

Д. А. Слинкин,

Шадринский государственный педагогический университет, Курганская область

ИСПОЛЬЗОВАНИЕ ЯЗЫКА ПРОГРАММИРОВАНИЯ FREE PASCAL И ПРОГРАММНОЙ БИБЛИОТЕКИ RUBIROBOTLIB ДЛЯ УПРАВЛЕНИЯ РОБОТАМИ НА ПЛАТФОРМЕ LEGO MINDSTORMS EV3

Аннотация

В статье рассматриваются робототехническая платформа LEGO MINDSTORMS EV3, достоинства и недостатки среды разработки LEGO MINDSTORMS Education EV3, возможности программирования робота с использованием альтернативных языков программирования и сторонних прошивок. Особое внимание уделяется авторскому проекту RubiRobot и библиотеке RubiRobotLib, позволяющей разрабатывать программы для LEGO MINDSTORMS EV3 на языке программирования Free Pascal.

Ключевые слова: LEGO MINDSTORMS EV3, образовательная робототехника, алгоритм, программирование, Free Pascal.

DOI: 10.32517/2221-1993-2018-17-7-8-12

Контактная информация

Слинкин Дмитрий Анатольевич, канд. пед. наук, доцент кафедры программирования и автоматизации бизнес-процессов Шадринского государственного педагогического университета, Курганская область; *адрес:* 641800, Курганская область, г. Шадринск, ул. К. Либкнехта, д. 3; *телефон:* (35253) 3-63-69; *e-mail:* xds1@list.ru

D. A. Slinkin,
Shadrinsk State Pedagogical University,
Kurgan Region

USING THE FREE PASCAL PROGRAMMING LANGUAGE AND THE RUBIROBOTLIB SOFTWARE LIBRARY TO CONTROL ROBOTS ON THE LEGO MINDSTORMS EV3 PLATFORM

Abstract

The article examines the LEGO MINDSTORMS EV3 robotic platform, the advantages and disadvantages of LEGO MINDSTORMS Education EV3 development environment, the possibilities of programming a robot using alternative programming languages and third-party firmware. Particular attention is paid to the RubiRobot authoring project and the RubiRobotLib software library which allows developing programs for LEGO MINDSTORMS EV3 on the Free Pascal programming language.

Keywords: LEGO MINDSTORMS EV3, educational robotics, algorithm, programming, Free Pascal.

Робототехнические системы различных типов прочно вошли в образовательный процесс школ, ссузов и вузов. Предмет «Робототехника» и робототехнические кружки позволяют изучить конструирование и программирование роботов, оценить их возможности и практическую применимость алгоритмических конструкций, использовать роботов в решении прикладных задач. Широко распространены российские и международные робототехнические соревнования: World Robot Olympiad [2, 26], «РобоФинист» [7], «Робофест» [3], соревнования JuniorSkills [18] в компетенции «Мобильная робототехника» и многие другие.

Одним из лидеров — производителей образовательных робототехнических систем заслуженно считается компания LEGO, особенностью работы которой является комплексный подход к продвижению своей продукции — это не только производство и продажа конструкторов, но и целая образовательная экосистема: курсы для учителей, печатная и электронная образовательная продукция, организация и спонсирование робототехнических олимпиад.

В последние годы широко распространение в России получила робототехническая платформа **LEGO MINDSTORMS Education EV3** [8] с сопутствующим программным обеспечением. «Сердцем» платформы является программируемый блок-микрокомпьютер, на порядок превосходящий по своим характеристикам блок предыдущей серии LEGO MINDSTORMS NXT. Однако подход к программированию блока особо не изменился, LEGO продолжает продвигать графический язык программирования, теперь уже в новой среде разработки LEGO MINDSTORMS Education EV3, которая поддерживает работу и с предыдущей платформой. В помощь учителям, школьникам и энтузиастам разработано множество самоучителей, учебников и методических рекомендаций, предназначенных для изучения самой платформы [1, 9, 11], обучения основным алгоритмам [1, 10] и программированию в среде разработки [1, 10, 11].

Среда разработки от компании LEGO имеет низкий порог вхождения, может использоваться школьниками и студентами всех ступеней обучения, однако, обладает рядом существенных **недостатков**:

- Создание сложных и объемных программ в данной среде крайне затруднено, уже при десятке собственных блоков резко увеличивается латентность отклика на любое действие, загрузка больших программ в память среды может длиться до нескольких минут, переключение между проектами или программами в одном проекте происходит с длительными задержками. Указанная проблема воспроизводилась нами на нескольких де-

сятках компьютеров, на операционных системах Microsoft Windows XP, 7, 8.1, 10, с различными блоками EV3 и разными версиями среды разработки. При создании простых алгоритмов проблема, естественно, не проявляется, но при решении серьезных задач, например, во время подготовки к робототехническим соревнованиям и во время них, работа крайне замедляется.

