

# **ИНТЕГРАЦИЯ ПОДДЕРЖКИ ЯЗЫКА PASCAL ДЛЯ TRIK И TRIK STUDIO**

Белков Р., бакалавр кафедры системного программирования  
СПбГУ, roman.belkov@gmail.com

Лимар И., бакалавр математико-механического факультета СПбГУ,  
limvan@mail.ru

## **Аннотация**

В докладе описывается опыт интеграции текстового программирования на языке Pascal в среду TRIK Studio. Рассмотрены возможности, предоставляемые DSM-платформой Qreal [1] для создания генераторов текстовых языков. Данный опыт может быть использован при создании генераторов текстовых языков промышленного программирования, например, C++ и Java.

## **Введение**

Значительное место в начале процесса обучения основам программирования занимает визуальное программирование, так как оно является наиболее понятным и доступным способом познакомиться с абстракциями выполнения программ. Визуальные программы, созданные учащимися, могут так же помогать им на последующих этапах обучения.

Например, при помощи генераторов визуальных диаграмм в код на текстовом языке возможно совершить плавный переход от визуального программирования к текстовому программированию, давая ученикам понять, как конструкции из диаграммы представляются на текстовом языке программирования.

## **Постановка проблемы**

Для набирающего популярность кибернетического контроллера TRIK и одноимённого конструктора потребовалось создать возможность использования языка Pascal для программирования роботов. Поскольку Pascal является самым популярным языком, преподаваемым в российских школах (см. Рис. 1), то разработчикам бесплатной среды программирования TRIK Studio с открытым исходным кодом постоянно поступали просьбы и вопросы о поддержке языка Pascal. Тогда школьники, имеющие навыки работы с Pascal на персональном компьютере, смогут без проблем попробовать свои силы в программировании роботов. Если же школьнику преподавали визуальное программирование, а дальнейшая

программа обучения подразумевает знакомство и работу с Pascal, то при помощи генератора ученик сможет осознать, как уже знакомая программа выглядит на Pascal.

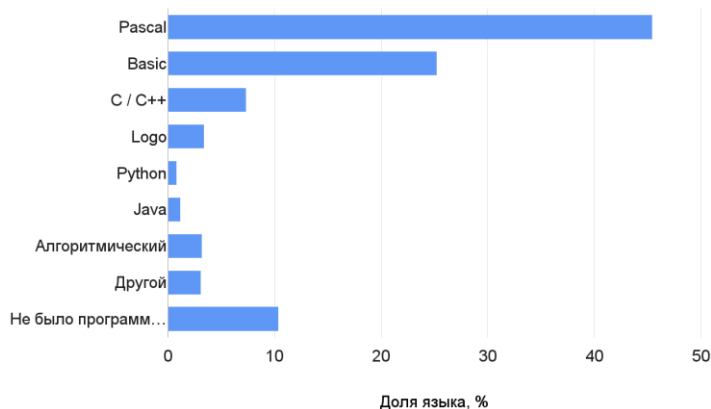


Рисунок 1: Распределение популярности ЯП, преподаваемых в российских школах по данным опроса на HabraHabr [2]

### *Pascal*

В начале работы надо было определиться, какую из многочисленных реализаций языка Pascal стоит использовать. Рассматривались самые популярные и поддерживаемые реализации.

Основными критериями оценки компиляторов стали следующие:

- популярность в образовательной среде;
- поддержка сообществом;
- свободное распространение компилятора;
- кодогенерация для архитектуры ARM.

Последний пункт — кодогенерация — подразумевает два разных сценария использования: кросс-компиляция и нативная компиляция прямо на работе. Надо понимать, что в первом случае возникает необходимость найти и собрать кросс-компилятор, поскольку маловероятно, что он будет присутствовать в поставке среды программирования. Во втором случае одним из важных факторов является доступность готовой сборки компилятора для конечного пользователя.

Дистрибутив на основе GNU/Linux для контроллера ТРИК собирается при помощи Yocto/OpenEmbedded [3], как и для многих других

контроллеров. На момент исследования, к сожалению, отсутствовали готовые рецепты для сборки компиляторов языка Pascal.

Так как самым популярным микрокомпьютером с процессором архитектуры ARM является Raspberry Pi, который широко используется, в том числе, в образовании, в сравнении мы рассматривали на равных компиляторы, доступные для этой платформы.

Результат анализа доступных компиляторов-кандидатов представлен в следующей таблице (См. Табл. 1).

Название	GNU Pascal	Free Pascal	Pascal ABC.NET	mikroPascal
Свободное ПО	+	+	+	-
Компиляция для ARM	-	+	Генерация в байт-код .NET (CIL)	+
Кросс-компиляция для ARM	-	+		+

Таблица 1. Сравнение различных компиляторов языка Pascal

В результате выбор сузился до Free Pascal или Pascal ABC.NET. При использовании Free Pascal перед запуском программы на роботе приходилось бы компилировать код прямо на контроллере, что занимает ценные при отладке время и ресурсы контроллера. Более того, пришлось писать библиотеку для работы с периферией контроллера, несмотря даже на такие разработки, как поддержка инструментария работы с периферией на уровне файловой системы [4]. С другой стороны, PascalABC.NET позволяет максимально переиспользовать существующие технологии, так как в результате компиляции получается байт-код на .NET, а с прошивкой контроллера ТРИК поставляется топо и уже была реализована .NET библиотека программирования роботов [5, 6]. Таким образом, при выборе Pascal ABC.NET нет необходимости разрабатывать и поддерживать кросс-компилятор и специальную библиотеку.

