



**Министерство образования и науки Российской
Федерации
Федеральное государственное бюджетное
образовательное
учреждение высшего образования
«Московский государственный технический
университет имени Н.Э. Баумана»
(национальный исследовательский университет)
(МГТУ им. Н. Э. Баумана)**



Утверждаю
Первый проректор-
проректор по учебной работе

Б.В. Падалкин
2017г.

Методическое пособие

Робототехника в школе

Автор: старший преподаватель кафедры СМ-7 «Робототехнические
системы и мехатроника» Зеленцов В.В.

МГТУ им. Н.Э. Баумана
2017г.

ОГЛАВЛЕНИЕ

	Стр.
Введение	3
1. Общая характеристика платформы LEGO® MINDSTORMS® Education EV3	6
2. Программное обеспечение для работы с конструктором LEGO® MINDSTORMS® Education EV3	9
2.1 LEGO Digital Designer.....	9
2.2 Программное обеспечение LEGO® MINDSTORMS® Education EV3	11
2.3 Альтернативные средства программирования роботов на платформе LEGO® MINDSTORMS® Education EV3.....	15
3. Управление движением мобильных роботов на платформе LEGO® MINDSTORMS® Education EV3	16
3.1 Выбор конструкции робота	16
3.2 Кинематика мобильного робота с дифференциальным приводом	17
3.3 Работа с конструктором LEGO® MINDSTORMS® Education EV3 на примере решения демонстрационной задачи	19
Литература.....	26

ВВЕДЕНИЕ

За последние несколько десятилетий роботы стали неотъемлемой частью жизни и деятельности человечества. Несмотря на то, что единого понятия робота не существует, обычно под роботом понимается многофункциональное устройство, выполняющее за человека его работу. Изучением и разработкой робототехнических систем занимается прикладная наука робототехника, опирающаяся на такие дисциплины, как теория управления, механика, электроника, электротехника, информатика и др. Как правило, робот представляет собой мехатронное устройство, в котором механическая и электронная часть взаимно дополняют друг друга и не могут функционировать отдельно, таким образом от инженера - разработчика роботов требуется уметь разрабатывать конструкцию робота, его электронную часть, алгоритмы и программное обеспечение для управления роботом.

Роботы классифицируются по различным признакам. По функциональному назначению различают следующие классы роботов:

- промышленные;
- сервисные;
- медицинские;
- специальные (подводные, космические, военные, роботы для обеспечения правопорядка, роботы для радиационно и химически опасных сред, пожарные роботы);
- учебные.

По признаку подвижности различают стационарные и мобильные роботы. По принципу управления роботы подразделяются на автономные, работающие самостоятельно без участия человека, и управляемые человеком, подразумевающие в той или иной мере участие оператора или пользователя в процессе управления роботом.

Схема, иллюстрирующая принцип функционирования роботов представлена на рисунке 1.

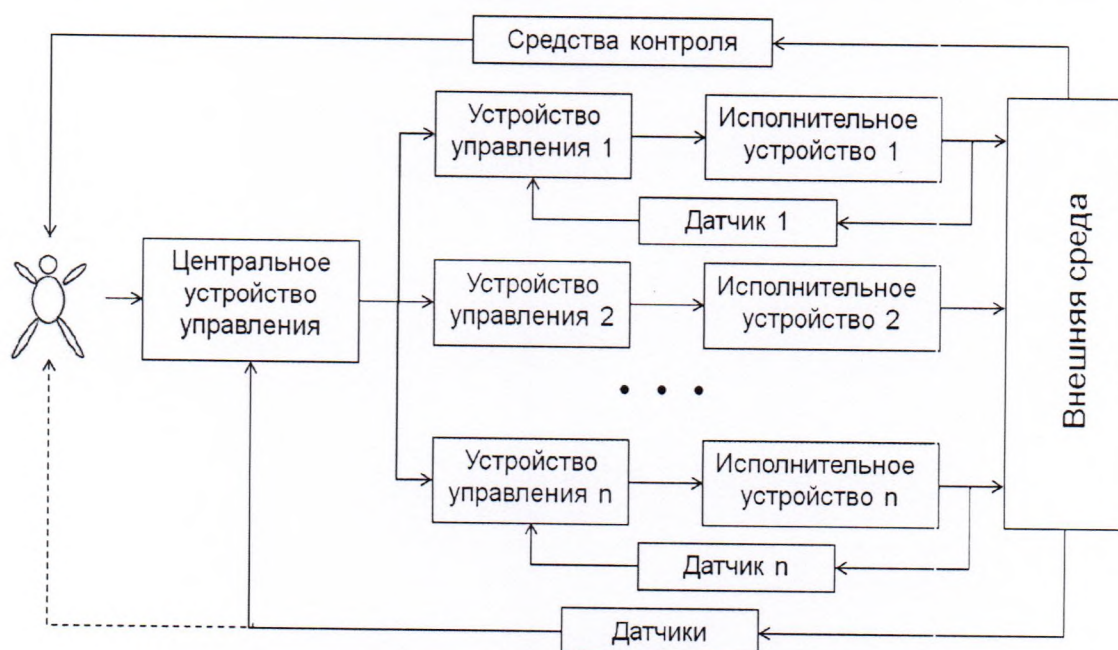


Рисунок 1

Робот состоит из систем, осуществляющих воздействие на внешнюю среду, в которой функционирует робот, посредством исполнительных устройств, управляющие воздействия для которых формируются входящими в состав систем устройствами управления. При этом управление может осуществляться по принципу обратной связи, когда устройство управления получает информацию о фактическом значении управляемого параметра от включенного в цепь обратной связи информационного устройства (датчика), вычисляет ошибку системы, представляющую собой разность заданного и фактического значения управляемого параметра и формирует на основе этого значения управляющее воздействие на исполнительное устройство, изменяя таким образом фактическое значение управляемого параметра до тех пор, пока оно не станет равным заданному. Аналогичным образом центральное устройство управления формирует управляющие воздействия на системы робота на основании информации от датчиков, характеризующей параметры внешней среды и находящихся в ней объектов, с которыми взаимодействует робот. Если робот является автономным, то он управляется исключительно центральным устройством управления, в иных случаях управляющие воздействия на центральное устройство управления в том или ином объеме формируются человеком-оператором или пользователем, получающим информацию о состоянии внешней среды и робота через средства контроля, а также от информационных устройств робота.

Использование реальных робототехнических систем для изучения основ робототехники в школах затруднительно вследствие высокой стоимости данного вида техники, что явилось предпосылкой для возникновения класса учебных роботов, предназначенного для изучения основ робототехники и обладающего сравнительно низкой стоимостью и простотой в освоении самостоятельно учащимся или в рамках занятий с

преподавателем. Роботы этого класса по сути представляют собой действующие модели более серьезных робототехнических систем, позволяющие реализовать все этапы создания робота и изучить основные принципы построения и функционирования роботов.

Существует два подхода к выбору материальной базы для проведения занятий по робототехнике в школе. Первый подразумевает использование выпускаемых промышленностью конструкторов для сборки учебных роботов, второй - использование имеющихся на рынке одноплатных контроллеров, датчиков, приводов, источников питания, элементов конструкции, а также самостоятельную разработку и изготовление электронных и механических элементов, входящих в состав мобильных роботов.

Достоинством первого подхода является то, что учащийся и преподаватель получают в свое распоряжение заранее отработанные аппаратные и программные решения "из коробки" для создания на их основе учебных роботов, что дает возможность сфокусироваться на принципах построения и управления робототехнических систем, не заботясь об изготовлении и отработке самих компонентов и отладке системного программного обеспечения. К недостаткам данного подхода можно отнести более высокую в большинстве случаев стоимость по сравнению со вторым подходом и некоторую функциональную ограниченность наборов-конструкторов. Таким образом, данный подход можно признать предпочтительным для начального обучения основам робототехники. В [1] приведен обзор имеющихся на рынке наборов-конструкторов для сборки учебных роботов.

Достоинством второго подхода является большая ценовая доступность (при создании простых роботов) и возможность обеспечить большую функциональность создаваемых роботов за счет подбора или изготовления компонентов, ориентированных на решение конкретных задач. При таком подходе максимально близко воспроизводится процесс создания реальных робототехнических систем, однако он требует наличия у учащегося и преподавателя более высокого уровня знаний и навыков в области механики, электроники и программирования, чем при первом подходе, поэтому его можно рекомендовать для углубленного изучения основ робототехники и создания прототипов роботов.

В данном пособии рассматривается построение и программирование учебных роботов на базе широко распространенного в московских школах конструктора LEGO® MINDSTORMS® Education EV3. Пособие предназначено для пользователей, не имеющих опыта работы с данным конструктором и желающих его освоить и применять в рамках программ основного и дополнительного образования, реализуемых в проекте "Инженерный класс в московской школе".

1. ОБЩАЯ ХАРАКТЕРИСТИКА ПЛАТФОРМЫ LEGO® MINDSTORMS® EDUCATION EV3

Конструктор LEGO® MINDSTORMS® Education EV3 принадлежит к третьему поколению платформы LEGO® MINDSTORMS® и характеризуется значительным увеличением по сравнению с предыдущим поколением (LEGO® MINDSTORMS® NXT) вычислительной мощности электронного блока управления и расширением номенклатуры информационных устройств (в состав набора введен гироскопический датчик). Состав, характеристики и основные принципы работы с набором освещены в руководстве пользователя [2], поэтому остановимся на некоторых особенностях, не задокументированных в руководстве, но полезных для понимания работы набора.

