

# Интеграция языка PyСи в робоконструкторы

Фадеев Виктор Юрьевич

5 июня 2020 г.

Научный руководитель: проф. каф. СП, д.ф.-м.н.,  
профессор А.Н. Терехов

Рецензент: директор ООО «Новые Мобильные  
Технологии» В.В. Оносовский

PyСи — язык с C-подобным синтаксисом, максимально защищенный от ошибок пользователя.

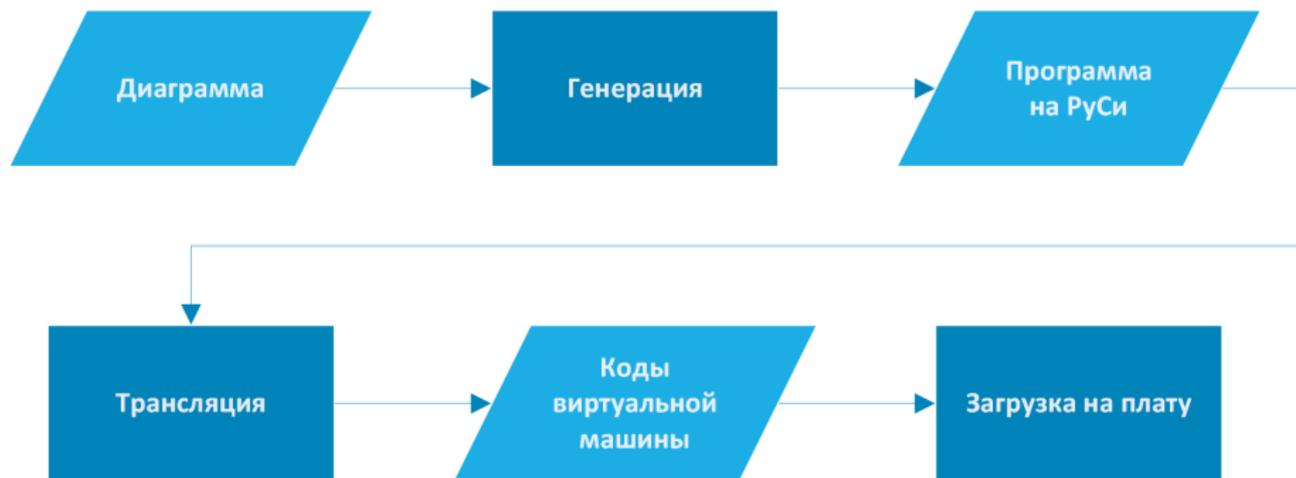
TRIK Studio — графическая среда разработки программ для роботов.

ЙоТик 32 v2.0 — контроллер компании MGBot на базе микропроцессора ESP32.

Целью работы является внедрение системы программирования PyСи в TRIK Studio для использования в робоконструкторе ЙоТик. Задачи:

- добавить в TRIK Studio плагин для контроллера ЙоТик
- реализовать генерацию кода на языке PyСи для контроллера ЙоТик
- разработать требования для новой прошивки ЙоТик 32 v2.0
- произвести рефакторинг проекта PyСи для выполнения требований новой прошивки

