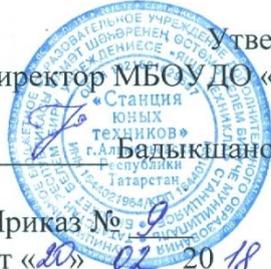


МУНИЦИПАЛЬНОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ДОПОЛНИТЕЛЬНОГО ОБРАЗОВАНИЯ
«СТАНЦИЯ ЮНЫХ ТЕХНИКОВ» ГОРОДА АЛЬМЕТЬЕВСКА РЕСПУБЛИКИ ТАТАРСТАН

Принято на заседании
методического совета
Протокол № 3
от «16» 02 20 18 г.

Утверждаю
Директор МБОУ ДО «СЮТ»

Бадлыкшанов Н.М.

Приказ № 9
от «20» 02 20 18 г.

ДОПОЛНИТЕЛЬНАЯ ОБЩЕОБРАЗОВАТЕЛЬНАЯ
ОБЩЕРАЗВИВАЮЩАЯ ПРОГРАММА
«РОБОТОТЕХНИКА»

Направленность: техническая
Возраст учащихся: 7 - 12 лет
Срок реализации: 3 года

Автор-составитель:
Федорова Венера Геннадиевна,
педагог дополнительного образования

Альметьевск, 2018 г.

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Задача инновационного развития экономики требует опережающего развития образовательной среды, в том числе развития детского технического творчества. Одной из наиболее инновационных областей в сфере детского технического творчества является образовательная робототехника, которая объединяет классические подходы к изучению основ техники и современные направления: информационное моделирование, программирование, информационно-коммуникационные технологии.

Робототехника – прикладная наука, занимающаяся разработкой автоматизированных технических систем. Робототехника опирается на такие дисциплины как электроника, механика, программирование.

Актуальность создания программы обусловлена следующими причинами:

- Наблюдается сильнейший дефицит качественных молодых инженерно-конструкторских кадров для существующих и развивающихся российских предприятий;
- У молодых людей при поступлении в вузы отсутствуют навыки практической работы, представления о задачах, решаемых инженерами и конструкторами, результатом чего становится неосознанный выбор специальности, рост числа молодых людей неработающих по специальности после вузов.

Новизна программы. Дополнительных образовательных программ по робототехнике для обучения детей с использованием конструктора LEGO Mindstorms Education NXT 2.0 очень мало, их структура и учебно-тематические планы не соответствуют требованиям учреждений дополнительного образования.

Программа реализуется во взаимосвязи со школьными предметами естественно-научного цикла. Теоретические и практические знания по робототехнике значительно углубят знания обучающихся по ряду разделов физики (статика и динамика, электричество и электроника, оптика), информатике и математике.

ПЕДАГОГИЧЕСКАЯ ЦЕЛЕСООБРАЗНОСТЬ ПРОГРАММЫ

Дети - неутомимые конструкторы, их творческие способности и технические решения остроумны, оригинальны. Чтобы достичь высокого уровня творческого и технического мышления, дети должны пройти все этапы конструирования. Необходимо помнить, что такие задачи ставятся, когда обучающиеся имеют определённый уровень знаний, опыт работы, умения и навыки. Юные исследователи, войдя в занимательный мир роботов, погружаются в сложную среду информационных технологий, позволяющих роботам выполнять широчайший круг функций.

Практическая значимость. Работа с образовательными конструкторами Lego Education WeDo и Lego MindStorms позволяет обучающимся в форме познавательной игры узнать многие важные идеи и развить необходимые в дальнейшей жизни навыки. При построении модели затрагивается множество проблем из разных областей знаний - от механики до биологии, истории - что является вполне естественным.

ЦЕЛЬ И ЗАДАЧИ ПРОГРАММЫ

Цель программы: развитие творческих познавательных и изобретательских способностей обучающихся, через ознакомление с основами робототехники, конструирования и программирования.

Задачи:

Обучающий:

- ставить цель, работать с информацией, моделировать;
- развитие образного, технического, логического мышления;
- развитие умения работать по предложенным инструкциям;
- развитие умения творчески подходить к решению задачи;
- способствовать формированию умения самостоятельно решать технические задачи в процессе конструирования моделей (планирование предстоящих действий, самоконтроль, умение применять полученные знания, приемы и опыт в конструировании других объектов);

- стимулировать смекалку, находчивость, изобретательность и устойчивый интерес к поисковой творческой деятельности;
- формировать умение работать с литературой, с видеотекой, в Интернете, в PowerPoint;
- Коммуникативные:
- умение с достаточной полнотой и точностью выражать свои мысли в соответствии с задачами и условиями коммуникации;
- развитие умения работать над проектом в команде, эффективно распределять обязанности.
- выстраивать доброжелательные отношения в коллективе, уметь разрешать конфликты, осуществлять взаимопомощь, а также эффективно добывать знания и приобретать соответствующие умения при взаимодействии со сверстниками умение сопереживать, учиться правильно оценивать себя и свои поступки.
- Развивающие:
- мотивация к изучению наук естественно - научного цикла;
- самостоятельно собирать нужные сведения.

Этапы реализации программы

В первый год учащиеся проходят курс конструирования LegoEducationWeDo.

Во второй год учащиеся проходят курс LegoMindStorms. Программирование в графической среде NXT – G.

На третий год учащиеся изучают основы теории автоматического управления, интеллектуальные и командные игры роботов, строят роботов для участия в соревнованиях республиканского, зонального и российского уровня, а также роботов повышенной сложности (с двумя и более NXT).

Материальные ресурсы:

- конструктора LEGO® WeDo™ (LEGOEducationWeDoSoftware);
- LegoMindstormsNXT;
- набор ресурсный средний;
- компьютер, проектор, экран.

Работа с родителями:

Основными формами работы с родителями для реализации программы является:

1. Рекомендации по индивидуальному обучению ребенка.
2. Индивидуальные беседы с родителями.
3. Родительские собрания.
4. Дни открытых дверей.

Методическая работа

1. Составление методических разработок.
2. Мониторинг образовательного процесса.
3. Посещение занятий педагогов других объединений с целью обмена опытом.

Ожидаемые результаты и способы их проверки:

- формирование устойчивого интереса к робототехнике и учебным предметам физика, технология, информатика;
- формирование умения работать по предложенным инструкциям;
- формирование умения творчески подходить к решению задачи;
- формирование умения довести решение задачи до работающей модели;
- формирование умения излагать мысли в четкой логической последовательности, отстаивать свою точку зрения, анализировать ситуацию и самостоятельно находить ответы на вопросы путем логических рассуждений.
- формирование умения работать над проектом в команде, эффективно распределять обязанности.

- подготовка к состязаниям по Робототехнике - конструированию.

Формы подведения итогов реализации ДОП:

- муниципальная, региональная состязания по Робототехнике;
- турниры на звание лучшего программиста и конструктора по Робототехнике.

Данная программа направлена на:

- помощь воспитанникам в индивидуальном развитии;
- мотивацию к познанию и творчеству;
- к стимулированию творческой активности;
- развитию способностей к самообразованию;
- приобщение к общечеловеческим ценностям.

