

Программирование роботов как путь в информатику

Харюнин Александр Сергеевич
аспирант Института информационных технологий, математики и механики
Нижегородский государственный университет им. Н.И. Лобачевского,
пр. Гагарина, 23, г. Нижний Новгород, 603950, (831)4623361
sharyunin@mail.ru

Кукушкин Даниил Александрович
магистр Института информационных технологий, математики и механики
Нижегородский государственный университет им. Н.И. Лобачевского,
пр. Гагарина, 23, г. Нижний Новгород, 603950, (831)4623361
daniil.kukushkin@list.ru

Борисов Николай Анатольевич
доцент, к.т.н., доцент кафедры Программной инженерии Института информационных технологий, математики и механики
Нижегородский государственный университет им. Н.И. Лобачевского,
пр. Гагарина, 23, г. Нижний Новгород, 603950, (831)4623361
nborisov.itmm@yandex.ru

Аннотация

В работе рассматривается программное обеспечение RobotON Studio – графическая среда программирования, созданное авторами статьи для возможности удобного программирования различных робототехнических платформ, а также для эффективного обучения основам написания кода.

The paper considers RobotON Studio – a graphical programming environment created by the authors of the article for the possibility of convenient programming of various robotic platforms, as well as for effective training in the basics of writing code.

Ключевые слова

Программирование, обучение, робототехника, Arduino, графическая среда программирования, алгоритм

Programming, teaching, robotics, Arduino, graphical programming environment, algorithm

Введение

В связи с широким распространением робототехники среди школьников, большинство детских образовательных учреждений так или иначе включают в свои программы курсы, связанные с конструированием и программированием роботов. Для обеспечения «быстрого входа» обучение проводят на робототехнических конструкторах. Самой распространенной платформой для детской робототехники на сегодня является Lego Mindstorms, вышло уже множество методической литературы про Lego Mindstorms NXT и Lego Mindstorms EV3 [1-9], но у этих конструкторов имеется ряд недостатков. Во-первых, на нем невозможно реализовать действительно разнообразные проекты (из-за ограниченного набора датчиков и малого числа типов моторов), во-вторых, стоимость достаточно внушительная, вследствие чего обычно на один робототехнический конструктор прикрепляется 2-3 ученика.

Поэтому часть учреждений дополнительного образования уже перешли на робототехнический конструктор Arduino [10-12]. Его стоимость является более демократичной, функционал шире, но обучающиеся сталкиваются с необходимостью программировать на языке C и запоминать большое количество синтаксических структур для реализации простейшего алгоритма. Так как необходимо предоставить молодым робототехникам возможность развиваться в этом направлении с самого юного возраста, то представляется целесообразным реализовать удобную графическую среду программирования, в которой создание программы будет аналогично построению алгоритма по блокам. В данной статье представлена графическая среда разработки RobotON Studio, которая позволяет юным робототехникам быстро начать программировать роботов на Arduino.

История создания

Разработка программы началась еще в 2015 г., когда не было адекватных аналогов. Первая бета-версия программы была запущена уже в октябре 2016 г., на момент ее реализации существовали такие аналоги, как *Ardublock* [16-18], обладающий меньшим функционалом и менее дружелюбным интерфейсом и *Scratch4Arduino* [18-20], который не прошивал робота, а только давал команды на исполнение, то есть был ограничен необходимостью постоянной связи робота с ПК.

За последующие несколько лет было создано несколько версий программ, которые включали в себя важные дополнения:

- улучшение навигации (скроллинг и масштабирование программы);
- возможность создания собственных функций;
- подсветка блока, в котором обнаружена ошибка.

На текущий момент данная среда разработки используется для обучения школьников от 6 лет программированию на платформе Arduino.

Функционал ПО RobotON Studio и преимущества использования

Большой интерес графическая среда разработки имела среди юных робототехников, которые ни разу до этого не сталкивались с Arduino, но очень хотели научиться его программировать. В среде RobotON есть возможность концентрироваться не на синтаксисе программы и возможными ошибками на этом уровне, а на разработке алгоритма. Благодаря этому подходу появляется ряд преимуществ:

- Сложность выполняемых задач вырастает гораздо быстрее, т.к. визуально понятно, что происходит в программе;
- Уменьшается количество времени, необходимого для освоения нового материала;
- Становится понятным использование функций;
- Уменьшается время отладки программы.