- В среде невозможна разработка некоторых видов алгоритмов, например, с использованием прямой или косвенной рекурсии. Данный факт является прямым препятствием к эффективному использованию среды в робототехнических соревнованиях.
- Данное программное обеспечение не функционирует ни на одной операционной системе российского производства из внесенных в Единый реестр российских программ для электронных вычислительных машин и баз данных (<https://reestr.minsvyaz.ru/>). Таким образом, с учетом реализуемой в России программы импортозамещения возможность дальнейшего применения среды в государственных образовательных учреждениях находится под вопросом.

Существует возможность заменить среду разработки LEGO MINDSTORMS Education EV3 на другие среды и языки программирования, как функционирующие на оригинальной прошивке LEGO EV3, так и требующие для корректной работы загрузки сторонних прошивок и операционных систем. Специфика контроллера EV3 позволяет осуществлять загрузку с microSD-карты, оставляя оригинальную прошивку без изменений, что, в свою очередь, позволяет вернуться к штатной загрузке при возникновении каких-либо ошибок.

Среди сторонних ОС следует выделить две:

- leJOS [20], представляющую собой минимальный дистрибутив Linux с запущенной виртуальной машиной Java,
- и ev3dev [17] — полноценный дистрибутив Linux Debian с возможностью использования любых языков программирования, доступных для ОС Linux.

Существует множество сред и языков программирования для EV3 — как текстовых, так и графических. Среди графических следует выделить TRIK Studio [25], которая кроме собственного контроллера TRIK поддерживает и контроллер LEGO EV3, является свободным программным обеспечением и функционирует в большинстве операционных систем. Среди учебных сред стоит обратить внимание на Scratch [12] и Open Roberta [23], среди коммерческих профессиональных — на LabVIEW [19] и его адаптированный для образования клон ROBOLAB [22].

Для текстовых языков программирования важным является наличие программных библиотек поддержки аппаратных подсистем и подключаемых устройств LEGO EV3. Указанными качествами обладают: Java проекта leJOS, EV3 Basic [16], такие языки проекта ev3dev, как Java, Python, Go, C, C++ и многие другие.

В то же время в РФ для обучения программированию в большинстве школ используются разные диалекты языка программирования Pascal (Delphi [15], PascalABC [14], Free Pascal [21] и т. п.). До настоящего времени не существовало средств разработки на языке Pascal для EV3 (за исключением экспериментальной генерации кода на Pascal в TRIK Studio), что затрудняло приме-

ние школьниками полученных на уроках информатики знаний и умений для программирования робототехнических систем.

Указанную проблему решают **проект RubiRobot** [13] и свободно распространяемая **программная библиотека RubiRobotLib**, доступная по адресу: <http://rubirobot.ru/files/src/RubiRobotLib.tar.gz>.

Библиотека RubiRobotLib разработана на языке программирования Free Pascal и представляет собой программный интерфейс в виде набора функций, классов и объектов с поддержкой всех внутренних и подключаемых устройств LEGO EV3. RubiRobotLib требует загрузки робота с оригинального или модифицированного в рамках проекта RubiRobot дистрибутива Linux Debian проекта ev3dev. Оригинальный дистрибутив доступен на сайте проекта ev3dev по адресу: <https://www.ev3dev.org/downloads>, модифицированный — на сайте проекта RubiRobot по адресу: <http://rubirobot.ru/files/img/ev3dev-rubirobot.img.bz2>. Библиотека содержит подробную документацию [6] в формате PDF/A, несколько десятков примеров различной степени сложности, консольный менеджер, позволяющий ускорить и упростить передачу данных между хостовым компьютером и роботом.