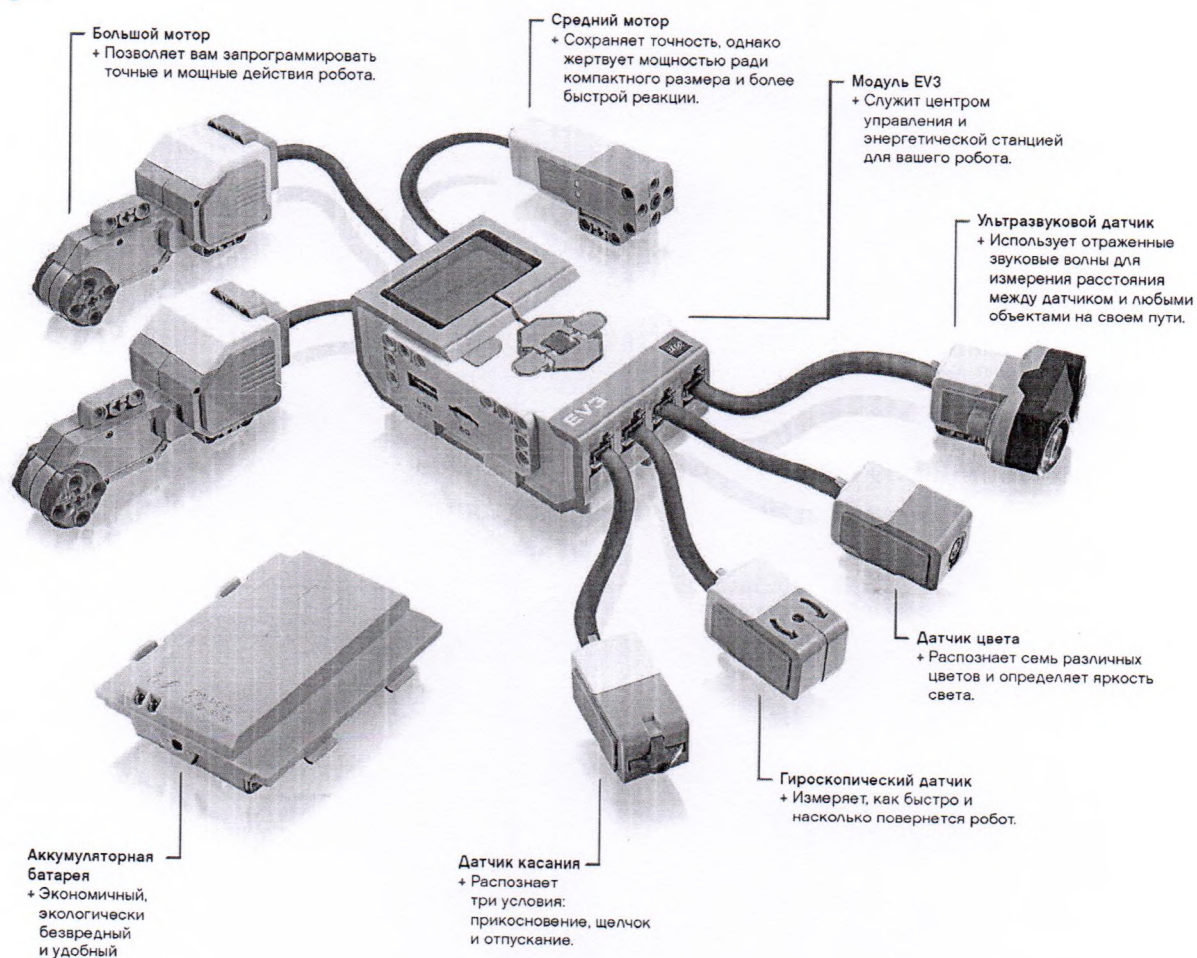


Рисунок 2

На рисунке 2 приведены электрические и электронные компоненты конструктора. Исполнительные устройства представлены 2 большими и 1 средним моторами (далее будем пользоваться терминологией LEGO, принимая во внимание ниже приведенные особенности). "Моторы" представляют собой мотор-редукторы на основе электродвигателей постоянного тока со встроенным датчиком положения импульсного типа, установленным на валу двигателя. В конструкции большого мотора (момент 0,20 Нм) применен 4-х ступенчатый цилиндрический редуктор с люфтом

выходного вала до 3° , что делает нежелательным использование данного привода для точного позиционирования исполнительных органов робота. Для этих целей предпочтительней использовать средний мотор (момент 0,08 Нм), имеющий в своем составе двухступенчатый планетарный редуктор практически без люфта.

Информационные устройства представлены ультразвуковым датчиком, датчиком цвета, гироскопическим датчиком и двумя датчиками касания.

Ультразвуковой датчик представляет собой ультразвуковой дальномер, измеряющий расстояние до находящегося перед ним объекта методом эхолокации, что делает его чувствительным к звукоотражающим свойствам объектов измерений, также у датчика широкая (до 40°) в горизонтальной плоскости диаграмма направленности и это необходимо учитывать при работе с ним. Также следует учитывать недостоверность измерений при расстоянии до объекта менее 3 см.

Датчик цвета представляет собой электронно-оптический датчик на основе трехцветного фотодиода, измеряющего интенсивности 3х цветовых компонентов падающего на него света. Датчик может работать в режиме распознавания цвета (7 настраиваемых цветов), измерения интенсивности падающего света и измерения интенсивности отраженного света, для чего имеет в своем составе светодиодный источник света.

Гироскопический датчик не использует гироскоп, а построен на основе микроэлектромеханического датчика угловых скоростей, использующего для измерений эффект Кориолиса. Следует учитывать, что первичным параметром, измеряемым датчиком, является угловая скорость его вращения в плоскости нанесенной маркировки, а угол поворота вычисляется электронным блоком путем численного интегрирования угловой скорости, таким образом показания датчика по углу сильно зависят от начальных условий интегрирования. В связи с этим, при использовании датчика в режиме измерения угла важно, чтобы он находился в покое при включении электронного блока управления.

Датчик касания представляет собой концевой выключатель, дискретно изменяющий состояние выходного сигнала при нажатии на шток датчика. Аналогичным образом можно использовать в качестве датчиков кнопки управления электронным блоком EV3. Кроме того для использования в процессе управления доступна информация с датчиков угла поворота выходного вала мотор-редукторов.

Электронный блок управления EV3 совмещает функции центрального устройства управления, устройств управления приводными системами робота и интерфейса оператора. Из особенностей блока следует отметить наличие интерфейса для карты памяти стандарта microSD, что в сочетании с оперативным запоминающим устройством объемом 64 МБ позволяет загружать в блок альтернативные операционные системы, например leJOS [3] и ev3dev [4], с сохранением штатного встроенного программного

обеспечения блока. Использование данной возможности позволяет существенно расширить диапазон применяемых для программирования роботов средств в дополнение к штатной системе программирования EV3-G.

Конструкция роботов собирается из стандартных конструкционных элементов LEGO Technics, не требующих использования для сборки дополнительного инструмента. Базовый набор (индекс производителя 45544) позволяет в дополнение к несущей конструкции реализовать колесное, гусеничное шасси, зубчатые и червячные передачи. Кроме базового набора существует ресурсный набор (индекс производителя 45560), содержащий дополнительные конструкционные элементы и позволяющий реализовать помимо остального передачи с зубчатой рейкой, карданные передачи, ременные передачи и передачи с гибким валом.

Таким образом, у роботов, создаваемых на базе конструктора LEGO® MINDSTORMS® Education EV3 присутствуют все элементы, характерные для реальных робототехнических систем: конструкция, устройство управления, исполнительные органы и информационные устройства, что позволяет успешно использовать конструктор для изучения основ робототехники.

2. ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ ДЛЯ РАБОТЫ С КОНСТРУКТОРОМ LEGO® MINDSTORMS® EDUCATION EV3

2.1 LEGO Digital Designer

Подобно тому, как при разработке реальных робототехнических систем используются системы автоматизированного проектирования, для разработки конструкции роботов из наборов LEGO используется программный продукт LEGO Digital Designer (LDD)[5], который позволяет осуществлять сборку виртуальных моделей роботов из доступных в системе LEGO Mindstorms элементов, производить их визуализацию и создавать в автоматическом режиме инструкции по сборке роботов со списками необходимых компонентов.