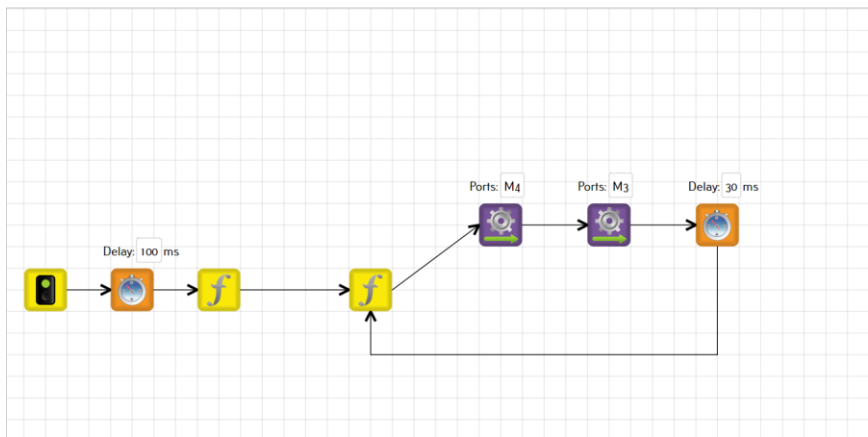
### ***Генератор Pascal в TRIK Studio***

Вторая часть работы — это реализация собственно поддержки языка Pascal: генерация из визуальных программ текстового представления, предоставление компилятора и подсветки синтаксиса, а также инструменты для загрузки скомпилированной программы и контроля

выполнения программы.

Значительную часть этой работы берёт на себя TRIK Studio, предлагая готовый инструментарий и основной каркас для реализации вышеперечисленных функций. Модульная архитектура генераторов TRIK Studio позволяет легко переиспользовать готовые компоненты, расширяя функциональность при необходимости. Фактически, только генерацию текстового представления программы и подсветку синтаксиса необходимо реализовывать вручную.

При этом создание простейшего генератора целевого языка сводится к описанию текстовыми шаблонами синтаксических конструкций языка. В TRIK Studio уже реализовано всё необходимое, включая преобразование графа потока управления к представлению, пригодному для генерации структурного кода. Уже созданные примеры генераторов языков JavaScript и F# позволяют понять, как именно корректно создать генератор желаемого целевого языка.



```

1  {reference './../trikSharp/Trik.Core.dll'}
2
3  const
4    pi := 3.1415926535897931;
5
6  var
7    model : Trik.Model;
8    S : integer;
9    Sold : integer;
10   u : real;
11
12 begin
13   Randomize;
14   model := Trik.Model.Create();
15
16   model.Sleep(100);
17   S := model.AnalogSensors[Trik.SensorPort.A1].Read();
18   Sold := S;
19   while true do
20     begin
21       u := 2.5 * (S - model.AnalogSensors[Trik.SensorPort.A1].Read()) + 5 * (Sold - model.AnalogSensors[Trik.SensorPort.A1].Read());
22       Sold := model.AnalogSensors[Trik.SensorPort.A1].Read();
23       model.Motors[Trik.MotorPort.M4].SetPower(50 + u);
24       model.Motors[Trik.MotorPort.M3].SetPower(50 - u);
25       model.Sleep(30);
26     end;
27 end.
28

```

Рисунок 2, 3: Диаграмма программы «езда около коробки» и сгенерированный код этой программы на языке Pascal

### *Дальнейшая работа*

Прототип описанного решения демонстрировался на VI всероссийской конференций «Современное технологическое обучение: от компьютера к роботу». На момент описания работы уже поддержан

полный цикл разработки программ для роботов с промежуточным представлением на Pascal, но пока ещё в стадии разработки реализация в текстовом редакторе среды TRIK Studio подсветки синтаксиса Pascal. В дальнейшем планируется оптимизировать процесс загрузки и последующего исполнения программы на роботе и поддерживать полный набор датчиков, входящих в комплект контроллера ТРИК.

## Заключение

В результате работы был реализован и интегрирован в среду TRIK Studio генератор языка Pascal ABC.NET для создания и последующего исполнения программ на кибернетическом контроллере ТРИК. Опыт, описанный в работе, может быть полезен при создании генераторов языков промышленного программирования, например Java и C++.

## Литература

1. Брыксин Т., Литвинов Ю. Технология визуального предметно-ориентированного проектирования и разработки ПО Qreal // Материалы второй научно-технической конференции молодых специалистов «Старт в будущее», посвященной 50-летию полета Ю.А. Гагарина в космос — СПб — 2011 — С. 222-225  
<http://qreal.ru/papers/QReal-technology-2011.pdf>
2. Процентное соотношение языков программирования, которые сейчас преподаются в средней школе <https://habrahabr.ru/post/207020/>
3. Главная страница проекта OpenEmbedded  
[http://www.openembedded.org/wiki/Main\\_Page](http://www.openembedded.org/wiki/Main_Page)
4. Бурдун Ф. Разработка инструментария уровня файловой системы для унифицированного управления периферийными устройствами контроллера ТРИК // Дипломная работа — 2014  
[http://se.math.spbu.ru/SE/diploma/2014/s/BurdunFiodar\\_Diploma.pdf](http://se.math.spbu.ru/SE/diploma/2014/s/BurdunFiodar_Diploma.pdf)
5. Kirsanov A, Kirilenko I, Melentyev K. Robotics Reactive Programming with F#/Mono // Proceedings of the 10th Central and Eastern European Software Engineering Conference in Russia. — 2014. — p. 16:1 — 16:5.  
<http://doi.acm.org/10.1145/2687233.2687249>
6. Кирсанов А. Программирование роботов с помощью .NET/Mono // Дипломная работа — 2015  
<http://se.math.spbu.ru/SE/diploma/2015/bmo/444-Kirsanov-report.pdf>