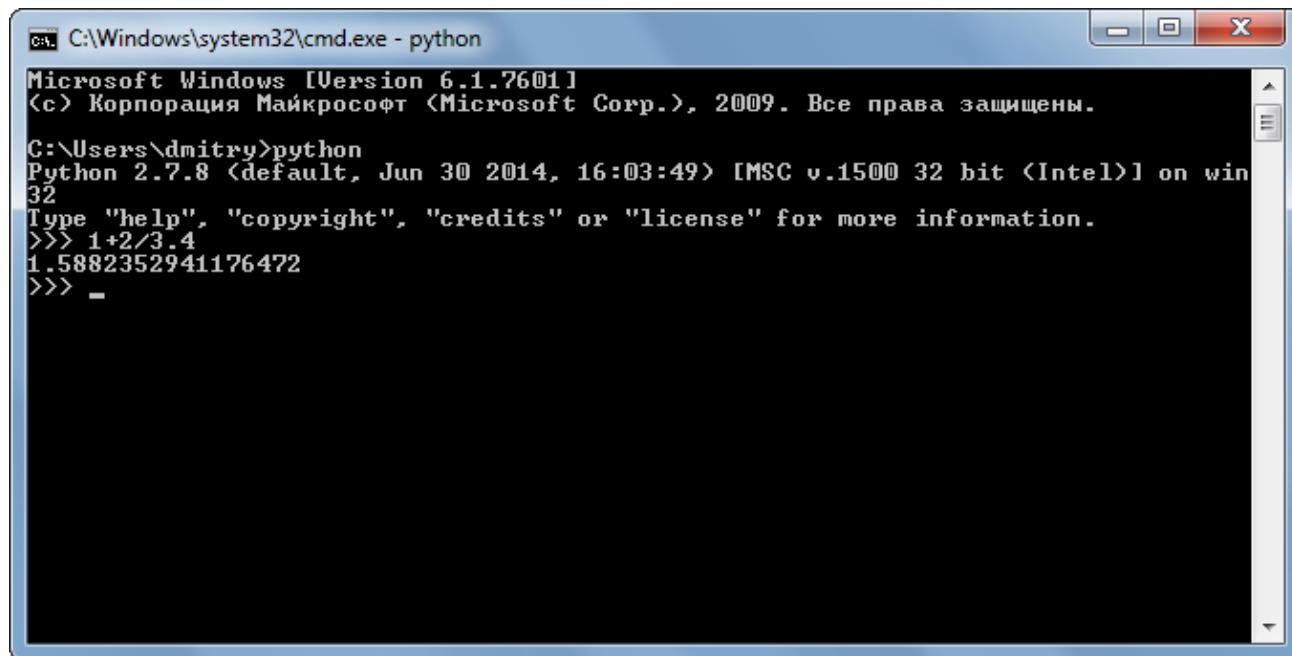


Customize » Python » Add python.exe to PATH » Next » Finish



24.2 Запуск

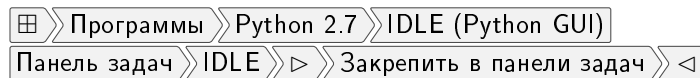
Из командной строки:  + R » cmd » python