Так же для удобства обучающихся в правой части экрана программы автоматически вместе с созданием и модификацией алгоритма формируется код на языке C в режиме реального времени. Благодаря этому коду пользователи поймут, что именно происходит при той или иной модификации программы и имеют возможность интерактивно изучать синтаксис языка C. При использовании сторонних и дополнительных библиотек, которые не включены в перечень блоков RobotON Studio, с помощью этого же кода можно в стандартную среду программирования вставить данный код, добавив лишь необходимую библиотеку.

Еще одно очевидное удобство связано с автоматической установкой необходимых драйверов для плат Arduino, в том числе и для ее китайских клонов. Это небольшое преимущество помогает учащимся сразу приступить к написанию программы, не уменьшая интерес.

Плюс ко всему, если робототехник в процессе обучения перейдет с языка C на иной язык программирования, то данную программу легко модифицировать с учетом требований нового языка, будто Python, Java и т.п. На текущий момент с небольшими изменениями уже создана отдельная версия среды разработки для программирования образовательного дрона Nanorix, который так же является разработкой компании RobotON.

Принцип работы RobotON Studio следующий: обучающиеся собирают разноцветные блоки в логическую цепочку, одновременно формируя программный код. Загрузка кода в плату Arduino происходит напрямую из программы. Таким образом, пользователи могут управлять различными устройствами, начиная от светодиодов и заканчивая различными электромоторами и датчиками. С каждым обновлением расширяется база поддерживаемых устройств, которые могут программироваться с помощью среды RobotON Studio.

Примеры

Чтобы показать, как на практике выглядит использование представляемой среды разработки, оценить удобство ее интерфейса и простоту в разработке алгоритмов, мы продемонстрируем это на нескольких примерах.

Пример №1 (простейший)

Создадим программы, в которой сервомотор перемещается влево-вправо. Это можно применить как для робота-манипулятора, так и для патрулирования местности, если к этому мотору прикрепить веб-камеру.

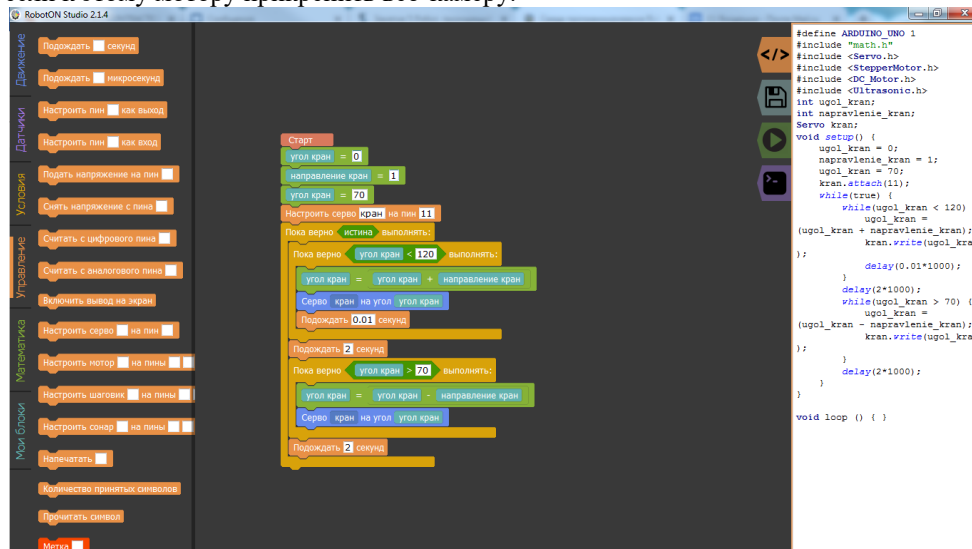


Рис. 1. Реализация управления сервомотором в RobotON Studio

На рисунке 1 представлена реализация простейшей программы. Интерфейс самой среды разработки похож на интерфейс программы Scratch: цвет блока зависит от модуля, которому этот блок принадлежит (синий — движение, оранжевый — управление и т.д.). Справа код, который отображается в реальном времени. Благодаря этой функции у учащихся будет возможность изучать стандартный для Arduino язык

С и отслеживать все изменения после каждого добавления блока. Здесь же реализована возможность использовать русские имена переменных, которые автоматически транслитерируются в основном коде программы. Так же, как и в обычной программе, здесь есть блоки, отвечающие за начальные условия, во вкладке «Математика» есть возможность добавления своих переменных с обозначением их типа данных. После начальных условий идёт бесконечный цикл, в котором мы видим блоки из разных модулей: тут и простейшие математические действия, и условия, и движение сервомотора, и ожидание.