Подготовка текста программы с использованием библиотеки и ее компиляция могут производиться непосредственно на LEGO EV3 в ssh-сессии без установки специализированного ПО на персональный компьютер. Для подключения к роботу по протоколу ssh можно использовать произвольный ssh-клиент, который в ОС российского производства присутствует обычно изначально, а в ОС Windows может быть установлен дополнительно [24]. Запуск полученного в результате компиляции исполняемого файла может также производиться в ssh-сессии либо с помощью управляющего LEGO EV3 приложения Brickman. В первом случае штатные средства ввода-вывода Free Pascal будут обмениваться с пользователем данными в рамках терминала ssh-соединения. Во втором — взаимодействовать с пользователем через дисплей LEGO EV3, кнопки блока и клавиатуру (в случае ее подключения к USB-порту контроллера). С учетом невысокой (по сравнению с персональным компьютером) производительности LEGO EV3 более эффективным вариантом является создание программ с использованием средств кросскомпиляции, что дает значительный выигрыш в скорости разработки. Пакет программ RubiRobotCross проекта RubiRobot, доступный по адресу: <http://rubirobot.ru/files/img/setupRubiRobot.tgz>, содержит в себе все необходимое для организации кросскомпиляции и предназначен для использования на российских операционных системах Alt Linux, седьмой [4] и восьмой [5] платформах. В частности, работоспособность RubiRobotCross была протестирована на дистрибутивах Simply Linux 7.0, Альт Линукс 7.0 Кентавр, Альт Образование 8.2 и некоторых других. Ведутся работы по модификации RubiRobotCross для установки на другие дистрибутивы Linux, такие как Debian, Arch, Calculate. С учетом реализуемой в России программы импортозамещения модификация RubiRobotCross для ОС Windows признана на текущий день автором проекта нецелесообразной. Но все компоненты проекта выпускаются под лицензией LGPLv3, что позволяет портировать RubiRobotCross на ОС Windows любому стороннему разработчику.

Рассмотрим некоторые возможности управления устройствами контроллера LEGO EV3 с использованием библиотеки RubiRobotLib. Предполагаем, что разработка программ ведется в ssh-сессии непосредственно на контроллере LEGO EV3 загруженного с модифицированного в рамках проекта RubiRobot дистрибутива Linux Debian проекта ev3dev. Все нижеприведенные примеры входят в состав библиотеки RubiRobotLib, их исходный код может быть получен совместно с библиотекой. В качестве текстового редактора предполагаем использование встроенного редактора файлового менеджера Midnight Commander. В то же время исключительно для удобства визуализации листинги программ в данной статье будут представлены в виде скриншотов из среды программирования Geany.

1. Светоиндикация.

Блок EV3 имеет красный и зеленый светоиндикаторы слева от центральной кнопки и аналогичную пару — справа от центральной кнопки. Яркость светоиндикаторов определяется числовым значением от 0 до 255. При одновременном свечении красного и зеленого индикаторов возникает эффект смешения цветов, что позволяет получать отдельные оттенки желтого и оранжевого. Данный эффект демонстрирует программа, представленная в листинге 1.

```
led0.pp x
1  {$mode objfpc}
2  uses uev3, rubiroLeds;
3  var i:integer;
4  begin
5    ev3Init();
6    for i:=0 to 255 do ev3Leds.all(255-i,i,i,255-i);
7    for i:=0 to 255 do ev3Leds.all(i,255-i,255-i,i);
8  end.
```

Листинг 1. Программа демонстрации светоиндикации

Программа пригодна для запуска как в ssh-сессии, так и через Brickman. Рассмотрим ее более подробно.

Строка 1 отключает установленный по умолчанию режим поддержки синтаксиса Турбо Паскаля (tp) и включает режим поддержки расширенного синтаксиса Free Pascal с объектными возможностями (objfpc). Если не указать данный режим, то в более сложных программах возможно появление различных видов ошибок — от синтаксических при компиляции до логических при исполнении, так как библиотека RubiRobotLib разработана с использованием возможностей режима objfpc.

В строке 2 программа подключает необходимые для работы модули. Наличие базового модуля uev3 является обязательным при работе программы на LEGO EV3, он содержит необходимые средства инициализации библиотеки RubiRobotLib. Модуль rubiroLeds содержит объект ev3Leds для управления светоиндикацией контроллера.

В строке 5 процедура ev3init() инициализирует основные объекты библиотеки. Ее вызов должен быть осуществлен первым действием в программе. До этого момента попытка обращения к любым возможностям библиотеки может завершиться ошибкой и аварийным остановом программы. Исключением из этого правила является набор специальных переменных, предназначенных для тонкой настройки возможностей библиотеки и дающих необходимый эффект только при изменении перед вызовом ev3init(). Например, чтобы «форсировать» чтение данных с устройств, достаточно до вызова ev3init() установить в 0 значение переменной getInterval из модуля uev3sysfs.