УЧЕБНО-ТЕМАТИЧЕСКИЙ ПЛАН НА 1 ГОД ОБУЧЕНИЯ

№ п/п	Тема занятия	Количество часов		Итого
		теория	практика	
1.	Вводное занятие, техника безопасности	2	-	2
2.	Знакомство с конструктором	8	20	28
2.1	Знакомство с LegoWeDo, его составляющими частями.	4	10	14
2.2	Элементы конструктора. Коммутатор, Мотор, Датчик наклона, Датчик движения	4	10	14
3.	Проект «Звери»	10	40	50
3.1	Изготовление модели «Танцующие птицы».	2	8	10
3.2	Изготовление модели «Голодный аллигатор»	2	8	10
3.3	Изготовление модели «Обезьянка – барабанщица»	2	8	10
3.4	Изготовление модели «Порхающая птица»	2	8	10
3.5	Изготовление модели «Рычащий лев»	2	8	10
4.	Проект «Приключения»	8	24	32
4.1	Изготовление модели «Умная вертушка»	2	6	8
4.4	Изготовление модели «Непотопляемый парусник»	2	6	8
4.5	Изготовление модели «Спасение самолета»	2	6	8
4.6	Изготовление модели «Спасение от великана»	2	6	8
5.	Проект «Футбол»	6	18	24
5.1	Изготовление модели «Вратарь»	2	6	8

5.2	Изготовление модели «Нападающий»	2	6	8
5.3	Изготовление модели «Ликующие болельщики»	2	6	8
6.	Проект «LEGO». Защита проектов.	-	6	6
7.	Итоговое занятие	-	2	2
	Итого:	34	110	144

СОДЕРЖАНИЕ ПРОГРАММЫ

1 год обучения

1. Вводное занятие, инструктаж по технике безопасности.

2. «Знакомство с конструктором».

Применение роботов в современном мире: от детских игрушек, до серьезных научных исследовательских разработок. Демонстрация передовых технологических разработок, представляемых на Международной выставке роботов. История робототехники. От глубокой древности до наших дней. Знакомство с конструктором с ЛЕГО-детальями, с цветом ЛЕГО-элементов, с формой. Знакомство с мотором. Построение модели, показанной на картинке. Выработка навыка поворота изображений и подсоединения мотора к ЛЕГО-коммутатору. Знакомство детей с панелью инструментов, функциональными командами; составление программ в режиме Конструирования. Знакомство с зубчатыми колёсами. Структура и ход программы. Датчики и их параметры: датчик поворота; датчик наклона. «Ременная передача» и «Перекрёстная ременная передача». «Зубчатые колёса», «Промежуточное зубчатое колесо», «Повышающая зубчатая передача», «Понижающая зубчатая передача» и «Коронное зубчатое колесо». Знакомство с блоками «Прибавить к Экрану», «Вычесть из Экрана», «Начать при получении письма», «Цикл» и т. д.

3. Проект «Звери»

Основной предметной областью является технология, понимание того, что система должна реагировать на свое окружение. Создание и программирование действующих моделей: Изготовление модели «Танцующие птицы», «Голодный аллигатор», «Обезьянка – барабанщица», «Порхающая птица», «Рычащий лев». Интерпретация двухмерных и трехмерных иллюстраций и моделей. Понимание того, что животные используют различные части своих тел в качестве инструментов. Сравнение природных и искусственных систем. Использование программного обеспечения для обработки информации. Демонстрация умения работать с цифровыми инструментами и технологическими системами.

4. Проект «Приключения»

Раздел «Приключения» сфокусирован на развитии речи, модель используется для драматургического эффекта. Ученики осваивают важнейшие вопросы любого интервью Кто?, Что?, Где?, Почему?, Как?, исполняют диалоги и последовательно описывают приключения. Создание и программирование действующих моделей: «Умная вертушка», «Непотопляемый парусник», «Спасение самолета», «Спасение от великана».

5. Проект «Футбол»

Данный раздел связан с математикой. На занятии воспитанники проводят: измерение расстояние, на которое улетает бумажный мячик; подсчет числа голов, промахов и отбитых мячей; использование чисел для оценки качественных показателей, чтобы определить наилучший результат в трёх различных категориях. Усвоение понятия случайного события. Использование чисел для задания звуков и для задания продолжительности работы мотора. Использование чисел при измерениях и при оценке качественных параметров. Создание и программирование действующих моделей: «Вратарь», «Нападающий», «Ликующие болельщики».

УЧЕБНО-ТЕМАТИЧЕСКИЙ ПЛАН 2 ГОД ОБУЧЕНИЯ

№ п/п	Тема занятия	Количество часов		Итого
		теория	практика	
	Вводное занятие. Инструктаж по технике безопасности.	2	-	2
	Что такое робототехника. История развития робототехники. Знакомство с деталями конструктора.	2	-	2
	Общий обзор технологии NXT.	2	2	4
	Экран отключения блока NXT	2	2	4
	Настройка NXT. Главное меню NXT.	2	2	4
	Среда программирования NXT.	2	2	4
	Создание или открытие программы.	2	2	4
	Доступ к программным блокам.	2	2	4
	Политра блоков NXT-G	2	2	4
	Первая программа.	2	2	4
	Исходный узел для построения программы на NXT-G.	2	2	4
	Котроллер NXT-G.	2	6	8
	Настройки блока Display.	2	2	4
	Программирование моторов: команда Move.	2	20	22
	Настройки блока Move.	2	14	16
	Движение одним мотором.	4	14	18
	Движение двумя мотором.	4	14	18
	Датчик цвета.	4	14	18
	Датчик касания.	4	14	18
	Простейшие виды циклов.	4	14	18
	Творческое конструирование собственной модели. Программирование.	-	28	28
	Экскурсия в объединения	2	2	4
	Подведение итогов.	2	2	4
ИТОГО:		54	162	216

СОДЕРЖАНИЕ ПРОГРАММЫ 2 года обучения

1. Вводное занятие. Инструктаж по технике безопасности;
 - правильно установить компьютер;
 - организация для работы в группе рабочее место с компьютером и свободным местом для сборки моделей;
 - предусмотреть место для контейнера с деталями и «сборочной площадки»;

- конструктор отрывайте правильно, придерживая крышку;
- детали держите в специальном контейнере;
- при работе с конструктором важно следить за деталями, так как они очень мелкие;
- нельзя детали брать в рот, раскидывать на рабочем столе;
- при работе с компьютерами надо быть очень осторожными, чтобы не повредить монитор, при подключении конструкции, соблюдать порядок подключения.
- после окончания сборки, проверки на компьютере, конструкция разбирается, детали укладываются в коробку, компьютер выключается.

2. Что такое робототехника. История развития робототехники. Знакомство с деталями конструктора.

Робот - это программируемое механическое устройство, способное выполнять задачи и взаимодействовать с внешней средой без помощи со стороны человека. Робототехника - это научная и техническая база для проектирования, производства и применения роботов.

Слово «робот» было впервые использовано чешским драматургом Карлом Чапеком в 1921. В его произведении «Универсальные роботы Россума» речь шла о классе рабов, искусственно созданных человекоподобных слуг, сражающихся за свою свободу. Чешское слово «robota» означает «принудительное рабство». Слово «робототехника» было впервые применено известным автором научной фантастики Айзеком Азимовым в 1941 году.