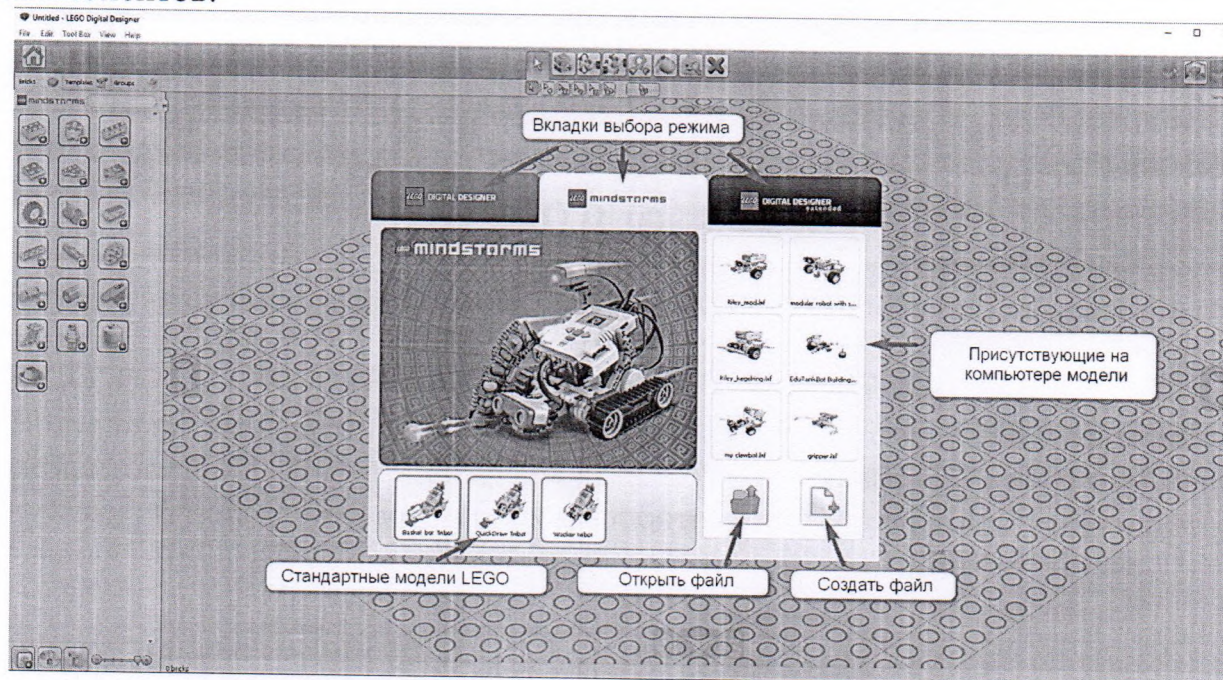


Рисунок 3

На рисунке 3 приведен входной экран LDD с элементами управления, позволяющими выбрать режим работы программы (для разработки роботов используется режим LEGO MINDSTORMS), загружать существующие модели и создавать новые. Основной экран LDD позволяет выполнять сборку модели, и переходить в режимы визуализации и инструкции по сборке (рисунок 4). Компоненты модели выбираются из палитры компонентов курсором мыши и размещаются в рабочем пространстве экрана. Изменение ориентации элементов производится кнопками мыши, выравнивание и стыковка элементов выполняются автоматически, за исключением элементов, требующих поворота на углы не кратные 90° (инструменты "Шарнир", "Выровнять шарнир").

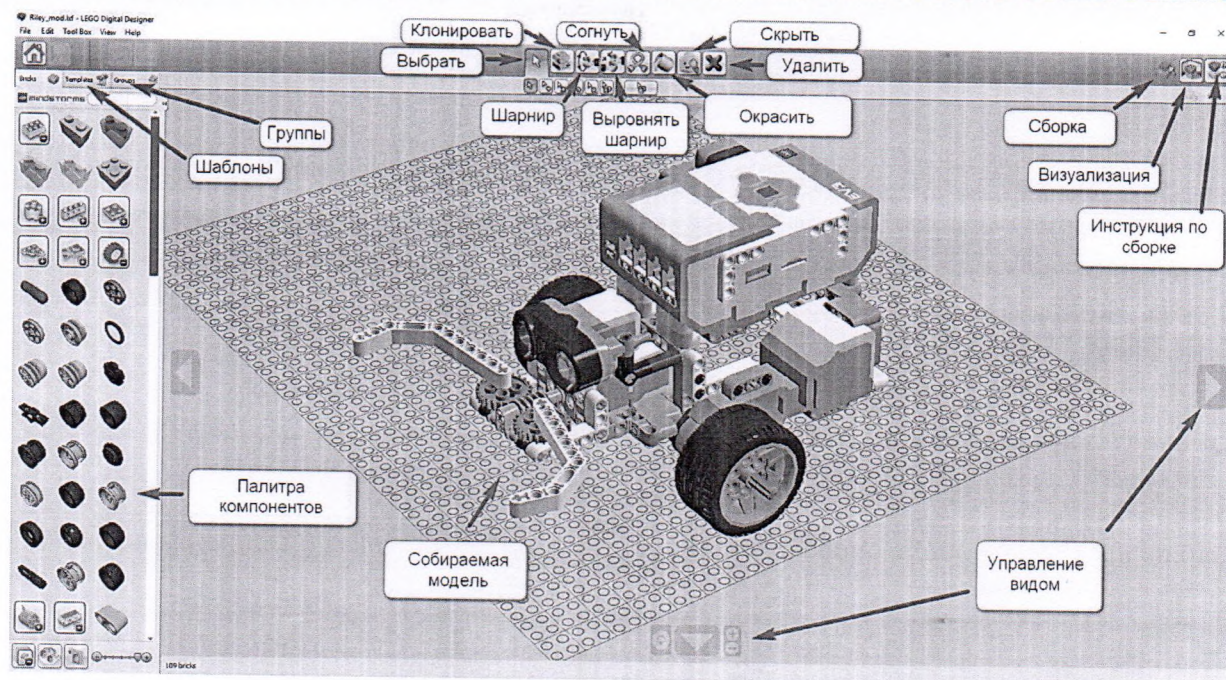


Рисунок 4

Основной экран приложения в режиме "Инструкция по сборке" представлен на рисунке 5. В данном режиме происходит автоматическое построение последовательности сборки с последующим пошаговым отображением на экране и возможностью создания HTML файла с инструкцией по сборке и списком компонентов, что позволяет собирать робота без использования LDD. К достоинствам LDD следует отнести его интуитивность и простоту в освоении, к недостаткам - невозможность редактирования автоматически создаваемой последовательности сборки. Данную проблему можно решить использованием программного продукта Blueprint [6], позволяющего вручную создавать сборочные инструкции на базе файлов LDD. Кроме того, существуют программные продукты SR 3D Builder [7] и MLCAD [8], обладающие большей гибкостью и более широким набором функциональных возможностей, чем LDD, но более сложные в освоении.

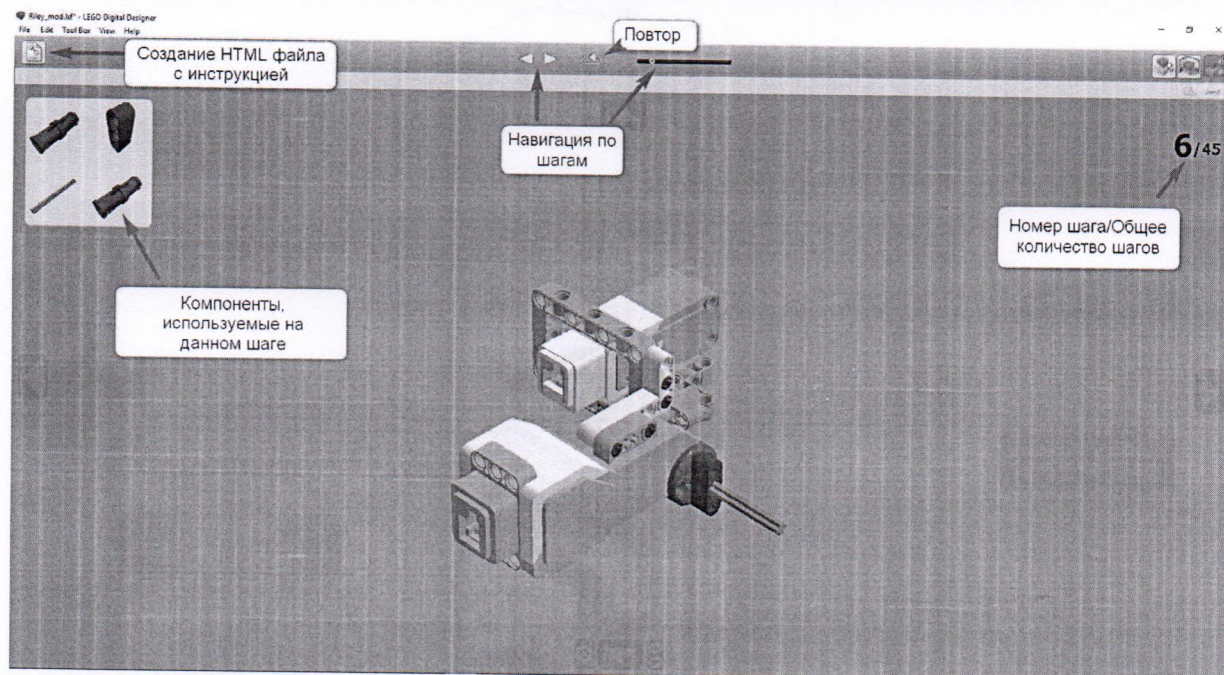


Рисунок 5

2.2 Программное обеспечение LEGO® MINDSTORMS® Education EV3

Программное обеспечение LEGO® MINDSTORMS® Education EV3 является штатным средством программирования роботов на основе данного конструктора, кроме этого оно играет роль обучающей среды и средства документирования процесса выполнения процесса, обработки и документирования полученных в ходе выполнения проекта данных.

После загрузки программы пользователь получает доступ ко входному экрану ("Лобби"), содержащему элементы доступа к различным справочным материалам, включая инструкции по сборки стандартных моделей конструктора, руководству пользователя, самоучителю по программированию роботов в среде EV3, а также элементам управления, позволяющим открывать существующие и создавать новые файлы проектов программного обеспечения роботов. При этом справочные и учебные материалы отображаются в области деятельности входного экрана, а проекты создаются в открывающейся вкладке, помеченной названием файла проекта, содержащим среду программирования EV3. Вид входного окна с элементами управления приведен на рисунке 6.

