

VASIKЪ

© Dmitry Ponyatov <dponyatov@gmail.com>, GNU Lesser GPL
институт комического приборостроения СГОУ

13 марта 2016 г.

This system is free software: you can redistribute it and/or modify it under the terms of the GNU Lesser Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

This system is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU Lesser Public License along with this program. If not, see <http://www.gnu.org/licenses/>.

Это свободная программа: вы можете перераспространять ее и/или изменять ее на условиях Меньшей общественной лицензии GNU в том виде, в каком она была опубликована Фондом свободного программного обеспечения; либо версии 3 лицензии, либо (по вашему выбору) любой более поздней версии.

Эта программа распространяется в надежде, что она будет полезной, но БЕЗО ВСЯКИХ ГАРАНТИЙ; даже без неявной гарантии ТОВАРНОГО ВИДА или ПРИГОДНОСТИ ДЛЯ ОПРЕДЕЛЕННЫХ ЦЕЛЕЙ. Подробнее см. в Меньшей общественной лицензии GNU.

Вы должны были получить копию Меньшей общественной лицензии GNU вместе с этой программой. Если это не так, см. <http://www.gnu.org/licenses/>.

Разрешается использование в составе **коммерческих программ** на условиях Меньшей общественной лицензии GNU (GNU LGPL) при условии использования оригинальной копии исходных текстов с <https://github.com/ponyatov/VASIC>¹, или предоставления полных исходных текстов модифицированной копии на условиях той же лицензии GNU LGPL. При отсутствии технических возможностей² разрешается также статическая линковка, при условии предоставления исходных текстов.

¹ или зеркала

² актуально для прошивок микроконтроллерных устройств

Оглавление

Введение	3
lex skeleton: скелет лексической программы (Си-вариант)	4
Загрузка и сборка	4
1 Синтаксис языка VASIKЪ	
(с примерами кода из батареи тестов)	6
1.1 Комментарии	6
1.2 Скаляры	7
1.2.1 sym: символ (идентификатор)	7
1.2.2 str: строка	7
1.2.3 int: целое число	8
1.2.4 num: число с плавающей точкой	8
1.2.5 hex: машинное целое	8
1.2.6 bin: битовая строка	9
1.3 Команды	9
1.3.1 PRINT	9

Введение

VASIКЪ — диалект языка *BASIC*, реализация основана на [видеокурсе по созданию интерпретатора BASIC](#) и кое-каких собственных наработках по *лексическому программированию*. Синтаксис построен на основе видеокурса, но сделаны некоторые заимствования из *Python* и функциональных языков.

Ядро написано на стандартном ANSI Си³, специально для кросс-компиляции и выполнения интерпретатора на микроконтроллерах. C_+^+ очень удобен для компьютеров, но для встраиваемых применений требует крайне профессиональной и аккуратной адаптации C_+^+ -кода и библиотек поддержки⁴ для выполнения на микроконтроллерах. Эта проблема связана с критически маленьким объемом ОЗУ, и очень активным использованием динамической памяти программами на C_+^+ .

³ gcc -ansi , специально чтобы можно было компилировать код другими кросс-компиляторами: для микроконтроллеров, не входящих в список gcc targets, и коммерческими типа IAR, Keil, Intel, OpenWatcom, Visual C,...

⁴ **newlib**, менеджер динамической памяти, C_+^+ -runtime

lex skeleton: скелет лексической программы (Си-вариант)

<code>test/</code> → <code>src.src</code>		исходники отдельных тестов интерпретатора
<code>log.log</code>		лог выполнения теста в интерпретаторе
<code>y.y</code>	yacc/bison	синтаксический анализатор: грамматика языка
<code>l.l</code>	lex/flex	лексер: распознавание единичных элементов <i>токенов</i>
<code>h.h</code>	C_+^+	хедеры общие для всех модулей кода (лексер/парсер/ядро)
<code>c.c</code>	C_+^+	ядро языка, реализация всех объектов языка на ANSI Си
<code>Makefile 2.1</code>	make	скрипты сборки проекта утилитой make
<code>bat.bat</code>	(g)Vim	запускалка редактора (g)Vim под <i>Windows</i>
<code>.gitignore</code>	git	маски файлов, не включаемых в git -репозиторий
<code>README.md</code>	github	описание проекта
<code>manual.tex</code>	\LaTeX	исходный текст этого руководства на языке верстки \LaTeX
<code>git⁵</code>		

Загрузка и сборка

Пакет предоставляется в **исходных текстах** с репозитория <https://github.com/ponyatov/VASIC>. Для загрузки и компиляции необходимо установить:

GNU toolchain пакет (кросс-)компиляторов GNU

Windows дистрибутив **MinGW** <http://www.mingw.org/>,
установить **gcc g++ make flex bison**,
если нужна компиляция научных библиотек **fortran**

⁵ система управления версиями исходного текста, и сетевое хранилище [GitHub](#)

Linux `sudo apt install gcc g++ make flex bison [fortran ...]`

Git система контроля версий

Windows <https://git-scm.com/>

Необязательное ПО

(g)Vim редактор текстовых файлов (IDE)⁶

Windows <http://www.vim.org/download.php>

Л^AT_EX система верстки на основе языка разметки

⁶ необязателен, клавиатурные сочетания и приемы работы отличны от *Windows*-feel и требуют привыкания, но отличается легковесностью, настраиваемостью (скриптами) и мультиплатформенностью

Глава 1

Синтаксис языка VASIKЪ (с примерами кода из батареи тестов)

1.1 Комментарии

Листинг 1: test/comment.src

```
# line comment from # to end of line  
# строчный комментарий, от # до конца строки
```

Листинг 2:

1.2 Скаляры

Скаляры — типы объектов, соответствующие **единичным** самодостаточным элементам данных, которые можно считать атомами информатики¹: символ (идентификатор), строка, число.

1.2.1 sym: символ (идентификатор)

Листинг 3: test/sym.src

```
this is symbol
```

Листинг 4:

```
<this> <is> <symbol>
```

1.2.2 str: строка

Листинг 5: test/str.src

Листинг 6:

¹ computer science

1.2.3 int: целое число

Листинг 7: test/int.src

Листинг 8:

1.2.4 num: число с плавающей точкой

Листинг 9: test/num.src

Листинг 10:

1.2.5 hex: машинное целое

Листинг 11: test/hex.src

Листинг 12:

1.2.6 bin: битовая строка

Листинг 13: test/bin.src

Листинг 14:

1.3 Команды

1.3.1 PRINT

Реализация команды `print` рассмотрено в уроке [Make Your Own Programming Language - Part 1 - Lexer](#)

Листинг 15: test/print.src

```
# print <expression>
# print <выражение>
print 'hello world'
```

Листинг 16:

```
<print:hello world>
```

Глава 2

Исходный код ядра

2.1 Makefile

Листинг 17: Makefile