В строке 6 первый цикл программы обеспечивает «переливание» цветов: левый ярко-красный индикатор плавно изменяет цвет на ярко-зеленый, правый ярко-зеленый индикатор плавно изменяет цвет на ярко-красный.

В строке 7 второй цикл проделывает обратную операцию.

Стоит обратить внимание, что подобного эффекта цветоиндикации невозможно добиться при использовании среды программирования LEGO MINDSTORMS Education EV3, так как в ней невозможно управлять уровнями яркости цветоиндикаторов — они могут находиться только в одном из двух состояний: либо выключенном, либо включенном на максимальной яркости.

2. Робот-путешественник.

В этом примере робот с закрепленным впереди датчиком расстояния (рис. 1) путешествует по открытому пространству с набором препятствий. При обнаружении препятствия в пределах 50 сантиметров робот случайным образом изменяет направление движения и продолжает свое путешествие.



Рис. 1. Робот-путешественник

Обязательными требованиями к роботу является наличие двух больших моторов (EV3 или NXT) и ультразвукового датчика расстояния (EV3). Левый мотор должен быть подключен к порту с меньшим номером, правый — к порту с большим номером (например: А и В, С и D, В и D и т. д.). Датчик расстояния может быть подключен к любому порту.

Программа управления роботом представлена в листинге 2.

```
sns7_1.pp x
1  {$mode objfpc}
2  uses uev3, rubiroSensors, rubiroMotors, rubiroButtons;
3  begin
4    ev3Init();
5    ev3Buttons.waitCenter;
6    while not ev3Buttons.back do begin
7      if ev3Ultra.lenCM>50 then
8        ev3Rule.Run(0,20)
9      else
10       ev3Rule.RunTurnWait(100-200*random(2),
11                           0.2+random(2,15));
12    end;
13  end.
```

Листинг 2. Программа управления роботом-путешественником

Как и в предыдущем случае, программа пригодна для запуска и в ssh-сессии, и через Brickman. Рассмотрим ее более подробно.

В строке 2 последовательно подключаются базовый модуль, модуль поддержки датчиков, модуль поддержки двигателей и модуль поддержки кнопок блока EV3.

Инициализирует библиотеку строка 4.

В строке 5 робот блокируется вплоть до нажатия кнопки CENTER блока, сигнализируя об ожидании миганием цветоиндикаторов.

В строке 6 начинается цикл, выход из которого предусмотрен по нажатию на кнопку BACK блока. Метод `ev3buttons.back` является асинхронным, не блокирующим выполнение программы. Он возвращает текущее состояние соответствующей кнопки.

Строки 7–10 цикла представляют собой ветвление, регламентирующее поведение робота. Если расстояние до препятствия больше 50 сантиметров, то робот продолжает движение вперед вызовом асинхронного метода `Run` объекта рулевого управления со скоростью в 20 % от максимальной. В противном случае робот осуществляет поворот вокруг оси синхронным методом `RunTurnWait` влево или вправо от 0,2 до 0,7 оборота со скоростью 15 % от максимальной.

3. Дистанционное управление роботом с помощью инфракрасных датчика и маяка.

Конструкция робота с дистанционным управлением (рис. 2) отличается от робота-путешественника заменой датчика расстояния на инфракрасный датчик, который будет использоваться для управления роботом совместно с инфракрасным маяком. Предполагается использование первого канала маяка, красные кнопки которого отвечают за скорость правого мотора, синие — за скорость левого мотора. В простейшем случае инфракрасный датчик можно установить на те же самые крепления, которые использовались датчиком расстояния (как показано на рисунке 2). Однако для увеличения радиуса приема сигнала рекомендуется вынести инфракрасный датчик на верхнюю часть робота.

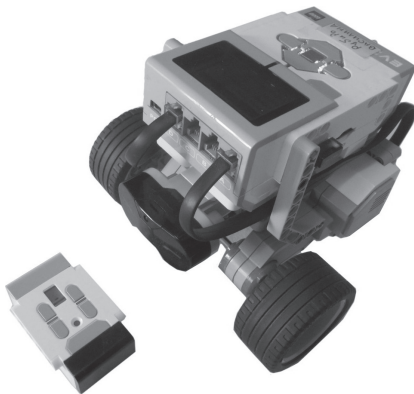


Рис. 2. Робот с дистанционным управлением

Программу можно запускать и в ssh-сессии, и через Brickman. Рассмотрим программу управления роботом (листинг 3), опуская очевидные моменты.