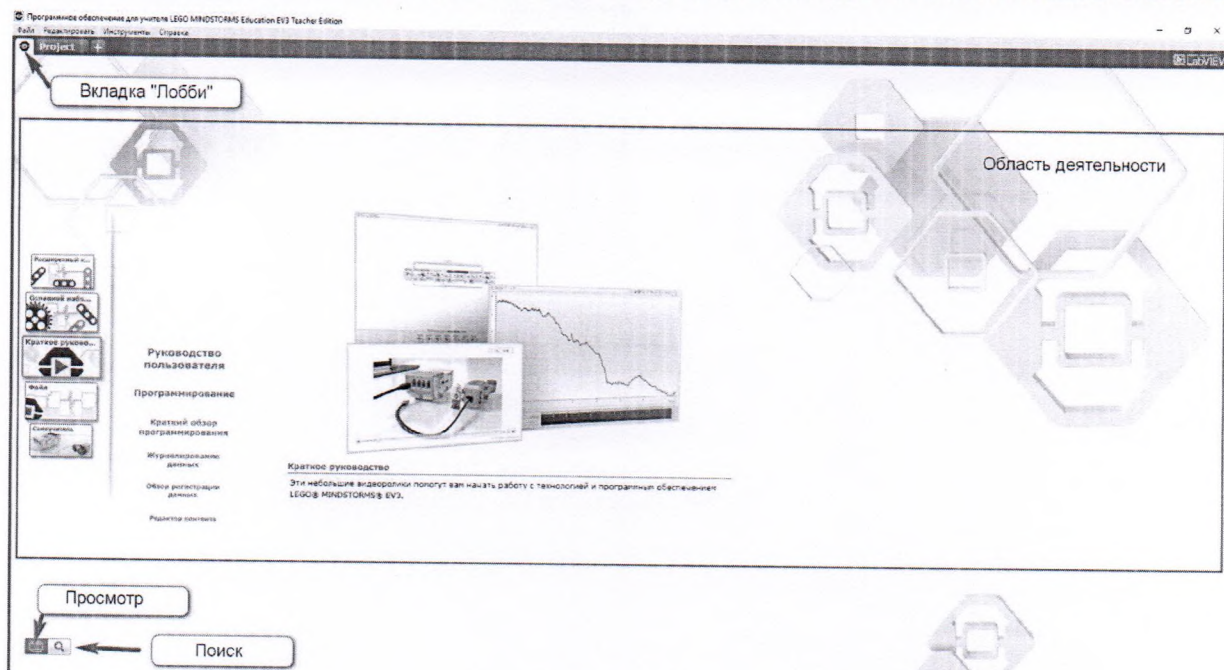


Рисунок 6

Файл проекта содержит все используемые в проекте программы, переменные, пользовательские блоки и загружаемые в электронный блок управления файлы изображений и звуков. Данная информация, а также введенное пользователем описание проекта доступны во вкладке "Свойства проекта", обозначенной пиктограммой с изображением гаечного ключа. Пример отображаемой во вкладке "Свойства проекта" информации приведен на рисунке 7.

Разработчиками системы программирования EV3 использован принцип визуального программирования, когда вместо написания текста программы она создается путем манипулирования графическими объектами. В роли таких объектов в данном случае выступают программные блоки, каждый из которых выполняет то или иное действие. Каждый блок имеет несколько режимов работы, в зависимости от режима может меняться набор входных и выходных параметров блока. Программа формируется путем организации цепи блоков в соответствии с задуманным алгоритмом, при этом возможно одновременное выполнение нескольких программных цепей, то есть организация параллельных вычислений.

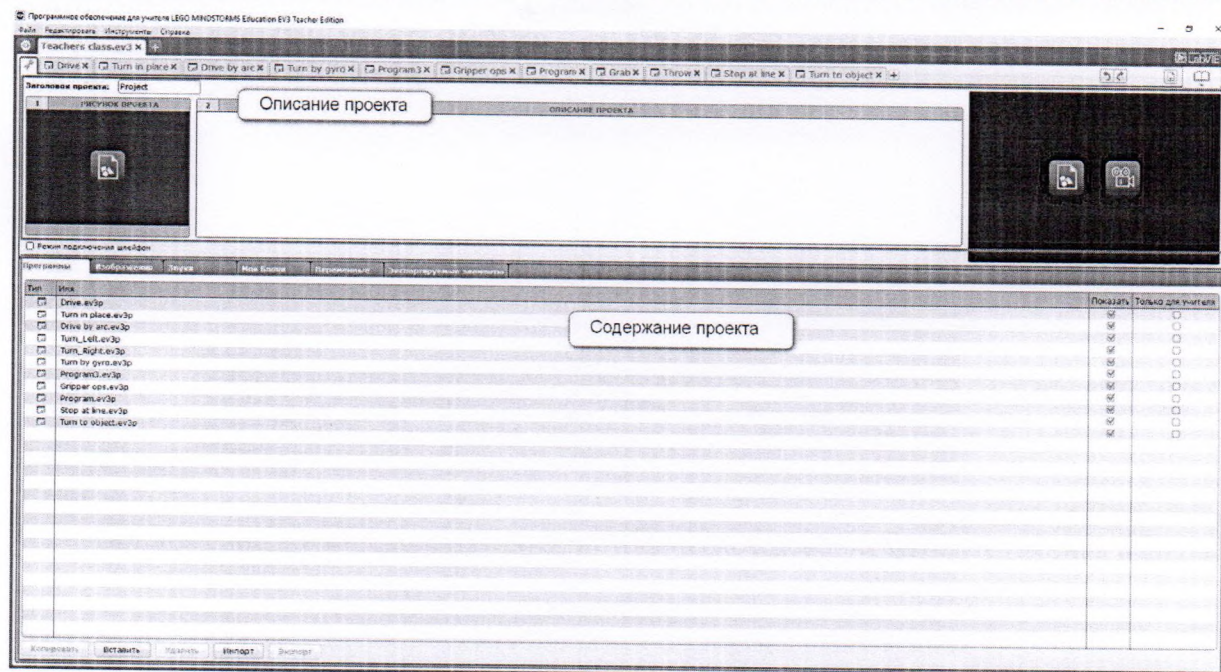


Рисунок 7

Блоки организованы в тематических палитрах в нижней части окна среды программирования (рисунок 8) следующим образом:

Палитра блоков действия (зеленая) содержит блоки "Средний мотор", "Большой мотор", "Рулевое управление", "Независимое управление моторами", "Экран", "Звук", "Индикатор состояния модуля" (рисунок 9, слева направо).

Палитра блоков-операторов (оранжевая) содержит блоки "Начало", "Ожидание", "Цикл", "Переключение", "Прерывание цикла" (рисунок 10, слева направо).

Палитра блоков датчиков (желтая) содержит блоки "Кнопки управления модулем", "Датчик цвета", "Гирокоспический датчик", "Инфракрасный датчик", "Вращение мотора", "Датчик температуры", "Таймер", "Датчик касания", "Ультразвуковой датчик", "Счетчик электроэнергии", "Датчик звука NXT" (рисунок 11, слева направо).

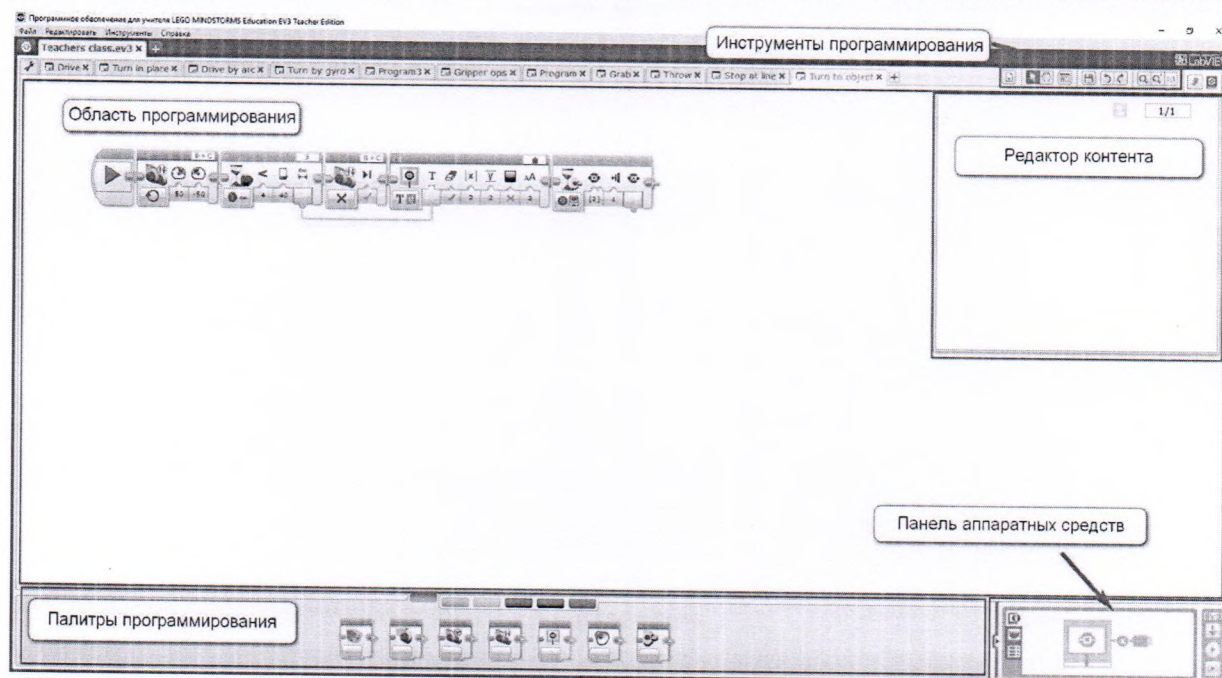


Рисунок 8

Палитра блоков данных (красная) содержит блоки "Переменная", "Константа", "Операции над массивом", "Логические операции", "Математика", "Округление", "Сравнение", "Диапазон", "Текст", "Случайное значение" (рисунок 12, слева направо).