Пример №2 (реализация собственных функций)

Чтобы программа была интуитивно понятней, её, конечно же, удобней разбивать на отдельные функции. Для этого реализована возможность использования и создания новых функций. Рассмотрим это на примере программы для рисования. Чтобы программа была структурированной, нам понадобятся функции, реализовать которые можно в модуле «Мои Блоки». Помимо основной программы, блок которой начинается с команды «Старт», мы видим еще множество разных функций, которые реализованы отдельно и имеют место в основной ветке программы. На вкладке «Мои Блоки» есть возможность добавить еще одну функцию, задать ее возвращаемое значение и указать параметры. Такая возможность структурировать позволяет многократно усложнять решаемые в данном ПО задачи без усложнения отладки. В данном примере все функции линейные, так как рисуются простые объекты. Здесь же есть и блок «Настройка», в котором выставляются все начальные значения. Основная программа начнется после команды «Старт». Именно возможность реализовывать свои функции сделало данное ПО конкурентноспособным.

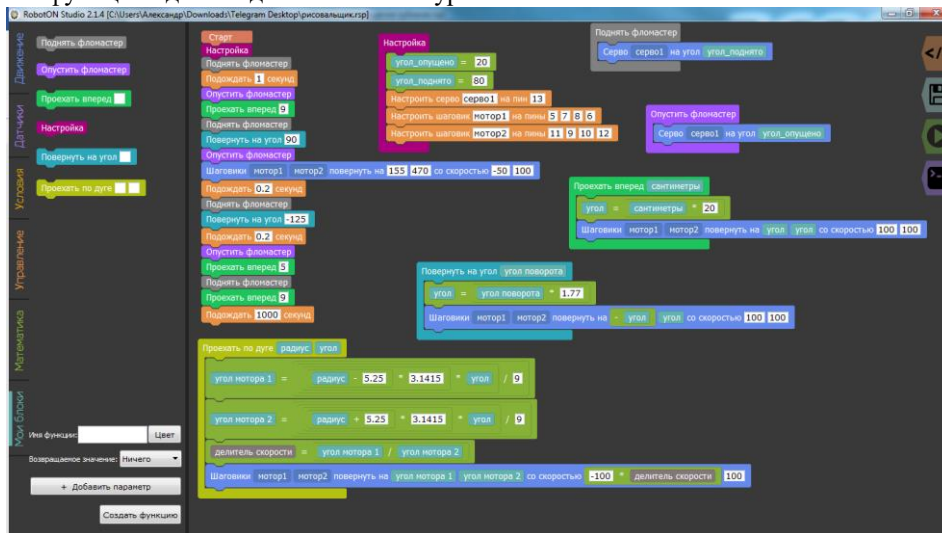


Рис. 2. Реализация программы с функциями

Пример №3.

Решим задачу по перемещению кубиков манипулятором. Здесь надо и полностью прописать все начальные условия, и уточнить все числовые параметры, и прописать, конечно же, сам алгоритм. В данном примере внутри функций уже стали появляться циклы, а также цикл есть и в основном коде программы.