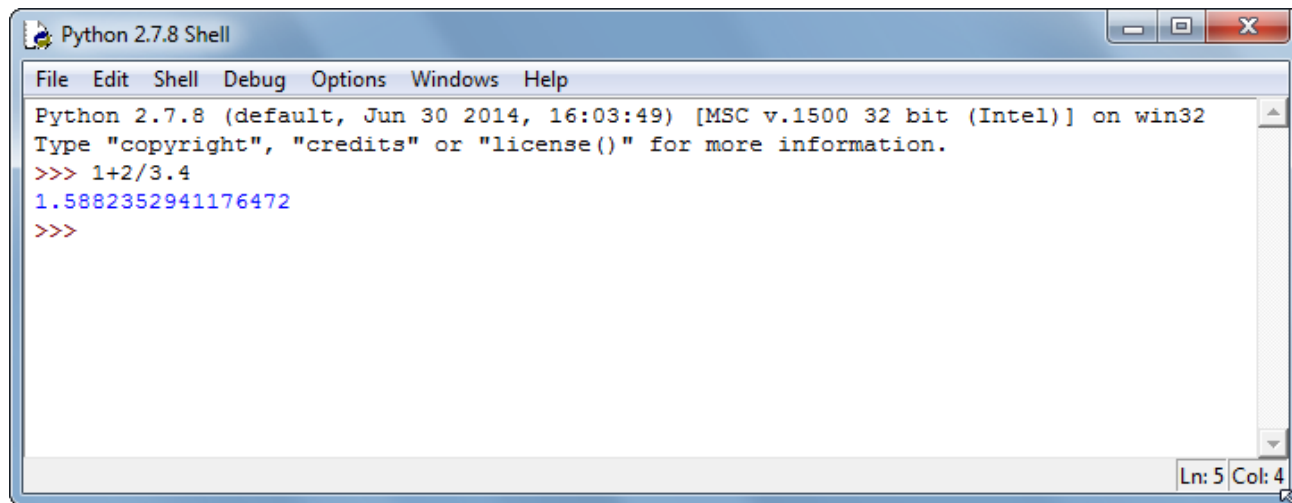
```
C:\Windows\system32\cmd.exe - python
Microsoft Windows [Version 6.1.7601]
(c) Корпорация Майкрософт (Microsoft Corp.), 2009. Все права защищены.

C:\Users\dmitry>python
Python 2.7.8 (default, Jun 30 2014, 16:03:49) [MSC v.1500 32 bit (Intel)] on win
32
Type "help", "copyright", "credits" or "license" for more information.
>>> 1+2/3.4
1.5882352941176472
>>> _
```

Простейшая среда IDLE²:



² на GUI-библиотеке Tkinter, идущей в комплекте



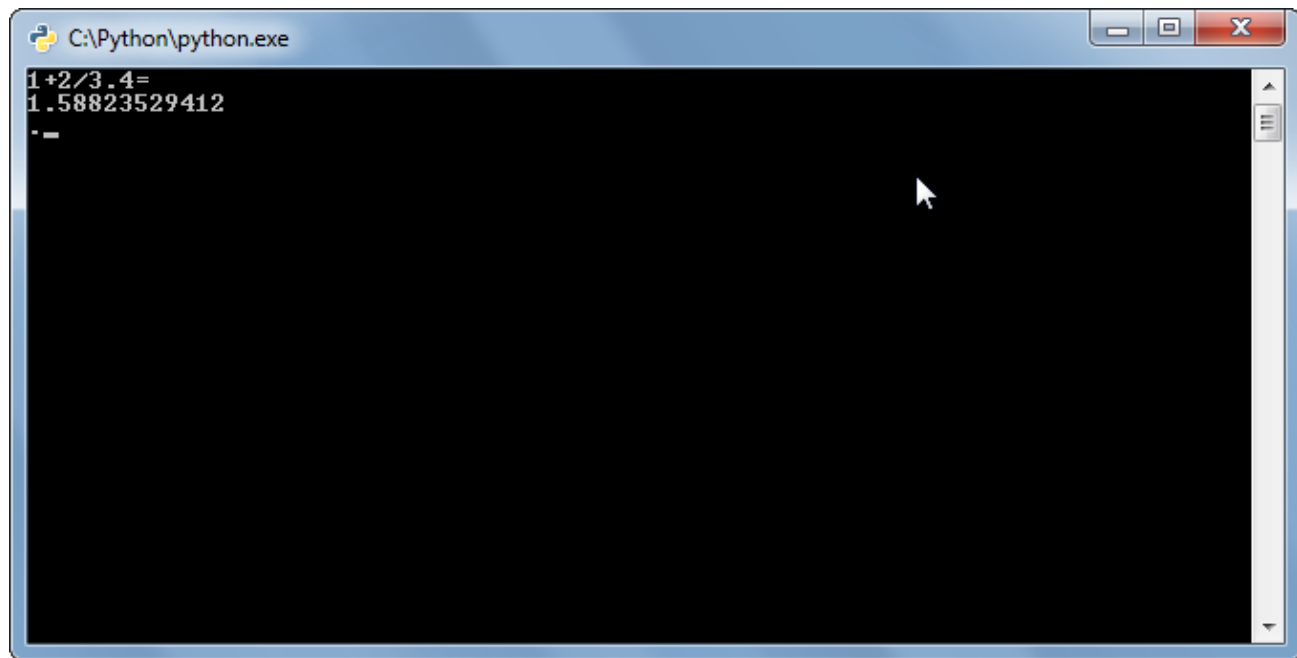
◁◁ по файлу скрипта:

 +  notepad /tmp/py.py

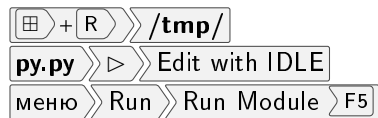
/tmp/py.py

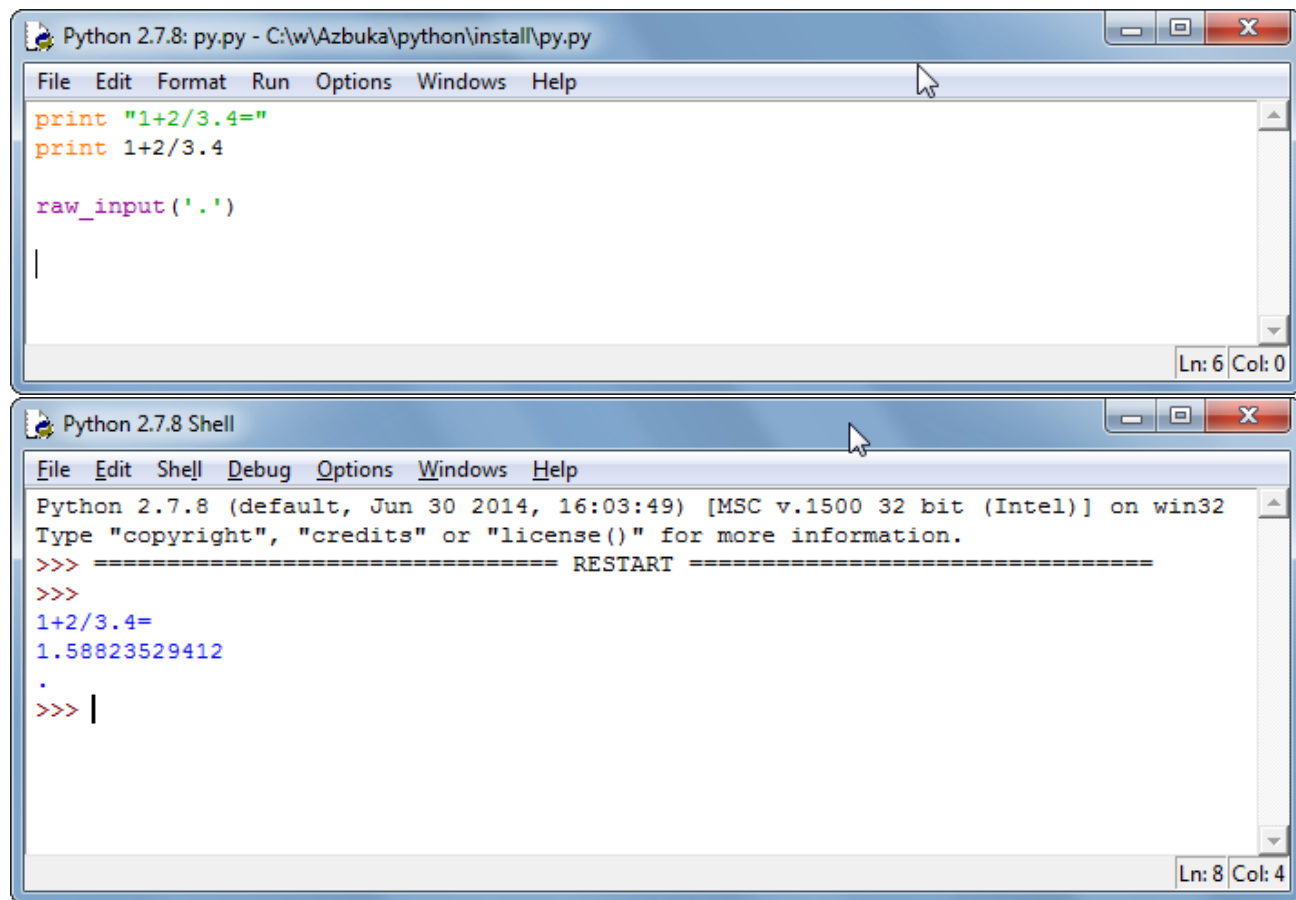
```
1 print "1+2/3.4="
2 print 1+2/3.4
3
4 raw_input( '. ' )
```