Палитра расширенных блоков (синяя) содержит блоки "Доступ к файлу", "Журналирование данных", "Обмен сообщениями", "Подключение Bluetooth", "Поддерживать в активном состоянии", "Необработанный значение датчика", "Нерегулируемый мотор", "Инвертировать вращение мотора", "Остановить программу" (рисунок 13, слева направо).

Палитра "Мои блоки" (голубая) содержит пользовательские блоки, создаваемые из часто используемых сегментов программ. Пользовательские блоки являются аналогами процедур и функций в текстовых языках программирования.

Детальная информация о блоках, режимах их работы, входных и выходных параметрах содержится в справочной системе программного обеспечения EV3 (пункт "Справка" главного меню). Использование ряда блоков будет рассмотрено в дальнейшем на примере конкретных задач управления учебными мобильными роботами.

Помимо палитр программирования среда программирования EV3 содержит панель аппаратных средств, позволяющую подключать к компьютеру и отключать от него электронные блоки EV3, загружать в электронный блок разработанные программы, управлять содержимым флэш-памяти электронного блока, контролировать состояние подключенных к блоку исполнительных и информационных устройств и управлять ими.

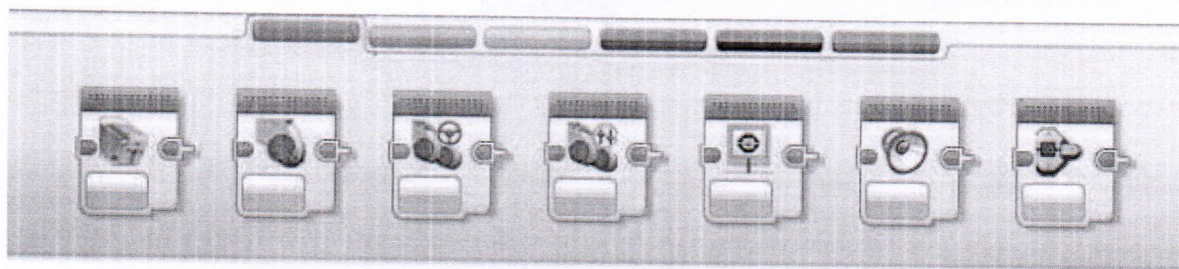


Рисунок 9

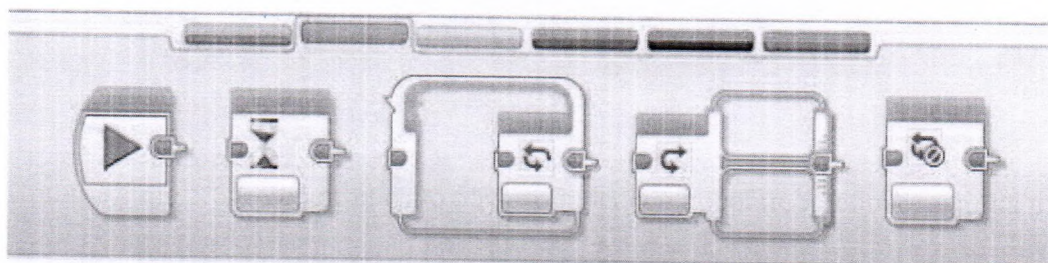


Рисунок 10

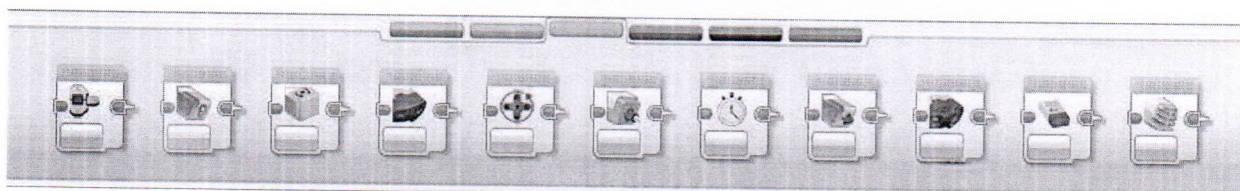


Рисунок 11

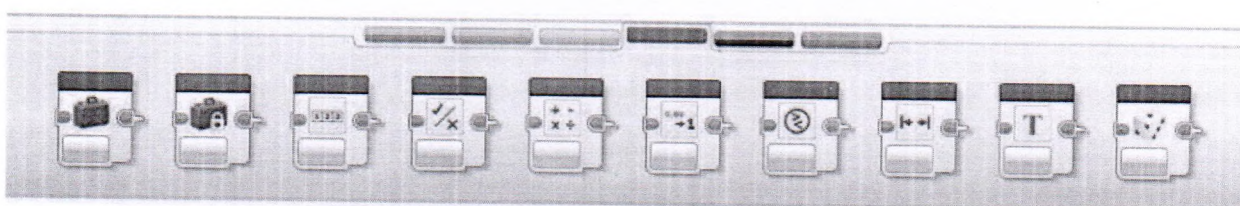


Рисунок 12

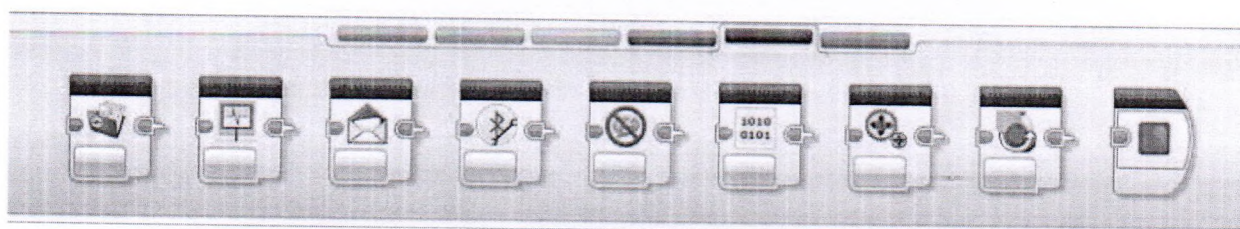


Рисунок 13

2.3 Альтернативные средства программирования роботов на платформе LEGO® MINDSTORMS® Education EV3

Кроме штатного программного обеспечения LEGO® MINDSTORMS® Education EV3 существует ряд программных продуктов (в большинстве своем свободно распространяемых), позволяющий создавать программы для роботов с использованием популярных текстовых и визуальных языков программирования.

EV3 Basic [9] представляет собой расширение пакета Microsoft Small Basic для работы с роботами LEGO EV3, позволяет использовать текстовый

язык программирования Basic, легко интегрируется с электронным блоком управления без изменения его встроенного программного обеспечения. Аналогично штатной среде программирования, программы загружаются в память электронного блока, видны в его файловом браузере и запускаются как с компьютера, так и с самого блока. Одним из преимуществ данного продукта также является наличие у него русской локализации.

Ранее упоминавшийся продукт leJOS [3] представляет собой миниатюрную виртуальную машину Java, загружается в оперативную память электронного блока с флэш-карты microSD и позволяет программировать роботы на популярном объектно-ориентированном языке Java. Требуется организации сетевого подключения между электронным блоком и компьютером.

Ev3dev представляет собой операционную систему на базе Debian Linux, загружаемую с флэш-карты microSD и позволяющую использовать для программирования робота Python, JavaScript, Java, Go, C++, C, Vala и Genie. Использование данного продукта требует наличия опыта использования операционной системы Linux и организации сетевого подключения по протоколу TCP/IP между компьютером и электронным блоком управления EV3.

Существует расширение набирающего популярность визуального языка программирования Scratch для работы с роботами EV3. Установка и использование данного продукта подробно освещены в [10].

Open Roberta Lab [11] представляет собой облачную среду программирования роботов Lego Mindstorms EV3, доступную через интернет из браузера. Электронный блок управления, загружаемый с флэш-карты, должен быть подключен к интернету.

Использование C-подобного языка программирования ROBOTC (коммерческий продукт) [12] на платформе Lego Mindstorms EV3 требует замены встроенного программного обеспечения электронного блока, что приводит к невозможности пользоваться штатной средой программирования.

Возможность программирования роботов Lego Mindstorms EV3 с использованием стандартного встроенного программного обеспечения электронного блока управления также присутствует у широко распространенных пакетов моделирования LabVIEW и MATLAB.

3. УПРАВЛЕНИЕ ДВИЖЕНИЕМ МОБИЛЬНЫХ РОБОТОВ НА ПЛАТФОРМЕ LEGO® MINDSTORMS® EDUCATION EV3

3.1 Выбор конструкции робота

Ввиду наличия в конструкторе всего трех приводов наиболее популярным видом роботов, реализуемым на платформе LEGO MINDSTORMS Education EV3 являются мобильные роботы с дифференциальным приводом. Робот данного типа имеет два привода, по одному на каждый бортовой движитель (колесо или гусеницу), изменение направления и скорости движения осуществляется за счет разницы скоростей

и перемещений движителей. Существует разработанная LEGO стандартная конструкция такого робота под названием "Приводная платформа", однако ее сборка занимает много времени и использование такого робота для школьных занятий нецелесообразно. Оптимальным видится использование более простых конструкций, например робота RaileyRover [13] разработки Дэмиена Ки (рисунок 14). Сборка занимает около 20 минут, конструкция позволяет удобно расположить соединительные кабели и предоставляет много возможностей для установки датчиков и дополнительного оборудования.