Первое место в мире по производству и использованию роботов занимает Япония. В 1928 году под руководством доктора НисимурыМакоото был создан робот, названный «Естествоиспытатель», высотой 3,2 метра. Оснащенный моторчиками, он мог менять положение головы и рук. А 21 ноября 2000 года на первой в истории выставке ROBODEX в городе Йокохама, Япония, TokyoSonyCorporation представляет своего первого человекоподобного робота "SDR-3X".

3. Общий обзор технологии NXT.

Проводится поверхностный обзор робототехнического комплекса, включающего конструктор LegoMindstorms NXT 2.0 и среду программирования NXT-G. Цель: познакомиться с основными компонентами конструктора LegoMindstorms NXT 2.0, интерфейсом среды NXT-G и научиться создавать простейшую программу "Hello, world!".

5. Внешний вид блока NXT и его разъёмы:



(Рис. 1.1) Порт 1: Датчик касания

Порт 2: Датчик касания

Порт 3: Датчик цвета

Порт 4: Ультразвуковой сенсор

Порт А: Мотор для дополнительных функций

Порт В: Мотор для движения

Порт С: Мотор для движения

Блок NXT работает от шести батарей типа АА. В образовательную версию набора входит аккумуляторная батарея.

Работа с меню NXT происходит при помощи четырёх кнопок:

- Серые треугольники - кнопки "вперёд" (вправо) и "назад" (влево) - позволяют перемещаться внутри меню до нужного пункта;
- Оранжевый квадрат соответствует клавише ввода; она же используется для включения робота;

- Тёмно-серый прямоугольник - "отмена" или переход назад к предыдущему пункту. Эта же кнопка используется для выключения робота. Для выключения NXT нажимайте кнопку до тех пор, пока на экране не появится надпись **Turnoff?**, после чего для подтверждения выключения нажмите оранжевую кнопку. Также можно нажать и удерживать тёмно-серую кнопку до полного выключения NXT.

4. Экран отключения блока NXT

Задание 1.1. Научитесь включать и выключать свой NXT. Попробуйте выключить NXT двумя способами, описанными выше.

Вся информация на дисплее NXT отображается на английском языке. При включении экран NXT



выглядит как на рис.:

5. Настроим NXT так, чтобы он автоматически выключался через две минуты простоя. Для этого выберем из главного меню **Settings\Sleep**. Затем при помощи стрелок выбираем нужное значение (текущее значение отображается на экране) и нажимаем на ввод (оранжевая кнопка). Если теперь не работать с NXT в течении двух минут, то он выключится сам. Обратите внимание на вариант **Never**(Никогда): в этом режиме NXT не будет выключен, пока Вы сами этого не сделаете. Но это может привести к более быстрому разряду батарей.

Пример 1.2. Попробуем проиграть звуковые файлы на блоке NXT. Главное меню **\My Files\Sound files\Good Job**. В окне отображается имя выбранного файла и варианты работы с ним, которые можно пролистать при помощи стрелок:

- **Run** - запустить выбранный файл на выполнение (иконка рабочего состояния). Если выбрать этот вариант, звуковой файл будет воспроизведён. В процессе работы файла на экран будет выведено сообщение **Running**, а по завершению - **Done**.
- **Delete** - удалить выбранный файл (иконка корзины). Если выбрать этот вариант, появится предупреждающее сообщение **Are you sure?** (Вы уверены?). Вариант по умолчанию - иконка в форме креста (Нет). В случае, если выбранный файл всё же нужно удалить, то при помощи стрелки следует выбрать иконку с галочкой (Да).
- **Send** - переслать (иконка письма). Для того, чтобы переслать файл другим устройствам, следует вначале настроить канал связи **Bluetooth**.

Задание 1.3. Установите максимальную громкость динамиков. Проверьте громкость, проиграв произвольный звуковой файл NXT. Установите комфортную для себя громкость.

Задание 1.4. Изучите различные варианты автоотключения NXT. Установите таймер автоотключения на 10 минут.

Задание 1.5. Протестируйте датчики касания, ультразвука, мотор при помощи меню **TryMe** (не забудьте про стандартные порты подключения сенсоров).

Работу с другими пунктами меню будем рассматривать по мере необходимости.

6. Среда программирования NXT.

Запустите программное обеспечение **Mindstorms NXT 2.0**. В открывшемся окне можно просмотреть видео "**GettingStarted Guide**" (Руководство к быстрому старту) и "**Software Overview**"



(Обзор программного обеспечения).

7. Создание или открытие программы.

Внимание! Экран NXT отображает текст только на английском языке, поэтому символы кириллицы в имени файла недопустимы.

В дальнейшем открыть уже созданную программу можно, выбрав её из списка в поле Open recent program (Открыть существующую программу) (Рис. 1.8).

Для создания и открытия файлов можно использовать и стандартные средства - комбинации клавиш Ctrl+N и Ctrl+O соответственно или воспользоваться меню File.

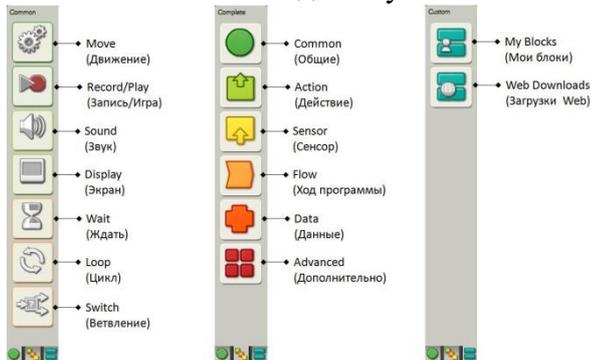
Вкладки в левой верхней части окна среды позволяют открыть RoboCenter и MyPortal. Эти возможности мы оставляем за рамками настоящего пособия и рекомендуем читателю самостоятельно с ними ознакомиться. Просто нажмите на крестик, чтобы увеличить область программирования.



8. Доступ к программным блокам.

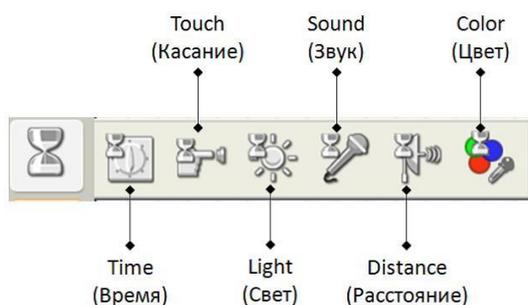
В NXT-G реализован визуальный способ проектирования программ, что очень удобно для обучения. Программа составляется из блоков. Каждый блок представляет различные типы действий. Блоки можно настраивать. Все блоки организованы и представлены в трёх палитрах программирования (Рис. 1.10) - "Common" (Общая), "Complete" (Полная) и "Custom" (Пользовательская). Одновременно можно работать только с одной палитрой.

Блоки имеют различную окраску, которая зависит от функционального назначения блока. Блоки, функциональное назначение которых, похоже, окрашены одинаково. Например, все блоки, которые отвечают за выполнение некоторого действия (Action), окрашены зелёным цветом, а блоки, которые отображают текущие значения датчиков - жёлтым цветом. Цветовая окраска позволяет легко находить нужные блоки в палитре.



9. Палитры блоков NXT-G

Блоки, которые используются наиболее часто, сгруппированы в палитре "Common". Каждый иконка этой палитры является отдельным программным блоком, кроме иконки **Wait**, которая показывает шесть вариантов этого блока:



10. Первая программа.