## На инструментальном компьютере



## Непосредственно на работе

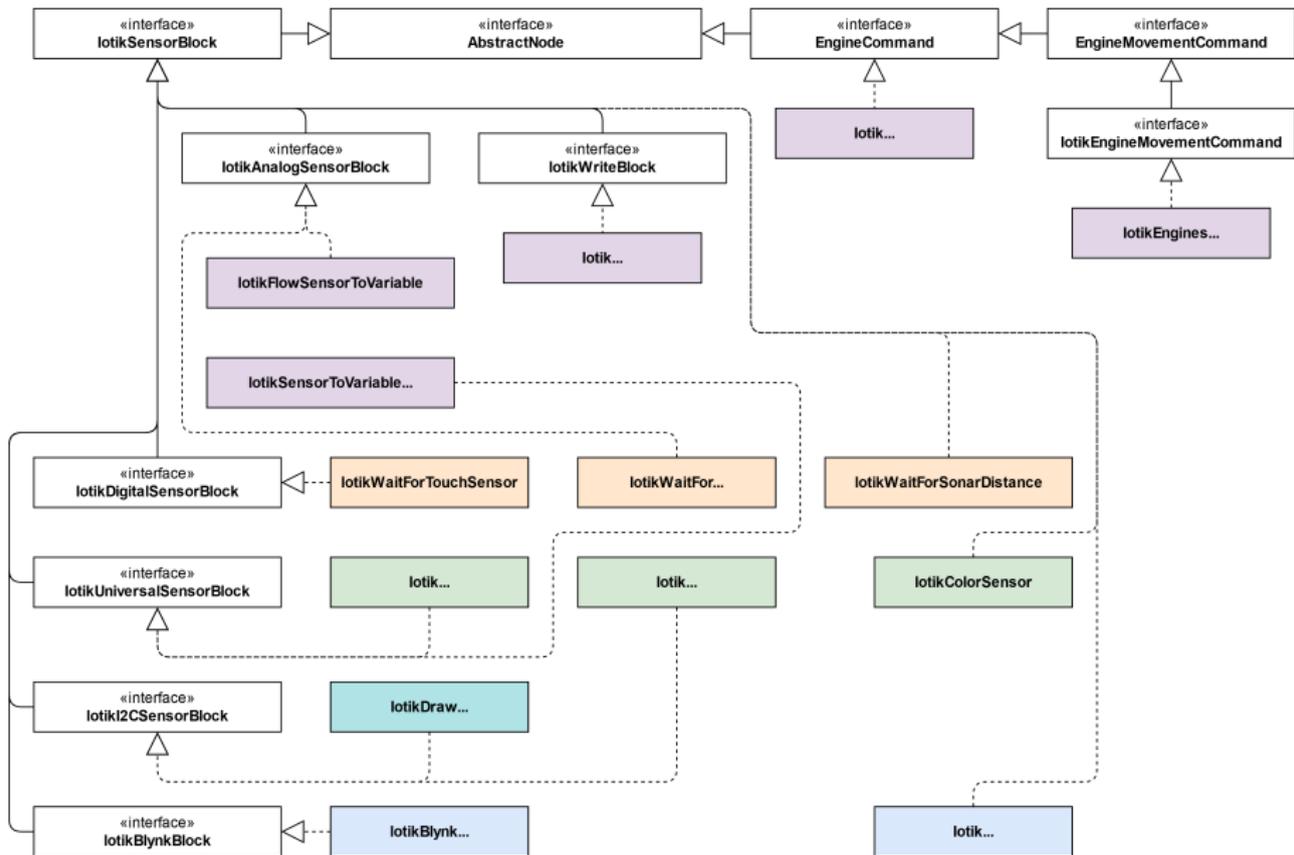
Основные программы динамической поддержки (среды выполнения) и интерпретатор виртуальной машины загружаются 1 раз, а коды исполняемых программ можно загружать много раз.

- добавление комплекта ЙоТик
- создание новых блоков датчиков (освещенности, расстояния, ультрафиолетовый, емкостный, акселерометр, компас, гироскоп, атмосферного давления, влажности и температуры), модулей (RGB светодиод, LED матрица, реле, I2C двигатели) и элементов работы с сетью
- реализация генераторов РуСи (русский/английский)