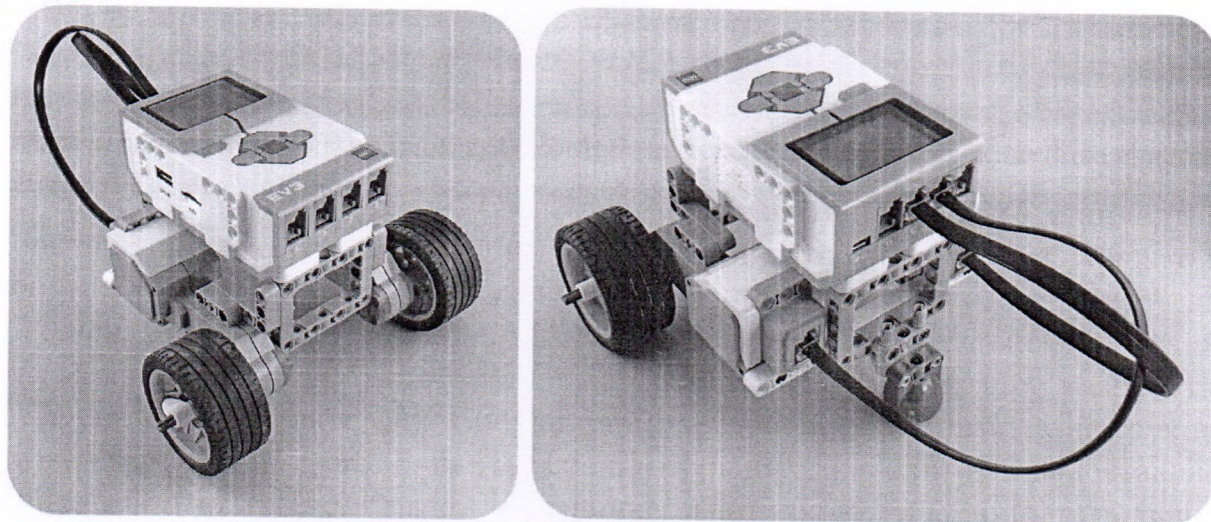


Рисунок 14

3.2 Кинематика мобильного робота с дифференциальным приводом

Робот функционирует в двумерном пространстве (на плоскости) и его положение характеризуется положением и ориентацией связанной системы координат робота $X_R O_R Y_R$ в мировой системе координат XOY : двумя линейными координатами x и y , а также углом курса θ . Робот с двумя приводами обладает недостаточным числом степеней подвижности, таким образом для перемещения из исходной точки в заданную необходимо использование как линейных перемещений так и разворотов (рисунок 15).

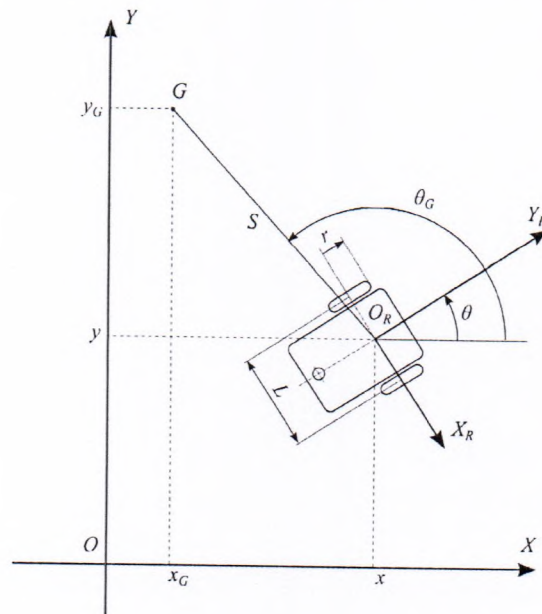


Рисунок 15

Величина необходимых для совершения маневра перемещения и разворота определяется следующим образом:

$$S = \sqrt{(x_0 - x_G)^2 + (y_0 - y_G)^2} \quad (1)$$

$$\theta_G = \text{atan} 2((x_0 - x_G), (y_0 - y_G)) \quad (2)$$

При расчете удобно пользоваться функцией двухаргументного арктангенса:

$$\text{atan} 2(x, y) = \begin{cases} \arctg\left(\frac{y}{x}\right) & \text{при } x > 0; \\ \arctg\left(\frac{y}{x}\right) + \pi & \text{при } x < 0, y \geq 0; \\ \arctg\left(\frac{y}{x}\right) - \pi & \text{при } x < 0, y < 0; \\ +\frac{\pi}{2} & \text{при } x = 0, y > 0; \\ -\frac{\pi}{2} & \text{при } x = 0, y < 0; \\ \text{неопределена} & \text{при } x = 0, y = 0. \end{cases} \quad (3)$$

Изменение угла курса и радиус поворота при движении по криволинейной траектории связаны количеством оборотов ведущих колес следующими соотношениями (рисунок 16):

$$\Delta\theta = \frac{2\pi r(n_R - n_L)}{L} \quad (4)$$

$$R = \frac{2\pi r(n_R + n_L)}{\Delta\theta} \quad (5)$$

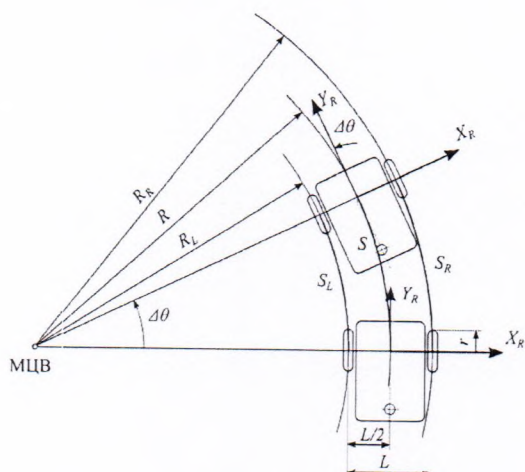


Рисунок 15

Необходимое для перемещение на расстояние количество оборотов колес рассчитывается по формуле:

$$n_R = n_L = \frac{S}{2\pi r} \quad (6)$$

Количество оборотов колес, необходимое для поворота на месте на угол рассчитывается следующим образом :

$$n_R = -n_L = \Delta\theta \frac{L}{4\pi r} \quad (7)$$

3.3 Работа с конструктором LEGO® MINDSTORMS® Education EV3 на примере решения демонстрационной задачи

Для демонстрации работы с конструктором сформулируем задачу, позволяющую понять принципы управления движением мобильных роботов. Робот находится на горизонтальной поверхности (стол). В секторе $\pm 90^\circ$ от продольной оси робота на расстоянии менее 300 мм от точки парковки робота находится цилиндрический объект диаметром около 60 мм и высотой около 100 мм (пластиковый стакан). Необходимо переместить объект в точку, находящуюся позади припаркованного робота и вернуть робота в исходное положение.

Для решения задачи используем доработанный вариант робота RileyRover, оснащенный захватным устройством на основе среднего мотора с червячной передачей для захвата объекта, гироскопическим датчиком для измерения ориентации робота в пространстве, а также ультразвуковым датчиком для обнаружения целевого объекта и измерения расстояния до него. При этом будем учитывать, что принятое в современной инженерной и научной практике направление отсчета углов - против часовой стрелки, а положительное направление отсчета угла датчиком - по часовой стрелке. Для

упрощения работы с датчиком разместим его на конструкции робота в ориентации, при которой поверхность датчика с маркировкой обращена в сторону пола. Виртуальная модель робота приведена на рисунке 16, кабельная сеть условно не показана.

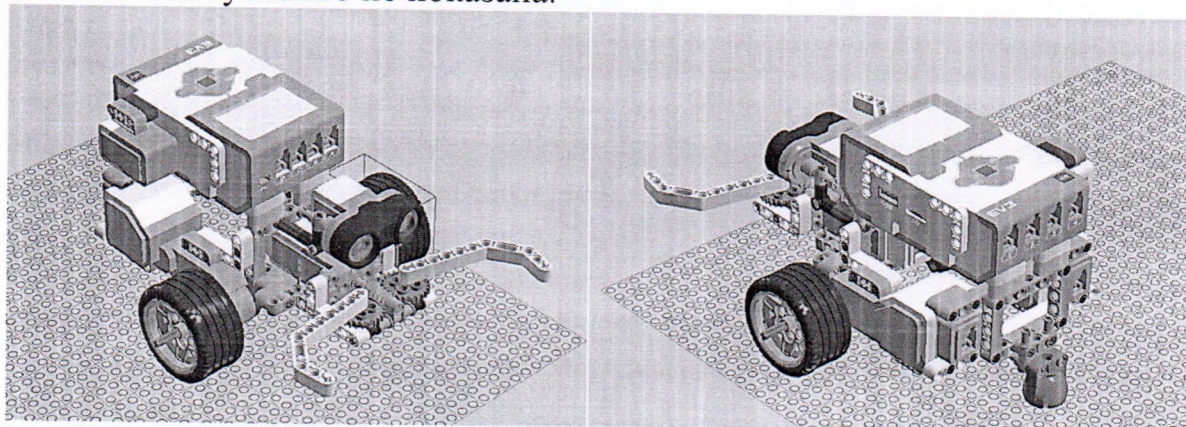


Рисунок 16

Коммутацию исполнительных органов и датчиков с электронным блоком управления выполним следующим образом: левый большой мотор (привод левого колеса) подключен к порту В, правый большой мотор (привод правого колеса) - к порту С, средний мотор (привод захватного устройства) - к порту А, ультразвуковой датчик - к порту 3 а гироскопический датчик - к порту 4.