Продолжим создание программы и тем самым завершим обзор интерфейса среды NXT-G. Правее палитры программирования находится большая область программирования. Блоки перетаскиваются мышью из палитры на специальную направляющую в виде балки **Lego**, которая увеличивается с очередным добавленным блоком.

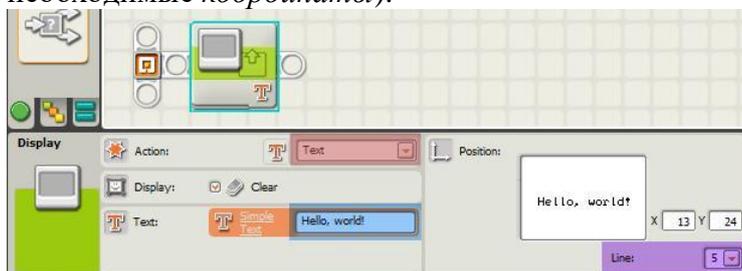


11. Исходный узел для построения программы на NXT-G

Пример 1.4. Создадим классическую для вхождения в язык программирования программу, которая будет выводить на экран сообщение: Hello, world!

Добавьте на *лист* программы блок Display из общей палитры. В нижней части окна отображается панель настроек текущего (выделенного) блока. По умолчанию блок настроен на *отображение* изображений (Action:Image), а именно, улыбающегося смайлика. Смена картинки возможна в свойстве File. Свойство Display позволяет очищать экран (Clear) перед выводом нового изображения.

Нам нужно вывести на экран текст, поэтому изменим свойство Action на Text (Рис. 1.12), заменим текст по умолчанию на Hello, world! и выберем позицию отображения Line - 5 (можно просто щёлкнуть мышью в нужном месте свойства Position или задать необходимые *координаты*).



12. Настройки блока Display

Обратите внимание на то, что изменение настроек блока изменяет и внешний вид блока.

Наша *программа* получилась очень компактной, она полностью умещается на экране. В дальнейшем мы встретимся с более длинным (в прямом смысле этого слова) кодом. Для того, чтобы увидеть код, который не поместился на экране, можно воспользоваться курсорными стрелками. Кроме этого можно щёлкать мышкой по области кода в правом нижнем углу экрана (становится доступной, если щёлкнуть по вкладке с изображением лупы).

Для загрузки и запуска программ на блок NXT в среде NXT-G используются *управляющие* кнопки или, по-другому, *контроллер*:



12. Контроллер NXT-G

Подключите робот шнуром *USB* к компьютеру. Включите робот нажатием на оранжевую кнопку. Запустите программу на выполнение при помощи центральной кнопки контроллера. Что Вы увидели? На экране мелькнула надпись *Hello!* и затем сразу *Done*. Это означает, что программа *Hello!* уже завершила свою работу, а Вы этого так и не увидели. Обратите внимание, что сейчас на экране NXT отображается *меню Softwarefiles*.

Для того, чтобы успеть увидеть текст на экране NXT до завершения программы, нужно использовать задержку времени. Добавим в программу новый блок *Wait* и настроим его на отсчёт двух секунд:



13. Настройки блока Wait

Ещё раз запустите программу на выполнение при помощи центральной кнопки контроллера. Теперь текст остаётся на экране в течение двух секунд. Поскольку сейчас на экране NXT активна надпись *Run*, то можно повторить выполнение программы нажатием на оранжевую кнопку ввода.

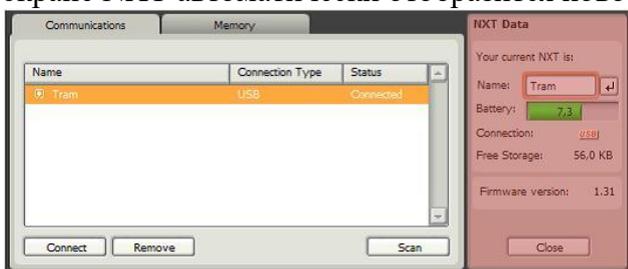
Задание. Создайте новую программу *Eyes*, в которой на экране NXT будут изображены глаза, смотрящие налево-направо. Используйте файлы *Lookingleft* и *Lookingright* для имитации взгляда налево/направо. Смена направления взгляда должна происходить четыре раза.

Завершим тему присваиванием нового имени Вашему NXT-блоку.

Пример. Изменение имени NXT. Нажмите на кнопку *NXT window* контроллера. В результате откроется окно, правая часть которого *NXT Data* содержит следующую информацию:

- Name (Имя)
- Battery (Заряд батареи)
- Connection (Соединение)
- FreeStorage (Свободная память)
- FirmwareVersion (Версия программного обеспечения, так называемой "прошивки")

Введите в *поле* Name новое имя Вашего NXT и нажмите на кнопку ввода рядом с этим полем. На экране NXT автоматически отобразится новое имя.



14. Программирование моторов: команда Move.

Рассматриваются возможности управления моторами робота LegoMindstorms NXT 2.0 при помощи блока Move. Изучаются настройки блока. Цель: научиться управлять роботом при помощи блока Move.

Одна из основных функций робота - движение. Двигаться может как весь робот целиком, так и отдельные его части. Движением управляют сервомоторы (или сервоприводы). В конструкторе LegoMindstorms сервомоторы имеют датчики оборотов, подсчитывающие количество градусов поворота главной оси. Важным компонентом сервомотора является редуктор, который через систему шестерней превращает чрезвычайно быстрое вращение внутреннего электрического двигателя в более медленное. Наличие датчика оборотов и редуктора позволяют сервомотору совершать прецизионные движения главной оси. Сервопривод Lego может быть повернут с точностью 1°. Внутреннее устройство сервопривода показано на рис.



Программирование движения происходит посредством блока Move (Движение), который находится в общей и полной палитрах



15. Настройки блока Move

На рисунке цифрами отмечены:

1. Моторы, которыми управляет этот блок.
2. Направление вращения моторов.
3. Уровень мощности мотора (скорость). Заметим, что реальная скорость робота будет зависеть также и от его конструкции, типа поверхности (скользящая, шершавая и пр.), наклона поверхности, массы робота и т. п.
4. Параметр длительности движения: без ограничения, в градусах, оборотах или секундах.

Настройка блока Move, как и других блоков, производится в нижней части экрана после выбора блока:



16. Движение одним мотором.

Измените предыдущую программу так, чтобы робот вращал только один из двух подключенных моторов. Установите длительность вращения в восемь оборотов. Понаблюдайте, как меняется характер движения робота в зависимости от направления движения.

Имейте в виду, что длины USB-кабеля может не хватить на восемь оборотов. Поэтому не забудьте отсоединить его перед запуском программы на выполнение!

Заполните таблицу:

Мотор	Направление	Характер движения
Левый	Вперёд	
Левый	Назад	
Правый	Вперёд	
Правый	Назад	

При выполнении дальнейших заданий Вам понадобится тестовое поле, которое поставляется в комплекте с роботом.

17. Движение двумя моторами .

Изучите доступные параметры продолжительности движения, изменяя их в поле Duration. Установите NextAction в Brake и проведите замеры расстояний¹, которые проезжает робот при различных значениях параметра Duration и заполните таблицу:

18. Датчик цвета.