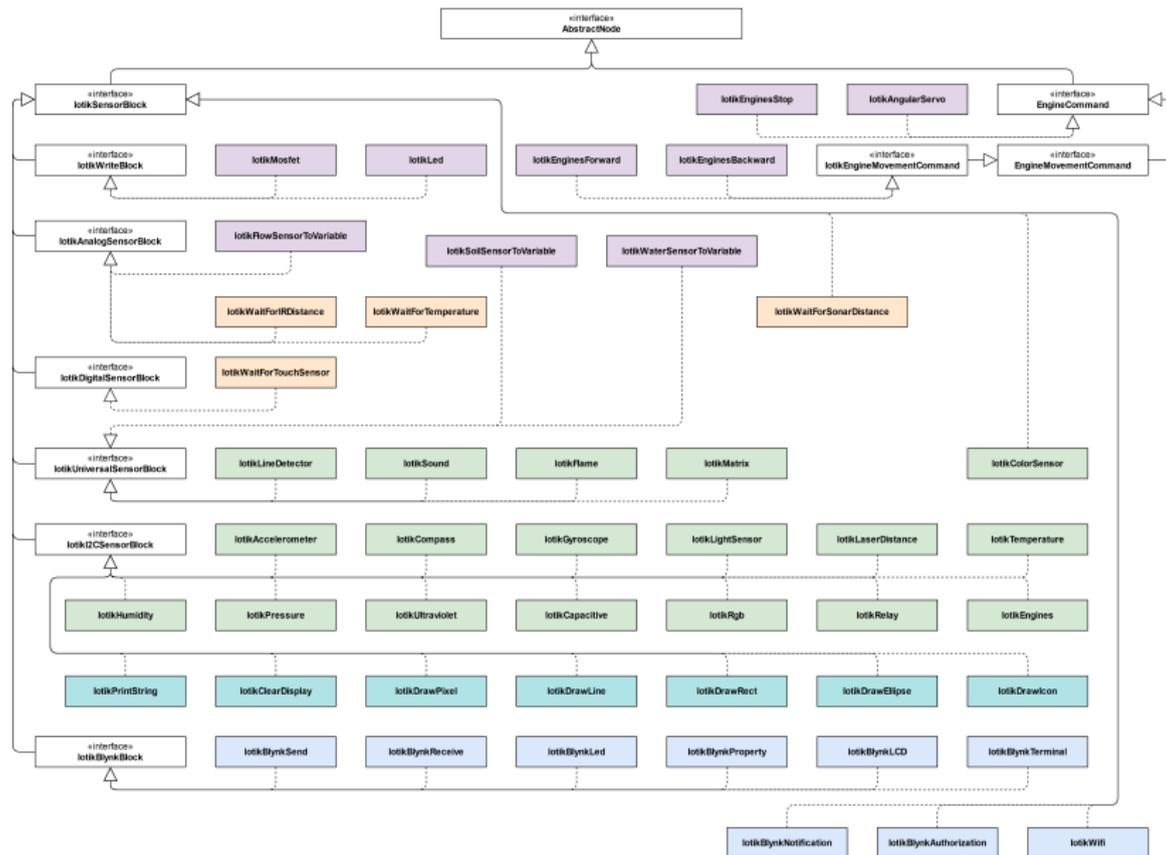
В отличие от TRIK и Лего для комплекта ЙоТик:

- нет необходимости конфигурировать порты в настройках TRIK Studio, тем самым в робот загружается на один файл меньше
- есть датчики, использующие несколько портов
- возможно использовать одни и те же порты для разных датчиков (I2C)