В строке 4 определяются два объекта класса `TEv3LargeMotor`, которые затем в строке 8 ассоциируются с левым и правым моторами EV3.

В строке 5 определяется переменная — множество типа `TEv3IRButtons`, которая будет содержать набор кнопок, нажатых на инфракрасном маяке.

Строки 9–17 представляют собой основной цикл программы, выход из которого производится по нажатию на кнопку BACK блока.

В строке 10 из объекта инфракрасного датчика считывается множество нажатых в настоящий момент на маяке кнопок.

```

1  {$mode objfpc}
2  uses uev3, rubiroSensors, rubiroMotors, rubiroButtons;
3  var
4    LM, RM: TEv3LargeMotor;
5    s: TEv3IRButtons;
6  begin
7    ev3Init();
8    LM:=ev3motor1; RM:=ev3motor2;
9    while not ev3buttons.back do begin
10     s:=ev3infra.anybuttons;
11     if [irRU,irRD]*s=[] then RM.stop;
12     if [irBU,irBD]*s=[] then LM.stop;
13     if irRU in s then RM.Run(100);
14     if irRD in s then RM.Run(-100);
15     if irBU in s then LM.Run(100);
16     if irBD in s then LM.Run(-100);
17   end;
18 end.

```

Листинг 3. Программа дистанционного управления роботом

В строках 11 и 12 при отпущенных кнопках управления выполняется остановка.

Строки 13–16 используются для организации прямого или обратного движения каждого из моторов в зависимости от нажатых кнопок блока. Положительное значение скорости дает движение вперед, отрицательное — назад.

4. Дистанционное управление роботом с помощью клавиатуры.

Если доступ к роботу организован через беспроводную сеть и программа запускается в ssh-сессии, то появляется возможность управления движением робота с клавиатуры (листинг 4). Конечно, клавиатурное управление возможно и при других способах доступа к EV3, но проводное подключение через USB ограничивает движение робота длиной USB-провода, а запуск программы с использованием Brickman требует подключения клавиатуры непосредственно к роботу, что ограничивает перемещение робота теперь уже длиной провода клавиатуры. Единственным приемлемым вариантом использования Brickman будет являться подключение беспроводной клавиатуры.

Конструкция робота аналогична любой из вышеописанных, наличие или отсутствие датчиков значения не имеет.

```

1  {$mode objfpc}
2  uses crt, uev3, rubiroMotors, sysutils;
3
4  var ch: char; command: char=#0; speed: integer=20;
5  begin
6    writeln('Робот с ручным управлением');
7    writeln('Клавиша ESC - завершение работы программы');
8    writeln('Клавиши курсора ↑, ←, →, ↓ - движение робота');
9    writeln('Клавиши от 1 до 9 - скорость робота');
10   writeln('Остальные клавиши - остановка робота');
11   ev3init();
12   repeat
13     ch:=readkey;
14     if ch in ['1'..'9'] then speed:=10*strToInt(ch)
15     else if ch=#0 then command:=readkey
16     else command:=#0;
17     case command of
18       #72: ev3rule.Run(0, speed);
19       #80: ev3rule.Run(0, -speed);
20       #75: ev3rule.Run(-100, speed);
21       #77: ev3rule.Run(100, speed)
22     else ev3rule.Stop();
23   end;
24   until ch=#27;
25 end.

```

Листинг 4. Программа управления роботом с помощью клавиатуры

Для получения информации о нажатых клавишах можно использовать разнообразные средства. Например, Free Pascal предлагает использовать кроссплатформенный модуль `keyboard` с поддержкой Unicode, функциональных клавиш, комбинаций клавиш и т. д. Однако в нашем случае возможности данного модуля останутся в основном невостребованными. Поэтому предлагаем для указанной задачи использовать модуль `ctrl`, портированный во Free Pascal из Turbo Pascal и остающийся неизменным уже много лет. Рассмотрим подробно код программы, опуская очевидные моменты.

В строке 4 определены три основные переменные: `ch` — код нажатой клавиши, `command` — сканкод нажатой клавиши/команда роботу, `speed` — скорость движения робота.

Строки 12–24 содержат основной цикл получения/обработки нажатий клавиш и реакций на нажатия. Выход из цикла — по нажатию на клавишу ESC (код 27).

Строки 13–16 используются для чтения значений нажатых клавиш, формирования команды роботу и установкой скорости его движения.