Датчик цвета (RGB-датчик) совмещает три функции:

1. Позволяет роботу различать цвета.
2. Позволяет роботу различать уровень освещённости путём измерения яркости отражённого света.
3. Цветовая подсветка.

19. Датчик касания.



Добавьте в конструкцию робота FiveMinuteBot датчик касания - кнопку. Его можно не закреплять, а подключить длинным кабелем и использовать как пульт дистанционного управления.

В настройках блока WaitTouch в качестве Action доступны варианты:

- Pressed - датчик нажат;
- Released - датчик отпущен;
- Bumped - выполнен щелчок (то есть кнопка нажата и сразу отпущена).

Задание. Напишите игру "Кто точнее", смысл которой состоит в том, чтобы остановить робот точно на заданной линии. После запуска программы робот должен начать движение по направлению к чёрной линии. Как только колёса робота коснутся линии, игрок должен нажать кнопку; при этом робот должен остановиться. Выигрывает тот, у кого расстояние от передних колёс до линии меньше. Настройки блока Move: неограниченное движение, мощность 80 %. Подберите наилучший вариант для решения задачи.

20. Простейшие виды циклов.

Задание 1. Подготовьте робот с датчиком цвета в передней части; датчик должен быть направлен вниз и находиться на расстоянии около 1 см от стола. Расположите робот перед чёрным полем (или чёрной линией) на расстоянии от неё около 20 см.

Задание 2. Движение до линии. Используя датчик цвета в режиме измерения освещённости, напишите программу, позволяющую роботу доехать до чёрной полосы и остановиться (см. задание 3).

Задание 3. Движение до линии и обратно. Измените программу из задания 4 так, чтобы после достижения чёрной линии робот отъезжал на белое поле.

Задание 4. Бесконечное циклическое движение до линии и обратно. Измените программу из задания 4 так, чтобы робот выполнял движение до линии и обратно бесконечное количество раз. Для этого заключите предыдущую программу в бесконечный (forever) цикл.

Задание 5. Циклическое движение до линии и обратно по счётчику. Измените программу из задания 4 так, чтобы робот выполнял движение до линии и обратно три раза. Для этого измените настройки цикла следующим образом:

21. Творческое конструирование собственной модели. Программирование.

22. Экскурсии в объединения.

УЧЕБНО-ТЕМАТИЧЕСКИЙ ПЛАН 3 ГОД ОБУЧЕНИЯ

№ п/п	Тема занятия	Количество часов		Итого
		теория	практика	
1.	Вводное занятие. Инструктаж по технике безопасности.	2	-	2
2.	Повторение второго года обучения.	-	4	4
3.	Способы крепления деталей. Высокая башня.	6	10	16
4.	Механический манипулятор (хваталка).	6	10	16
5.	Механическая передача. Волчок.	6	10	16
6.	Механическая передача. Ручной миксер.	6	10	16
7.	Тележки. Одномоторная тележка.	6	10	16
8.	Знакомство со средой Robolab 2.9. Шагающий робот.	6	10	16
9.	Приводная платформа EV3 на гусеничном ходу	8	8	16
10.	Типы команд. Команды действия. Базовые команды.	8	8	16
11.	Моторы NXT.	8	8	16
12.	Кегельринг. Танец в круге.	6	10	16
13.	Движение вдоль линии. Один датчик.	6	10	16
14.	Творческое конструирование собственной модели. Программирование.	-	16	16
15.	Защита модели.	-	14	14
16.	Экскурсии в объединения.	-	2	2
17.	Итоговое занятие	-	2	2
	Итого:	74	142	216

СОДЕРЖАНИЕ ПРОГРАММЫ 3 год обучения

1. Вводное занятие. Инструктаж по технике безопасности;

- правильно установить компьютер;
- организация для работы в группе рабочее место с компьютером и свободным местом для сборки моделей;
- предусмотреть место для контейнера с деталями и «сборочной площадки»;
- конструктор отрывайте правильно, придерживая крышку;
- детали держите в специальном контейнере;
- при работе с конструктором важно следить за деталями, так как они очень мелкие;
- нельзя детали брать в рот, раскидывать на рабочем столе;
- при работе с компьютерами надо быть очень осторожными, чтобы не повредить монитор, при подключении конструкции, соблюдать порядок подключения.
- после окончания сборки, проверки на компьютере, конструкция разбирается, детали укладываются в коробку, компьютер выключается.

2. Повторение второго года обучения. Детали конструктора, программирование.



3. Высокая башня.

Задача, которую вам предстоит решить сегодня - строительство самой высокой башни из всех возможных деталей конструктора. Желательно до потолка. При этом учитываем, что микроконтроллер не надо вешать на самую верхушку, шпиль должен быть легким, а основная масса сосредоточена внизу.

4. Механический

манипулятор (хваталка).

Используя балки и штифты, создать механизм, способный изменять длину и захватывать детали. Требования к конструкции следующие:

- хватательный механизм должен иметь минимальную длину в сложенном состоянии и максимальную в разложенном;
- у механизма должно быть две ручки, как у щипцов, и многоколенчатое соединение, ведущее к хватательной части;
- изобретатель должен суметь взять с помощью хваталки некоторый предмет и перенести его с места на место.

5. Механическая передача. Передаточное отношение. Волчок.

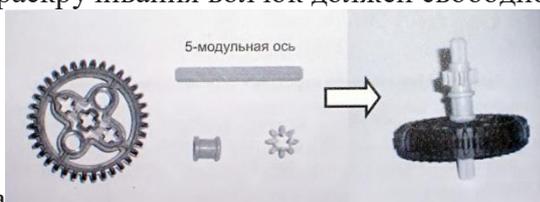
Что такое волчок?

Каждый из нас с детства знаком с замечательной игрушкой - юлой. Несколько движений - и юла около минуты держит вертикальное положение. С волчком еще проще. Стержень, закрепленный на нем диск, быстрое круговое движение пальцами, и с волчком происходит то же самое, что и с юлой, только без какого-либо механизма. Но так или иначе через некоторое время волчок падает, исчерпав заложенный в нем ресурс.

Ваша задача создать механизм, который многократно увеличит начальную скорость вращения волчка. Вторая задача - максимально продлить время вращения волчка.

Требования к волчку и механизму:

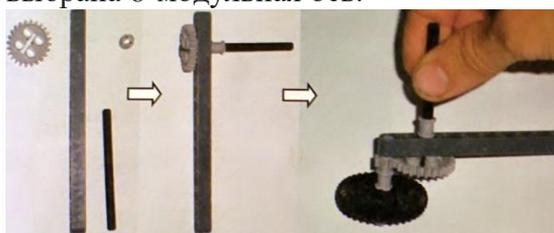
- волчок должен иметь ось вращения и достаточно тяжелый диск, маховик, который сохранит инерцию вращения;
- центр тяжести волчка должен быть расположен достаточно низко, но не слишком, чтобы края диска не цеплялись за поверхность стола (или пола);
- на оси вращения волчка необходимо установить шестерню для начального сцепления с механизмом;
- на механизме должны присутствовать две детали: для удержания одной рукой и придания вращения другой;
- в момент раскручивания волчок должен иметь плотное соприкосновение с механизмом;
- сразу после раскручивания волчок должен свободно отделяться от механизма.