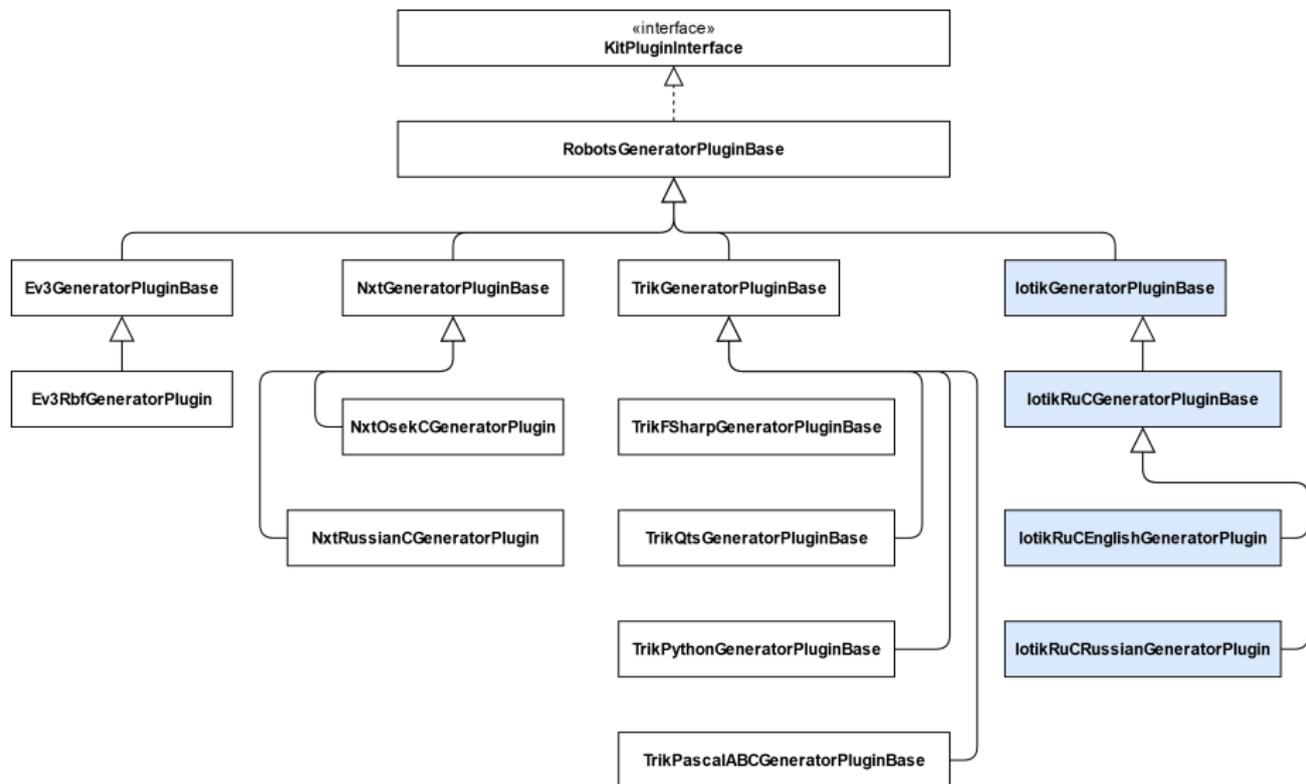
# Обобщенная метамодель



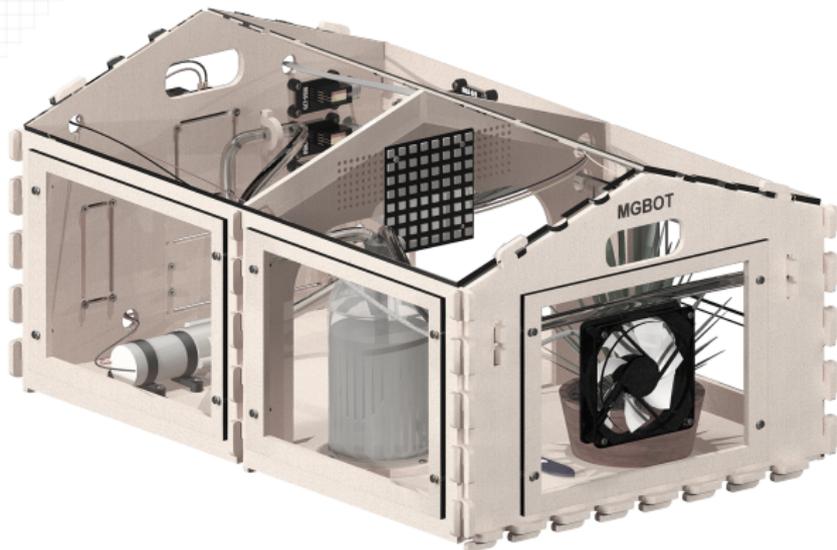
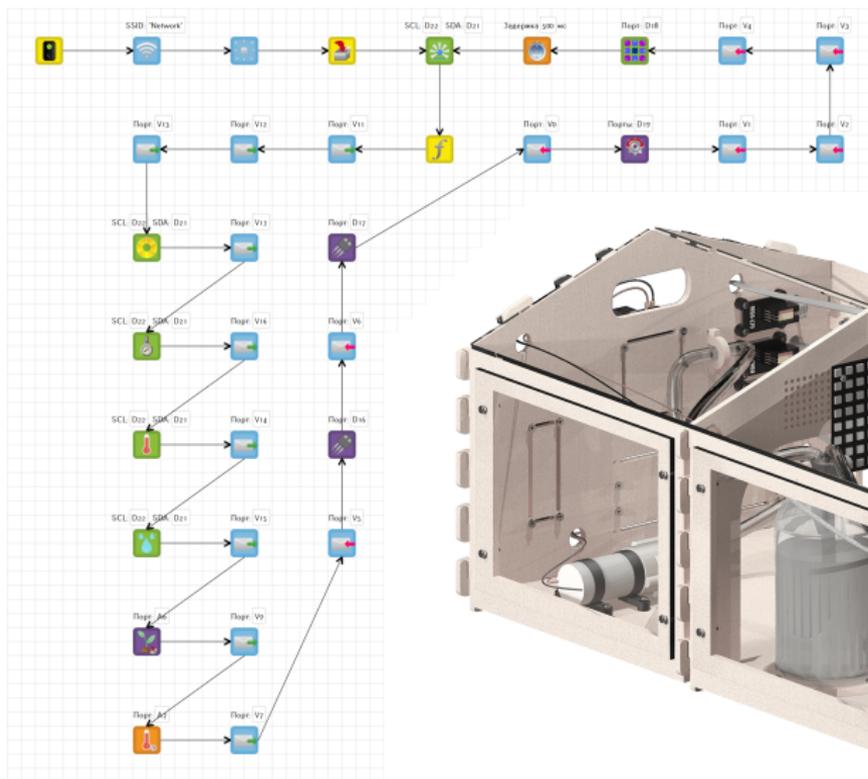
# Полная метамодель



# Архитектура генераторов



# Апробация — проект «Теплица»



Накопленный опыт применения TRIK Studio для конструктора ЙоТик показал необходимость существенной модификации способов построения прошивки контроллера:

- документирование
- локализация ошибок
- простота и атомарность функций
- самостоятельная инициализация и деинициализация
- обработка вызовов из параллельных задач
- корректное завершение и очистка памяти при ошибке

В состав новой прошивки должны входить следующие компоненты:

- виртуальная машина РуСи с возможностью независимого обновления
- сетевой интерфейс для передачи файлов с возможностью полного освобождения памяти при отключении
- файловая система с возможностью управления файлами
- система распределения портов и I2C каналов с реализацией Arduino IDE интерфейса

Проект РуСи развивается уже более 5 лет, в нем приняли участие несколько поколений студентов, придерживающихся различных стилей программирования. Научным руководителем была поставлена задача рефакторинга проекта в целом.

- автоматизация тестирования (Travis CI) и проверки оформления кода (ClangFormat и Clang-Tidy)
- полный переход на систему сборки CMake
- улучшение структуры и исправление ошибок
- разделение репозитория на транслятор и виртуальную машину
- выпуск в формате динамической библиотеки

В 2017 году похожая работа проводилась С.В. Приходько с микроконтроллером STM32. Данная работа отличается в следующих аспектах:

- использование языка РуСи в качестве промежуточного
- более сложная метамодель в связи с более сложными требованиями
- возможность свободного конфигурирования контактов датчиков
- загрузка только кода исполняемой программы (Wi-Fi / USB), а не полной прошивки

- в TRIK Studio для ЙоТик добавлены все требуемые компоненты (репозиторий Victor-Y-Fadeev/qreal)
- выработаны требования к новой прошивке
- проведен рефакторинг проекта РуСи (репозиторий andrey-terekhov/RuC)
- осуществлена апробация на содержательных примерах («Теплица», «Танк», «Робо-футболист»)