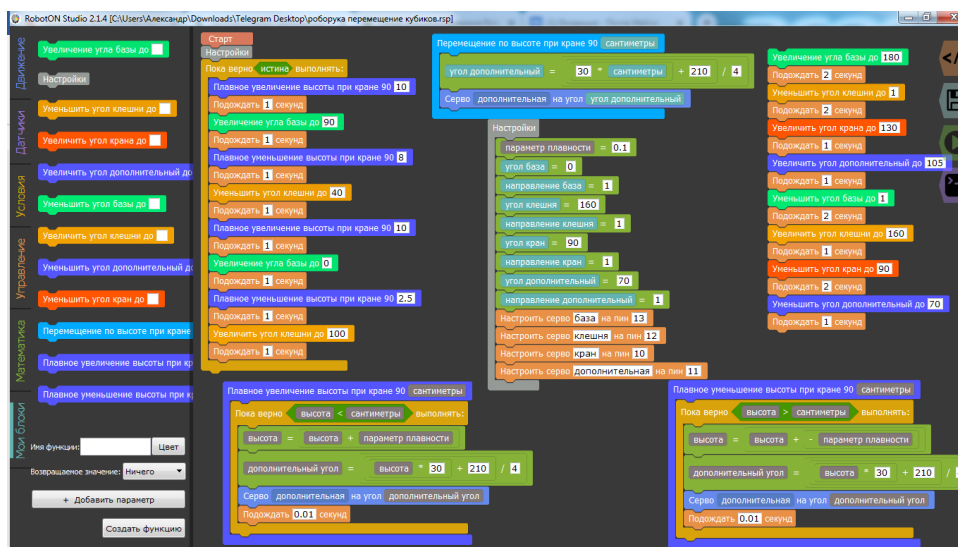


Рис. 3. Реализация программы управления манипулятором

Заключение

Несмотря на возможность использования данного программного обеспечения уже сейчас, многие возможности находятся в процессе разработки. В ближайших обновлениях будет доступна возможность создания проекта из нескольких отдельных программ в разных вкладках. Данный подход удобен для реализации своих идей на нескольких платах Arduino.

Также в среду разработки добавляется возможность следить за всеми функциями проекта благодаря навигационной карте (расположенной в нижнем правом углу экрана при необходимости), возможность удобной отладки: выполняемый блок будет подсвечиваться.

Но и в распространяемой на данном этапе разработки версии RobotON Studio реализовывать свои проекты очень удобно. Работая с данным ПО, появляется возможность работать на языке программирования более высокого уровня, чем Arduino C. Здесь обучаемые при разработке программы будут сталкиваться со сложностью алгоритмов, а не со сложностью синтаксиса.

Здесь можно провести аналогию с развитием языков программирования. От языка Assembler до Pascal и C++. От C++ до C# и Python. Если в Assembler нужно написать целую программу на вывод надписи на экран, то в Pascal это делается в одну строчку. Если нужно нарисовать график в Pascal, то на это уйдет много кода, когда на Python – это опять-таки одна строчка. Так и в нашем случае: интерес учащихся не ослабнет из-за проблем с установкой драйверов или из-за пропущенных точки с запятой или фигурных скобок, из-за которых весь алгоритм воспринимается машиной по-другому. Именно на этом этапе программирование превращается из рутины в творчество.

И не стоит забывать, что навыки программирования, полученные школьниками на занятиях по робототехнике, несомненно, будут востребованы при поступлении в ВУЗы, независимо от выбранного после школы направления обучения и специальности [13-15].

Основы программирования, заложенные в школе, в том числе при обучении робототехнике, позволят будущим студентам с меньшими проблемами овладеть основами математического моделирования [21]. Уверенное владение

программированием как основным инструментом решения различных практических задач с использованием современной вычислительной техники требуют все современные стандарты подготовки специалистов, как федеральные, так и разработанные ведущими вузами страны [22-24]. Современный грамотный специалист в любой области невозможен без владения информационными технологиями, реализуемыми через программирование.