Построение волчка

Маховик волчка может быть **любым колесом**, не обязательно зубчатым. Волчок с более тяжелым маховиком вращается дольше.

Простейший механизм для запуска с передаточным отношением 1:3. В качестве ведущей выбрана 8-модульная ось.

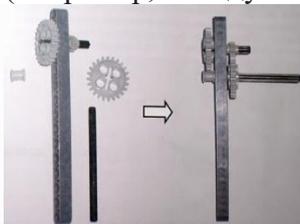


При запуске необходимо быстро повернуть ведущую ось на механизме и сразу поднять его, чтобы волчок вращался свободно.

Задание. Самостоятельно постройте механизм запуска с передаточным отношением 1:5.

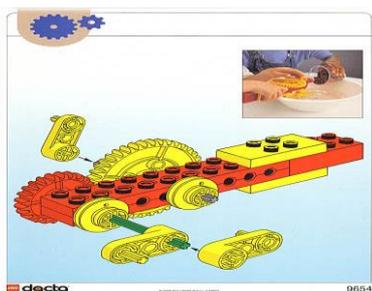
Построение волчка с двухступенчатой передачей

Схема двухступенчатой передачи 1:9. Следует заменить первую ось на более короткую (например, 4-модульную), а вместо втулки поставить малую 8-зубчатую шестеренку.



6. Механическая передача. Ручной миксер.

- Познакомить с понятиями “ведущее колесо”, “ведомое колесо”.
- В ходе практической работы исследовать, как меняются скорости вращения элементов механизмов при использовании зубчатых колёс.
- Расширить словарный запас детей.
- Учить следовать инструкции.



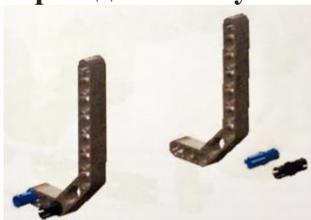
7. Тележки. История колеса. Одномоторная тележка.

Требования к конструкции.

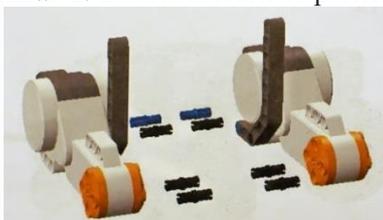
Это самая распространенная разновидность роботов. Тележка может быть с тремя точками опоры, две из которых - ведущие колеса, а третья - волокуша, или свободно вращающееся колесико.

Процесс построения одной из наиболее простых тележек

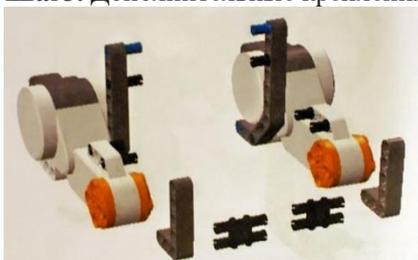
Шаг 1. Для крепления моторов, которые будут располагаться по обе стороны от NXT, берем две изогнутые балки:



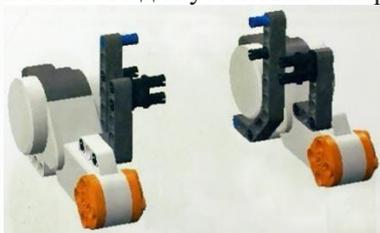
Шаг 2. В зависимости от расположения балок центр тяжести тележки может быть смещен:



Шаг3. Дополнительные крепления для придания устойчивости:



Все готово для установки контроллера:



Шаг 4. Колеса устанавливаются на 6-модульные оси, втулки предохраняют от нежелательного трения о корпус.



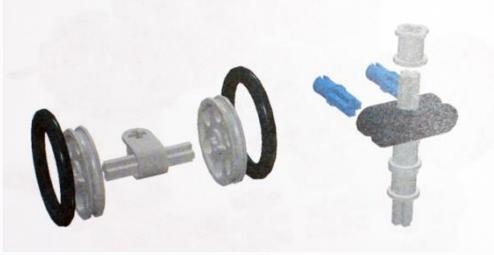
Шаг 5. Конструкция для крепления третьего колеса:



Шаг 6. Элементы подвижного колеса. Длины осей - 3 и 5 модулей. Колесо должно вращаться свободно.



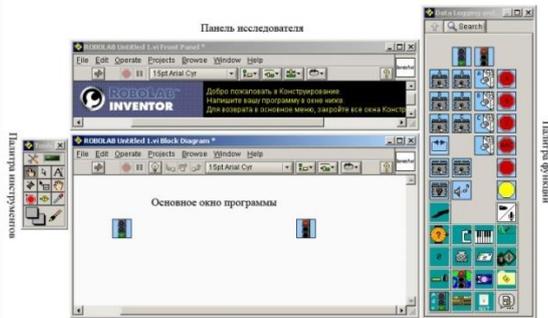
Сборка заднего подвижного колеса. Оси должны вращаться свободно.



Тележка готова.



8. Знакомство со средой Robolab 2.9



Палитра функций включает в себя различные типы команд:

- Команды действия
- Команды ожидания
- Управляющие структуры
- Модификаторы
- Команды в новых разделах для NXT встречаются всех типов



Простейшие

алгоритмы

1. Квадрат. По умолчанию считаем, что левый мотор подключен к порту В, правый - к порту С. Проверьте подключение.

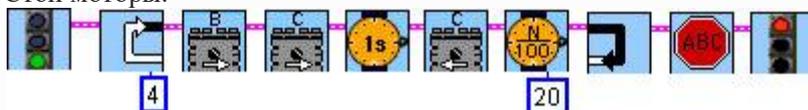
Повторяй

4

раза:

- моторы В, С вперед;
- жди N/100 секунды;
- мотор С назад (В продолжает двигаться вперед);
- жди N/100 секунды.

Стоп моторы.



2. Парковка. Напишите самостоятельно программу парковки в гараж (конструкцию строим из подручных материалов).

Шагающий робот.

Требования к конструкции

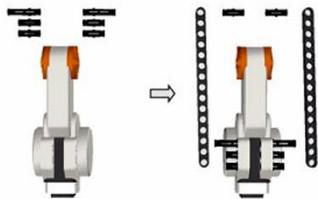
Наиболее подходящим для начального изучения представляется четвероногий робот. Робот на четырех ногах устойчив, как табурет. При правильном соединении он будет двигаться по прямой линии, притопывая подобно маленькому слонику. Его первая задача проста - идти вперед.

Требования к конструкции:

- механизм должен стоять на поверхности, упираясь только на четыре конечности, каждая из которых не может совершать вращательное движение вокруг одного из центра;
- движение конечностей должно быть возвратно-поступательным;
- в конструкции робота запрещено использование колес, соприкасающихся с поверхностью земли;
- конечности робота приводятся в движение одним мотором с помощью механической передачи;
- мотор присоединен к источнику питания;
- центр тяжести робота должен быть смещен вперед по ходу движения.

Процесс сборки

Шаг 1. Единственный привод обеспечит прямолинейное движение робота:



Шаг 2. 15-ти модульные балки крепятся к мотору и к угловым балкам из дополнительного набора. NXT крепится к угловым балкам:



Шаг 3. Ведущие оси будут разной длины: 5 и 6-ти модульные. Они вставляются в мотор с двух сторон так, чтобы из соответствующих балок выступали части осей длиной ровно 1 модуль.