 +  /tmp/py.py



Открытием файла скрипта в IDLE:





24.3 Дополнительные материалы

- [2] Г. Россум, Ф.Л.Дж. Дрейк, Д.С. Откидач, [Язык программирования Python](#)
- [1] Аллен Дауни [Думать на языке Python: Думать как компьютерный специалист](#)

Глава 25

Make: управление сборкой проектов

Глава 26

VCS: системы контроля версий

26.1 CVS

26.2 Subversion

26.3 Git

26.3.1 GitHub

Глава 27

ОСНОВЫ СИ И C_+^+

27.0.2 Установка MinGW (win32)

27.1 Особенности C_+^+ в embedded

Глава 28

LLVM и разработка собственных компиляторов

28.1 Лексический и синтаксический анализ

28.2 Применение flex/bison для разбора текстовых форматов данных

28.3 Компилятор Паскаля

Глава 29

Сборка кросс-компилятора GNU toolchain

Часть VI

Микроконтроллеры Cortex-Mx

Глава 30

Отладочные платы

30.1 STM32DISCOVERY /Cortex-M3 STM32F103/

30.2 STM32F4DISCOVERY /Cortex-M4 STM32F407/

Часть VII

Периферия

Часть VIII

Встраиваемый emLinux

Глава 31

CROSS

Глава 32

BuildRoot

Глава 33

Особенности OpenWrt

Глава 34

Библиотека SDL

34.1 Реализация microGUI

Глава 35

Приложения для X Window

Глава 36

Программирование сетевых приложений

Глава 37

Сборка кросс-компилятора GNU мальтийским крестом

Часть IX

IDE  ECLIPSE

Часть X

Подготовка публикаций в L^AT_EX

LaTeX (по-русски произносится **латéx**) — наиболее популярный набор макрорасширений (или макропакет) системы компьютерной вёрстки T_EX, который облегчает набор сложных документов. В типографском наборе форматируется как L^AT_EX.

Главная идея L^AT_EX состоит в том, что авторы должны думать о содержании, о том, что они пишут, не беспокоясь о конечном визуальном облике (печатный вариант, текст на экране монитора или что-то другое). Готовя свой документ, автор указывает логическую структуру текста (разбивая его на главы, разделы, таблицы, изображения), а L^AT_EX решает вопросы его отображения. Так содержание отделяется от оформления. Оформление при этом или определяется заранее (стандартное), или разрабатывается для конкретного документа.