Задачу будем решать по следующему алгоритму. Примем точку парковки за начало отсчета мировой координат, а исходное направление продольной оси робота за начало отсчета угла курса. Для обнаружения объекта начнем сканировать пространство ультразвуковым датчиком, поворачивая робот в диапазоне углов курса $[-90^\circ, 90^\circ]$ до тех пор, пока показания ультразвукового датчика не станут меньше 30 см, что будет свидетельствовать о попадании края объекта в поле зрения датчика. Зафиксируем значение курса в момент обнаружения края объекта и продолжим поворот в исходном направлении до тех пор пока показания ультразвукового датчика не превысят 30 см, что будет свидетельствовать о выходе противоположного края объекта из поля зрения датчика. Зафиксировав значение курса в этот момент, сложив с ранее зафиксированным значением курса для противоположного края объекта и поделив сумму на 2 получим курс на центр объекта. Повернем робот на полученный курс, замеряем расстояние до объекта, переместим робота вперед на это расстояние и выполним захват объекта. После этого переместим робота на то же расстояние в противоположном направлении и развернем его в ориентацию, отличную от исходной на 180° . Переместим робота вперед на некоторое небольшое расстояние, достаточное для того, чтобы вернувшись в точку парковки развернуться и не задеть перемещенный объект, освободим объект из захвата, переместим робота назад в точку парковки и развернем его в исходную ориентацию.

Для реализации данного алгоритма создадим программу в среде программирования LEGO® MINDSTORMS® Education EV3, приведенную на рисунке 17. Нумерация блоков нанесена дополнительно в процессе подготовки иллюстрации для удобства анализа работы программы. Разработка программы для робота на платформе LEGO® MINDSTORMS® подразумевает детализацию действий задуманного алгоритма до уровня операций с исполнительными и информационными устройствами, органами управления и отображения информации, данными и каналами их передачи, обеспечиваемых программными блоками среды визуального программирования. Разберем работу программы.

Блок 1 "Начало" является стандартным первым блоком программы LEGO® MINDSTORMS® EV3. Блок 2 выполняет сброс на 0 и инициализацию гироскопического датчика, в процессе которой происходит его калибровка. Таким образом устанавливается начало отсчета угла курса, измеряемого датчиком. Далее реализуются два параллельных потока вычислений, в первом осуществляется сканирование пространства в секторе шириной 180°, во втором - обнаружение объекта и все последующие действия. Сканирование реализуется бесконечным циклом (блок 3 "Цикл") в теле которого выполняются блоки 4-11. В блоке 4 "Переменная" выставляется признак направления разворота против часовой стрелки путем присваивания

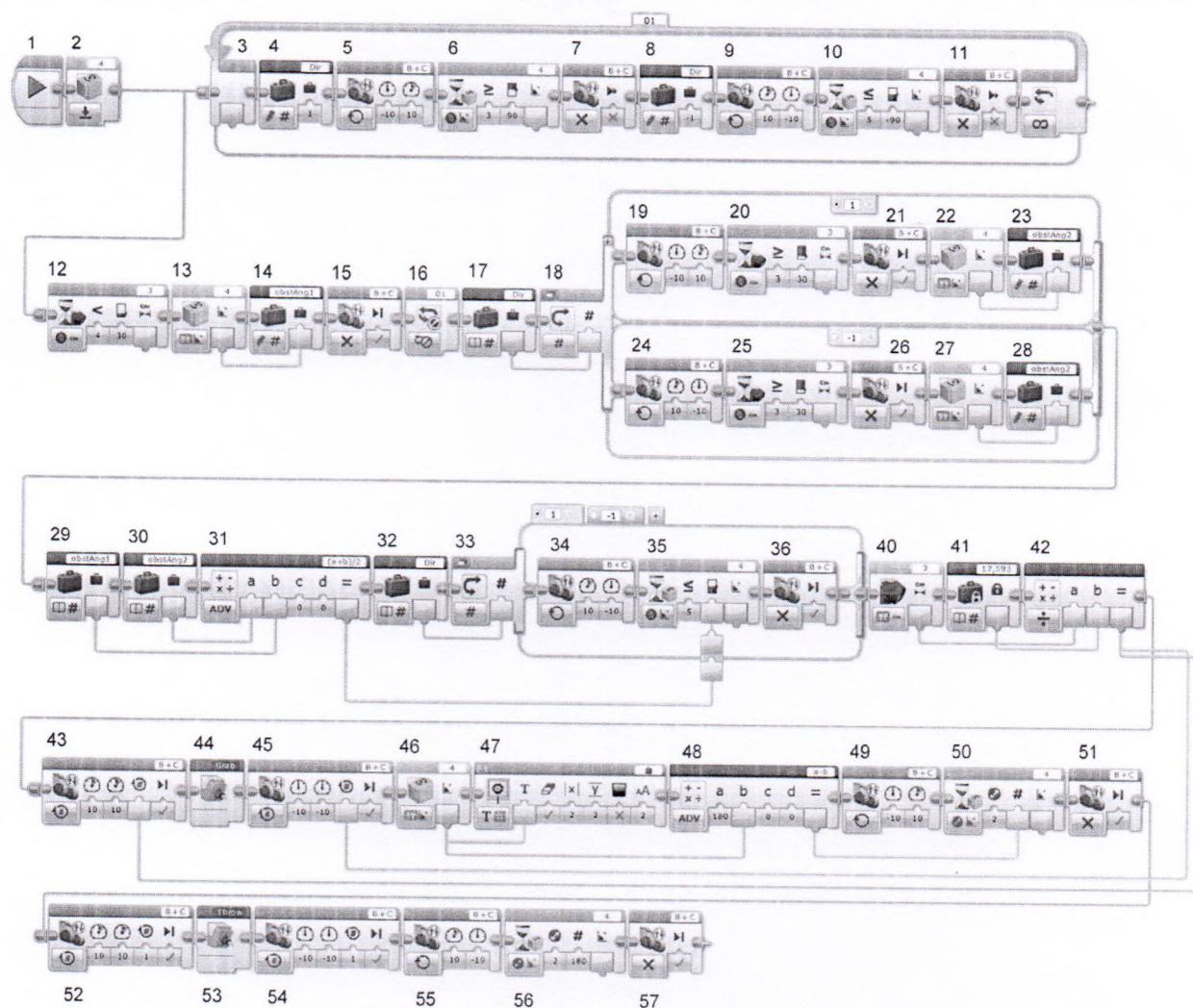


Рисунок 17

переменной Dir значения 1 (поворот против часовой стрелки характеризуется значением 1, по часовой стрелке - значением -1). Блоки 5, 6, 7 реализуют поворот робота на месте на 90° в направлении против часовой стрелки. Блок 5 "Независимое управление моторами" в режиме "Включить" осуществляет включение и вращение в противоположные стороны с одинаковыми скоростями левого и правого мотора, подключенных соответственно к портам В и С. Параметр "Мощность левого мотора" устанавливается на уровне -10, что соответствует движению колеса назад, а параметр "Мощность правого мотора" - на уровне 10, что соответствует движению вперед, таким образом происходит поворот робота на месте относительно вертикальной оси, проходящей через центр оси ведущих колес робота. При этом следует понимать что параметр "Мощность" в контексте его использования в системе программирования LEGO® MINDSTORMS® EV3 не является мощностью в физическом смысле этого понятия, а представляет собой аналог ограничения на максимальную скорость вращения двигателя и блок управления использует этот параметр для регулирования скорости на заданном уровне при использовании блоков управления моторами из палитры "Действия".

Робот поворачивается до тех пор, пока не наступит событие, установленное в блоке 6 "Ожидание", то есть до тех пор, пока показания гироскопического датчика не станут больше или равны 90° . По наступлении этого события срабатывает блок 7 "Независимое управление моторами" в режиме "Выключить" с включенной опцией "Торможение", что приводит к остановке приводов колес, их фиксации в заторможенном состоянии, и, как следствие, прекращению поворота робота. В блоке 8 выставляется признак направления разворота по часовой стрелке ($Dir = -1$) блоками 9, 10, 11 реализуется поворот робота по часовой стрелке до достижения угла курса равного -90° по алгоритму, аналогичному алгоритму работы блоков 5, 6, 7. Далее цикл повторяется. Конструкция "включение"- "ожидание события"- "выключение" будет неоднократно использоваться в дальнейшем.