В строках 17–23 применяется сформированная ранее команда. В зависимости от команды обеспечивается старт моторов вперед или назад с заданной скоростью, вращение робота вокруг оси влево или вправо. Если команда не соответствует сканкодам клавиш курсора, производится остановка робота.

* * *

Представленная программная библиотека может применяться при изучении программирования на уроках информатики, в программировании роботов на предмете «Робототехника», на занятиях в робототехнических кружках, при подготовке к робототехническим соревнованиям, на любых робототехнических соревнованиях, где отсутствуют ограничения на используемое программное обеспечение, а также при самостоятельном программировании LEGO EV3. Автор библиотеки не ограничивает возможностей ее модификации, разрешает ее распространение, изучение и применение в любых целях в соответствии с лицензией LGPLv3. Проект RubiRobot динамично развивается, ведутся работы по расширению его возможностей. Автор приглашает всех заинтересованных лиц к использованию библиотеки и участию в развитии данного проекта.

Список использованных источников

1. Валк Л. Большая книга LEGO MINDSTORMS EV3: подробное руководство для начинающих по постройке и программированию роботов / пер. с англ. С. В. Черникова. М.: Издательство «Э», 2017.
2. Всероссийская робототехническая олимпиада. <http://robolymp.ru/>

3. Всероссийский робототехнический фестиваль «Робофест». <http://www.russianrobofest.ru/>
4. Дистрибутивы и репозиторий 7-й платформы Alt Linux // Файловый архив Alt Linux. <http://ftp.altlinux.ru/pub/distributions/ALTLinux/p7/>
5. Дистрибутивы и репозиторий 8-й платформы Alt Linux // Файловый архив Alt Linux. <http://ftp.altlinux.ru/pub/distributions/ALTLinux/p8/>
6. Документация проекта RubiRobot // Проект RubiRobot (о роботах и не только). <http://rubirobot.ru/files/src/rubiro.manual.archive.pdf>
7. Ежегодный Международный фестиваль робототехники «РобоФинист». <https://robofinist.ru/>
8. Знакомство с EV3 // LEGO.com RU — Вдохновлять и учить строителей будущего. <https://www.lego.com/ru-ru/mindstorms/about-ev3>
9. Исогава Й. Книга идей LEGO MINDSTORMS EV3. 181 удивительный механизм и устройство / пер. с англ. О. В. Обручева. М.: Издательство «Э», 2017.
10. Овсянникова Л. Ю., Овсяницкий Д. Н., Овсяницкий А. Д. Алгоритмы и программы движения по линии робота LEGO MINDSTORMS. М.: Перо, 2015.
11. Овсянникова Л. Ю., Овсяницкий Д. Н., Овсяницкий А. Д. Курс программирования робота LEGO MINDSTORMS EV3 в среде EV3: основные подходы, практические примеры, секреты мастерства. Челябинск: ИП Мякотин И. В., 2014.
12. Программирование робота LEGO MINDSTORMS EV3 с помощью Scratch 2.0 // ProXays. <http://www.proghouse.ru/article-box/52-scratch2-ev3>
13. Проект RubiRobot (о роботах и не только). <http://rubirobot.ru>
14. Язык программирования PascalABC.NET. <http://pascalabc.net>
15. Delphi — Обзор // Fast Cross-Platform App Development Software — Embarcadero. <https://www.embarcadero.com/ru/products/delphi>
16. EV3 Basic // The official site for tutorials on EV3 Basic (Microsoft Small Basic with the EV3 extension). <https://sites.google.com/site/ev3basic>
17. ev3dev is your EV3 re-imagined. <http://www.ev3dev.org>
18. JuniorSkills — программа ранней профессиональной подготовки и профориентации школьников. <https://juniorskills.ru/>
19. LabVIEW — программное обеспечение для системного проектирования // National Instruments. <http://www.ni.com/ru-ru/shop/labview.html>
20. leJOS. Java for LEGO MINDSTORMS. <http://www.lejos.org>
21. Open source compiler for Pascal and Object Pascal. <http://freepascal.org>
22. Platform ROBO LAB // LEGO Engineering, Center for Engineering Education and Outreach. <http://www.legoengineering.com/platform/robolab>
23. Programming environment Open Roberta Lab. <https://lab.open-roberta.org>
24. PuTTY is an SSH and telnet client. <http://www.putty.org/>
25. TRIK. Learn to invent the future // ООО «КиберТех». <http://blog.trikset.com>
26. World Robot Olympiad. <https://wro-association.org/>