Шаг 4. На ведущие оси насаживаются 24-зубые шестерни. Остальные штифты крепятся на бежевые штифты-полуси. Синие для этой цели не подойдут, поскольку создают очень большое трение.



Шаг 5. Выравнивание крайних шестеренок. Все 4 основные шестерни (24 зуба) должны быть расположены так, чтобы пары отверстий на них были строго параллельны друг другу. Расположение 40-зубых центральных шестерней не играет роли.

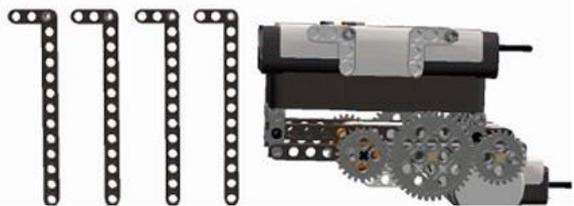


Шаг 6. Часть из восьми гладких штифтов вставляется а 24-зубые шестеренки симметрично относительно центра большой шестерни. И ровно наоборот с другой стороны: то есть если

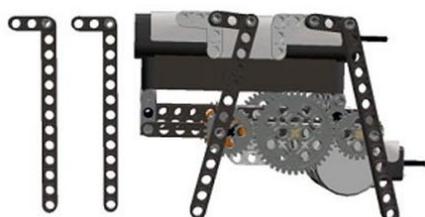
справа от NXT они были ближе к центру, то слева должны быть дальше от центра. Штифты серые!!!



Шаг 7. Ноги робота должны быть одинаковыми, но их длину можно менять.



Установка конечностей:



Шаг 8. Для запуска шагающего робота потребуется такая же простая программа, как и для запуска тележки. Однако следует обратить внимание, что в предложенной конструкции мотор должен вращаться не вперед, а назад. Если запустить его вперед, робот будет пританцовывать на месте, почти не перемещаясь. Программа в LegoMinstorms NXT программа будет выглядеть следующим образом:



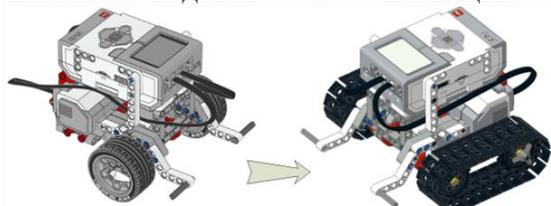
```

Backward
Empty
Backward
Empty
Loop
    
```

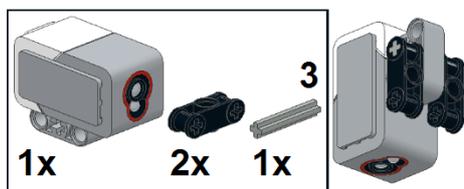
Итак, робот пошел. Теперь вам предоставляется свобода творчества. Варьируя длину конечностей, размер "плеча" и "колена", смещение центра тяжести, можно получить самые неожиданные эффекты в "походке" робота. Можно нацепить "башмачки".

9. Приводная платформа EV3 на гусеничном ходу.

В образовательном наборе конструктора MindstormsEducation EV3 всё обучение робототехники в классе ведётся с помощью приводной платформы, на колёсном ходу.



Получилась вот такая платформа. Высота платформы уменьшилась примерно на 4,5 мм, поэтому датчик цвета нужно поднимать на 1 деление, например, чтобы идти по линии. Конечно, он тогда получается высокомерно, но всё равно работает. Вместо стандартного крепления к приводной платформе, нужно будет сделать, например, вот такое:



Знакомство со средой программирования Robolab2.9. Режим «Администратор». Режим «Программист».

Среда программирования **RoboLab** фирмы LEGO Dacta A/S позволяет легко понять основы робототехники и научиться конструировать умные управляемые машины. В 1997 году, когда NASA рассекретила свою программу SOJOURNER ROVER'S, выяснилось, что для обеспечения ориентации, приземления и функционирования космических аппаратов применялась программа **LabVIEW** National Instruments (Техас, США). LabVIEW - это мощная среда программирования, используемая инженерами и учёными в исследовательских институтах и промышленности. Также LabVIEW является основой для RoboLab.

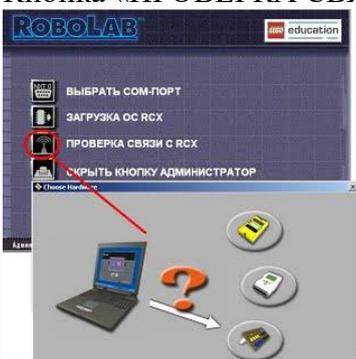
Режим «Администратор»

Режим "Администратор" служит для настройки программы **RoboLab** и ЛЕГО-микрокомпьютера **NXT**.



Как и любой другой компьютер, ЛЕГО-микрокомпьютер NXT управляется своей операционной системой. В новый NXT или NXT, оставленный без батареек более чем на 30 секунд, должна быть загружена операционная система. Для этого необходимо щелкнуть по кнопке «ЗАГРУЗКА ОС RCX», а затем кнопку «Загрузка» в появившемся окне. При этом NXT должен быть включен и подключен с помощью специального USB-кабеля к ПК. Передача ОС осуществляется примерно 5 минут, в течение которых нельзя выключать и передвигать NXT. После успешной загрузки ОС появиться окошко «Успешное соединение».

Кнопка «ПРОВЕРКА СВЯЗИ С RCX» служит для выбора устройства, с которым соединяется ПК



Режим «Программист»

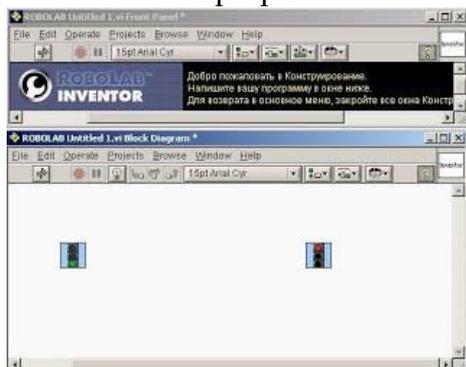
Этот режим позволяет создать и проверить работу программ. Здесь можно выбрать два режима: «Управление» и «Конструирование», а также выбрать уровень сложности программ от 1 до 4.

Режим «Управление» использует простые шаблоны, которые можно менять по своему усмотрению. Это позволяет быстро освоить принцип программирования и управления основными элементами модели (моторы, лампы, датчики).

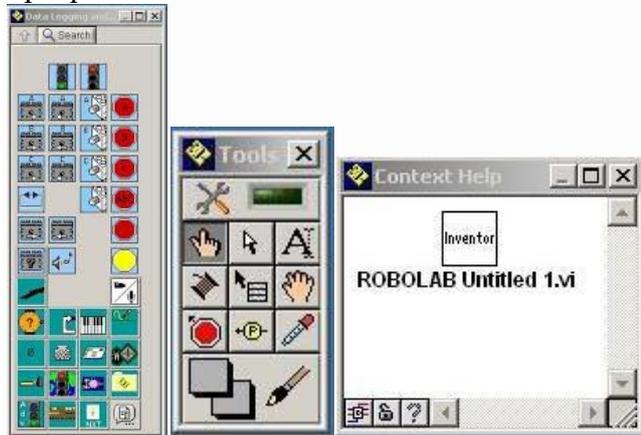
В Режиме «Конструирование» (Invertor4) программирование осуществляется путем

подбора и размещение на программном листе пиктограмм из палитры команд и путем связывания этих пиктограмм в одну командную линию (программу) с помощью инструмента «катушка» из палитры инструментов.