Во втором потоке вычислений по наступлению ожидаемого в блоке 12 события, то есть при достижении измерений ультразвукового датчика значения меньше 30 см, происходит считывание в блоке 14 и присваивание переменной `obstAng1` значения угла курса, соответствующего обнаруженному краю объекта. Блок 15 осуществляет выключение приводов ведущих колес с торможением и в блоке 16 происходит прерывание цикла 01 первого потока вычислений, то есть прекращение сканирования пространства ультразвуковым датчиком для поиска объекта. Считанное из переменной `Dir` в блоке 17 значение признака направления поворота используется в качестве критерия выбора ветви блока 18 "Переключение" в режиме "Числовое значение". При значении равном 1 (поворот против часовой стрелки) блоками 19, 20, 21 выполняется поворот робота на месте против часовой стрелки до достижения измерений ультразвукового датчика значения больше или равного 30 см с последующей остановкой, считыванием значения угла курса из гироскопического датчика в блоке 22 и присваиванием этого значения переменной `obstAng2` в блоке 23. При значении признака направления поворота равном -1 (поворот по часовой стрелке) выполняется последовательность блоков 24-28 аналогичная вышеописанной последовательности 19-23, за исключением того, что блок 24 выполняет поворот по часовой стрелке. Значения переменных `obstAng1` и `obstAng2` считываются (блоки 29, 30) и используются для вычисления курса на центр объекта блоком 31 "Математика" путем суммирования и последующего деления на 2. По считанному значению признака направления поворота (блок 32) осуществляется выбор ветви программы в блоке 33. Обе ветви блока реализуют поворот робота на месте в направлении, противоположном предшествующему до достижения угла курса, вычисленного в блоке 31, с последующей остановкой. Следует отметить, что для обеспечения возможности передачи в блоки 35 и 38 значения из блока 31 использован режим отображения блока 33 "с вкладками", позволяющий использовать связи с внешними по отношению к ветвям программы блока "Переключение" блоками. Вкладка, содержащая ветвь "-1" блока 33 показана на рисунке 18.

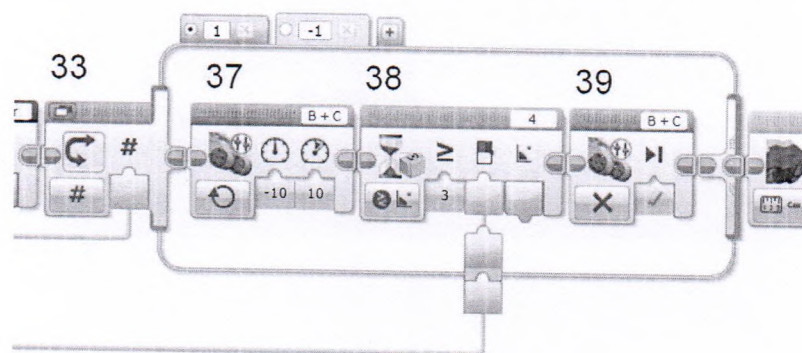


Рисунок 18

Таким образом, по завершении работы блока 33, робот оказывается развернутым своей продольной осью в направлении центра объекта. В блоке 40 из ультразвукового датчика считывается расстояние до объекта, которое используется для вычисления необходимого для сближения с объектом количества оборотов ведущих колес в блоке 42 путем деления на значение длины окружности колеса из блока 41 "Константа". Значение 17,593 см соответствует длине окружности колеса с шиной размерности 56x28 мм, стандартных для базового набора LEGO® MINDSTORMS® Education EV3. В блоке 43 производится включение обоих приводов ведущих колес с параметром "Мощность" равным 10 на число оборотов, рассчитанное в блоке 42, обеспечивая перемещение робота по прямой вперед на расстояние, равное расстоянию до объекта.

Операции закрытия и открытия захватного устройства реализованы пользовательскими блоками Grab и Drop. Последовательность блоков, реализованная в блоке Grab, приведена на рисунке 19. Блоком 58 "Нерегулируемый мотор" с параметром "Мощность" равным -47 производится включение среднего мотора, подключенного к порту А и используемого в качестве привода губок захватного устройства. мотор включается на период 2 секунды, реализуемый блоком 59 "Ожидание" с последующей остановкой с торможением блоком 60. Параметр "Мощность" блока нерегулируемый мотор является аналогом подаваемого на двигатель питающего напряжения. Параметр мощность и время включения мотора определяются экспериментальным путем с целью обеспечения надежного захвата объекта. Последовательность блоков пользовательского блока Drop показана на рисунке 20 и аналогична блоку Grab с той разницей, что мотор включается в противоположном направлении для открытия губок захвата.

Блоком 44 Grab выполняется захват объекта. Блоком 45 выполняется операция обратная выполненной в блоке 43, то есть движение робота захваченным объектом назад с числом оборотов ведущих колес,

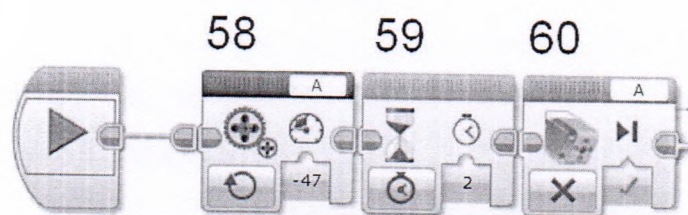


Рисунок 19

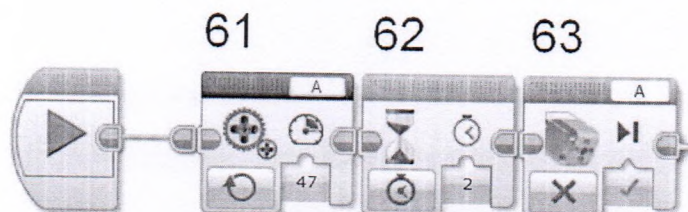


Рисунок 20

вычисленным в блоке 42, в результате чего робот оказывается в исходной точке (точке парковки), развернутым по курсу на угол, измеряемый гироскопическим датчиком в блоке 46. Значение, полученное в блоке 46 отображается блоком 47 на экране блока управления в текстовом режиме шрифтом размера 2 в позиции 2,2 в целях отладки программы. В блоке 48 то же самое значение вычитается из 180° для вычисления угла, на который нужно повернуть робот против часовой стрелки, чтобы он оказался на курсе 180° . Поворот на вычисленный угол осуществляется блоками 49, 50, 51 с той особенностью, что блок 50 срабатывает по факту изменения показаний гироскопического датчика на величину, полученную в блоке 48. Блок 52 перемещает робот вперед на расстояние, равное окружности колеса, блок 53 открывает захватное устройство и освобождает объект, Блок 54 перемещает робот обратно в точку парковки. Блоками 55, 56, 57 робот поворачивается на 180° по часовой стрелке и оказывается в точке парковки на курсе 0° , то есть в исходной ориентации.

Приведенная выше программа позволяет составить представление о работе различных блоков среды программирования LEGO® MINDSTORMS® EV3 а также алгоритмов управления поступательными перемещениями и разворотами мобильного робота с дифференциальным приводом на плоскости. В данном примере не использовалось движение робота по криволинейным траекториям в силу отсутствия необходимости. Тем не менее, возможность реализации таких траекторий в рассматриваемой среде программирования существует, она реализуется использованием блока "Рулевое управление", однако расчет управляющих параметров данного блока для заданного радиуса кривизны траектории с учетом геометрии

Приложение 5.6 к мероприятию 5 Методическое пособие «Робототехника в школе»
робота крайне затруднен, в связи с чем данные параметры подбираются экспериментальным путем.

Литература

1. Конструкторы программируемых роботов.
<http://www.proghouse.ru/article-box/26-robots>
2. Общее руководство пользователя платформы LME EV3. https://le-www-live-s.legocdn.com/sc/media/files/user-guides/ev3/ev3_user_guide_ru-55c924b2039b054f2d746ce24f95302e.pdf
3. leJOS. <http://www.lejos.org/>
4. ev3dev. <http://www.ev3dev.org/>
5. LDD. <http://ldd.lego.com/ru-ru/>
6. Blueprint.
<https://www.eurobricks.com/forum/index.php?/forums/topic/108346-software-blueprint-a-building-instruction-generator-for-ldd/>
7. SR 3D Builder. <http://sr3dbuilder.altervista.org/>
8. MLCAD. <http://mlcad.lm-software.com/>
9. EV3 Basic. <https://sites.google.com/site/ev3basic/>
10. Программирование робота Lego Mindstorms EV3 с помощью Scratch 2.0
<http://www.proghouse.ru/article-box/52-scratch2-ev3>
11. Open Roberta Lab – новый способ Scratch-подобного программирования роботов Lego Mindstorms EV3. <http://www.proghouse.ru/article-box/86-open-roberta-lab>
12. ROBOTC. <http://www.robotc.net/>
13. RileyRover. <http://www.damienkee.com/home/2013/8/2/rileyrover-ev3-classroom-robot-design.html>

Ст. преподаватель



В.В. Зеленцов

Выписка
из протокола № 5
заседания учебно-методической комиссии НУК ФН МГТУ им. Н.Э.Баумана
от 20 апреля 2017 года

Слушали:

сообщение председателя учебно-методической комиссии Еркович Ольги Станиславовны об опубликовании методического пособия «Робототехника в школе» (автор – старший преподаватель кафедры СМ7 – «Робототехнические системы и мехатроника» Зеленцов В.В.) Данное методическое пособие предназначено для учителей и педагогов-организаторов, работающих по проекту Департамента образования г. Москвы «Инженерный класс в московской школе» и преподавателей профильных школ МГТУ им. Н.Э. Баумана, проходящих курс повышения квалификации в университете, а также может быть использовано при профориентационной работе с учащимися старших классов профильных школ МГТУ им. Н.Э. Баумана. Представленное методическое пособие полностью соответствует утвержденной программе курса повышения квалификации.

Постановили:

рекомендовать методическое пособие старшего преподавателя кафедры СМ7 – «Робототехнические системы и мехатроника» Зеленцова В.В. «Робототехника в школе» к использованию при реализации программы курса повышения квалификации «Введение в инженерную специальность. Инженерные школы России» и при профориентационной работе с учащимися старших классов профильных школ МГТУ им. Н.Э. Баумана..

Председатель
учебно-методической комиссии НУК ФН



О.С. Еркович

Секретарь
учебно-методической комиссии НУК ФН



Г.М. Петрова