Литература

1. Программирование Lego EV3 Mindstorms - [Электронный ресурс] – URL - <https://legoteacher.ru/robototexnika-dlya-nachinayushhix/ev3-programmirovanie.html>, дата обращения – 29.11.2019
2. Овсяницкая, Л.Ю. Курс программирования робота Lego Mindstorms EV3 в среде EV3 / Л.Ю. Овсяницкая, Д.Н. Овсяницкий, А.Д. Овсяницкий. – М.: «Перо», 2016. – 296 с.
3. Самоучитель для учащихся или методическое пособие для преподавателей применения Lego ® Mindstorms ® EV3 в предметных дисциплинах средней школы [Электронный ресурс]. — URL - <https://mirrobo.ru/pilot/metodicheskie-posobija-dlja-prepodavat/> (дата обращения: 29.11.2019)
4. Lego: больше чем конструктор [Электронный ресурс]. — Режим доступа: <http://edurobots.ru/2014/11/lego-bolshe-chem-konstruktor/> (дата обращения: 14.06.2018).
5. Добриборщ Д.Э., Артемов К.А., Чепинский С.А. - Основы робототехники на Lego® Mindstorms® EV3. Учебное пособие – М.:Лань, 2019 г. - 108 с.
6. Програмируем робота LEGO Mindstorms EV3 на Python - [Электронный ресурс]. — Режим доступа: <http://www.proghouse.ru/article-box/117-ev3-python> (дата обращения: 29.11.2019).
7. LEGO® MINDSTORMS® Education EV3: книги учителя - [Электронный ресурс]. — Режим доступа: <https://education.lego.com/ru-ru/support/mindstorms-ev3/teacher-introduction> (дата обращения: 29.11.2019).
8. Овсяницкая, Л.Ю. Алгоритмы и программы движения по линии робота Lego Mindstorms EV3 / Л.Ю. Овсяницкая, Д.Н. Овсяницкий, А.Д. Овсяницкий. – М.: «Перо», 2015. – 168 с.
9. Штадлер А. Моя книга о LEGO® EV3. Построить собственного робота и создать для него программу с конструктором LEGO® MINDSTORMS® - М.: Фолиант, 2017. – 288 с.
10. Джереми Блум. Изучаем Arduino – СПб.: BHV, 2015 – 336 с.
11. Михаил Момот. Мобильные роботы на базе Arduino – СПб: БХВ-Петербург, 2017. – 288 с.
12. Петин В.А., Биняковский А.А. Практическая энциклопедия Arduino. – М.: ДМК Пресс, 2017 – 152 с.
13. Борисов Н. А. Изучение различных технологий программирования на примере курса «Программирование роботов» - XIV Всероссийская конференция «Преподавание информационных технологий в Российской Федерации», 19.05.2016 - 20.05.2016, Санкт-Петербург, СПбГУ – [Электронный ресурс] – URL - <http://it-education.ru/2016/section/230/96354/index.html>, дата обращения – 29.11.2019.
14. Борисов Н.А. Организация сквозной практики по программированию на основе предметной области «Робототехника» // Преподавание информационных технологий в Российской Федерации: Материалы Тринадцатой открытой Всероссийской конференции – Пермь: ПГНИУ, 2015. – с. 94-96.

15. Борисов Н.А. С информатикой, через робототехнику – в университет! // Преподавание информационных технологий в Российской Федерации: Материалы Двенадцатой открытой Всероссийской конференции – Казань: КФУ, 2014. – с. 265-267.
16. Виталий Петров. ArduBlock. Дистанционное управление. – М.: AB Technologies, 2019 – 28 с.
17. Ardublock для Arduino – графический язык программирования/ - [Электронный ресурс] – URL - <https://arduinomaster.ru/program/visual-ardublock-arduino/>, дата обращения – 29.11.2019.
18. ScratchDuino. Робоплатформа и ArduBlock - [Электронный ресурс] – URL - <https://www.lab169.ru/материалы/робототехника-и-конструирование/> scratchduino-робоплатформа-и-ardublock/, дата обращения – 29.11.2019.
19. Поляков К.Ю., Винницкий Ю.А. Конструируем роботов на ScratchDuino. Первые шаги. – М.: Лаборатория знаний, 2016. - 119 с.
20. СкретчДуино.Робоплатформа. [Электронный ресурс] – URL - <http://robbo.ru/products/robotkit/>, дата обращения – 29.11.2019
21. Солдатенко И.С., Сорокин С.В., Захарова И.В., Медведева О.Н., Кузенков О.А. Об общеразвивающей программе «Основы математического инженерного моделирования» // Инженерное образование. 2017. № 21. С. 104-109.
22. Zakharova I., Kuzenkov O. Experience in implementing the requirements of the educational and professional standards in the field of ICT in Russian education //В сборнике: CEUR Workshop Proceedings Selected Papers of the 11th International Scientific-Practical Conference Modern Information Technologies and IT-Education, SITITO 2016. 2016. С. 17-31.
23. Захарова И.В., Кузенков О.А. Опыт актуализации образовательных стандартов высшего образования в области ИКТ//Современные информационные технологии и ИТ-образование. 2017. Т. 13. № 4. С. 46-57.
24. Гугина Е.В., Кузенков О.А. Образовательные стандарты Нижегородского государственного университета им. Н.И. Лобачевского // Вестник Нижегородского университета им. Н.И. Лобачевского. 2014. № 3-4. С. 39-44.