В практическом смысле использование L^AT_EX позволяет (в порядке уменьшения важности):

- с помощью макросов и T_EX-программирования реализовывать любые стили и самую сложную верстку, существует множество готовых пакетов для верстки графических химических формул, разнообразных схем, транскрипционных знаков, внезапно электронных схем, цветных листингов и т.п.
- автоматизировать работу с документами: пересобирать выходные файлы через [Make](#), генерировать части документов с помощью своих скриптов²
- получить выходной документ в .pdf .html .txt .PostScript .djvu ... с кликабельными ссылками, анимированными, а иногда и интерактивными элементами
- не использовать файлы документов в закрытом формате

¹ копияста <https://ru.wikipedia.org/wiki/LaTeX>

² отчеты, стандартные формы, результаты работы любых программ

- легко держать набор файлов в [VCS](#)
- не покупать текстовый процессор

Особенно важен пункт про сложную верстку: она всегда нужна в крупных технических публикациях, особенно в учебной литературе, или отчетных работах. Вам обязательно понадобится вставлять графики экспериментальных данных, тематически специфичные схемы, листинги, выходные данные работы ваших программ и т.п.

Традиционно \LaTeX любим математиками, и всеми кто готовит публикации с большим количеством формул и перекрестных ссылок: после небольшого обучения формулы вводятся с листа со скоростью набора текста, особенно если ваш редактор умеет автодополнение, и никакой мышью возни.

Естественно всякие чисто автоматические вещи типа автонумерации ссылок и формул, сборки оглавлений и индексов, цветовая подсветка синтаксиса в листингах программ, размещение плавающих иллюстраций и т.п. выполняются автоматически \TeX -процессором в пакетном режиме, и на выходе получается красивый печатный или электронный (.pdf) документ.

Единственная область, не удобная в \LaTeX -верстке — создание сложных таблиц. Для этого были созданы визуальные редакторы, позволяющие отрисовать структуру таблицы мышью, а затем заполнить готовый шаблон данными.

37.1 Установка MikTeX под Windows

37.2 Структура документа

37.2.1 Заголовочный файл или блок

37.2.2 Стили документа

37.2.3 Пакеты

37.2.4 Автор и название

37.2.5 Верстка титульных страниц

37.2.6 Оглавление

37.3 Верстка слайдов

37.4 Список литературы и цитирование

L^AT_EX умеет мощную подсистему управления цитированием и списками литературы. В простейшем случае, например при написании единственной статьи, раздел *библиографии* можно создать в том же документе, добавив в конец `thebibliography`:

```
\documentclass{article}
```

```

\input{header}

\author{Вася Пупкин}
\title{Пример статьи с цитатами}

\begin{document}
\maketitle

В статье используются книги: \cite{A} и \cite{B}

\begin{thebibliography}{99}

\bibitem{A} Книга А

\bibitem{B} Книга В

\end{thebibliography}
\end{document}

```

Но если вы регулярно работаете с документацией, или часто пишете статьи, возникает естественное желание вынести весь список литературы в отдельную базу данных, прописать авторов, названия, издательства и т.п. Это делается с помощью программы **biber** и пакета **biblatex**.

Пример использования этой системы вы легко найдете в исходниках этой книги:

- файл **header.tex** содержит секцию подключения пакета и подгрузки библиофайлов:

```
% books bib management
\usepackage{biblatex}
\addbibresource{../bib/python.bib}
\addbibresource{../bib/eskd.bib}
...
```

- библиофайлы хранятся в **соседнем** репозитории `../bib`, клонированном с <https://github.com/ponyatov/bib>.
- порядок вызова `pdflatex` и `biber` см. `Makefile`

37.5 Команды секционирования: часть, глава, раздел,..

37.6 Таблицы

37.7 Формулы

37.8 Перекрестные ссылки и гипессылки

37.9 Листинги скриптов и текстовых данных

37.10 Подготовка иллюстраций

37.10.1 Графики GNUPLOT

37.10.2 Схемы и графы в GraphViz

Список литературы

- [1] Аллен Дауни. *Думать на языке Python: Думать как компьютерный специалист*. 1999. URL: <https://drive.google.com/file/d/0B0u4WeMj0894Q2hWV1QwOFFQ0Vk/view?usp=sharing>.
- [2] Г. Россум и др. *Язык программирования Python*. Stichting Mathematisch Centrum, 1990–1995 и др., 2001, с. 454. URL: <http://rus-linux.net/MyLDP/BOOKS/python.pdf>.