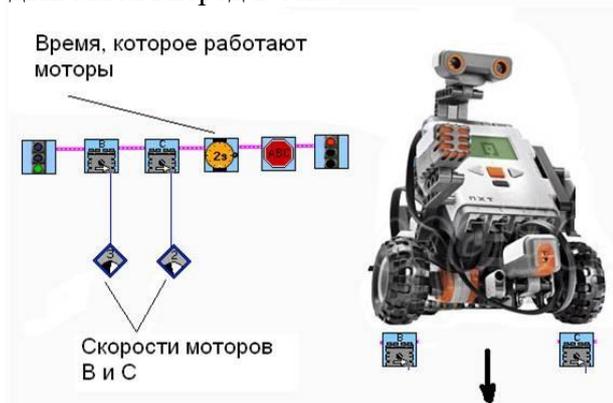
Рабочее окно программы:



Программа составляется из блоков-команд, которые соединяются линиями с помощью инструмента "катушка". Эти блоки желательно располагать на одной линии на одинаковом расстоянии друг от друга между светофорами. Зеленый - начало программы, красный - конец программы.



Так выглядит программа, позволяющая роботу с моторами, подключенными к портам В и С двигаться вперед 2 сек.:



Готовую программу надо загрузить в NXT. Для этого надо нажать в РобоЛаб кнопку **Run**, а на экране NXT выбрать меню MyFiles. Нажимая несколько раз на оранжевую кнопку, выбирая последнюю загруженную программу, затем в меню выбирают команду **Run**.

10. Типы команд. Команды действия. Базовые команды.

- Блоки Functions Palette можно классифицировать следующим образом:
- Команды действия
- Команды ожидания
- Управляющие структуры
- Модификаторы.
- Начнем с простейших команд. Их можно разделить на два типа: Жди и Делай.

- Команды типа «Делай» посылают управляющий сигнал на одно из устройств управления микроконтроллера. Например, «включить моторы», «остановить моторы», «издать звуковой сигнал», «отправить сообщение», «обнулить таймер» и т.п. Это действие, как правило, выполняется практически мгновенно (за исключением звуковых сигналов) после его программа переходит к следующему блоку. Следует заметить, что включенный мотор продолжает работать до тех пор, пока не выполниться команда выключения или программа не закончиться.

- Команды типа «Жди» ведут себя иначе. Они не выполняют никакого ощутимого действия, хотя и активно взаимодействуют с оборудованием NXT. Эти команды останавливают ход выполнения программы (точнее, задачи) в ожидании некоторого события. Как только событие происходит, управление переходит к следующей команде. Примеры таких команд: «жди громкого звука», «жди яркого света». «жди заданного времени» и т.д. Во время выполнения команды «Жди» все запущенные ранее процессы (включенные моторы и т.д.) продолжают работать.

- Следующий тип блоков – это управляющие структуры. Среди них представлен основной джентельменский набор программиста:

Ветвления

Циклы и безусловные переходы

Подпрограммы

Параллельные задачи

Обработчики событий.

И, наконец, вспомогательный тип блоков – модификаторы. Они по сути являются параметрами для выполнения различных команд и прикрепляются к блокам команд разноцветными проводами.

Практически все блоки, оставшиеся в Robolab со времен RCX, пригодны для NXT. Но при этом появилась часть новых блоков, относящихся преимущественно к датчикам NXT, а также расширяющий функциональные возможности Robolab для обеих систем. Следует заметить, что между датчиками, моторами и микроконтроллерами сохранена совместимость «снизу вверх», так что практически во всех случаях можно использовать старые датчики и моторы с новым микроконтроллером. Только для этого нужны соответствующие переходники, которые входят в состав набора 9797, а также продаются отдельно.

Команды действия.

Команды действия применимы к управляющим устройствам NXT. К ним можно отнести:

Моторы

Динамик

Таймеры

Встроенная память (контейнеры)

Датчики при инициализации и др.

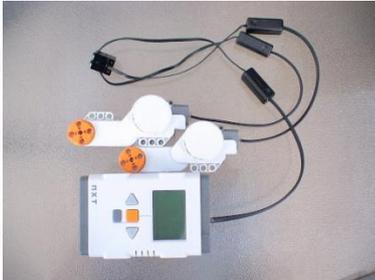
11. Моторы NXT.



В комплекте с базовым образовательным набором LegoMindstorms (#9797 LEGO EducationBasekit) идет кабель-переходник (#8528 NXT Convertercable), предназначение которого - предоставить возможность подключить RCX моторы и сенсоры к NXT блоку. Но у него обнаруживается и другое применение. Что если RJ-разъем каждого кабеля включить не в блок, а моторы, а свободные контактные элементы (в виде пластины 2x2 кнопки) соединить между собой.



Теперь нужно взять еще один кабель-переходник и его уже подключить к блоку. А оставшийся свободный контакт-пластину подключить к связке моторов:



Если сейчас подать напряжение на разъем, к которому подключены связка двигателей, то начнут двигаться и тот и другой. При этом, меняя расположение соединительных пластин, можно заставить двигаться моторы в разном направлении.

У данного подхода тоже есть ограничения:

- Моторы нельзя заставить двигаться по отдельности
- В программе нельзя больше сказать - "мотор сделай 2 оборота" или "мотор повернись на 90 градусов". Но остается все еще возможность управлять моторами по времени - "мотор двигайся в течение 5 секунд"

Автор сего способа описывает при этом работу с такими моторами следующим образом:

- Когда вы хотите включить мотор, используйте в NXT-G блок работы с сенсором расстояния (ultrasonic), настроенный на работу с портом, куда подключен мотор
- Когда вы хотите выключить мотор, используйте блок сенсора касания, также настроенный на работу с этим портом

Когда программа встречает блок работы с сенсором расстояния, блок начинает подавать напряжение на порт и мотор начинает вращаться. Когда встречается блок работы с сенсором касания - напряжения на порт перестает подаваться.

Ограничения этого способа - нельзя менять направление вращения. Но, в случае чего, это не должно быть проблемой - тут поможет использование шестерней.

12. Кегельринг. Танец в круге.

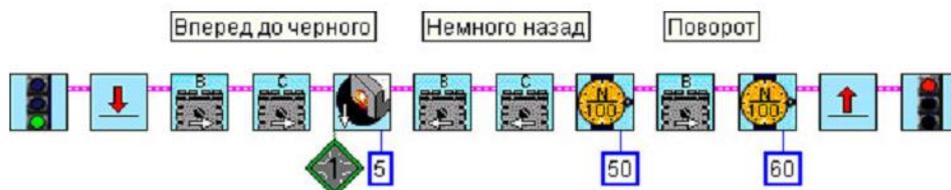
Робот живет внутри круга, за пределы которого нельзя выходить. Для выполнения этой задачи надо собрать стандартную трехколесную тележку: два передних колеса ведущие, одно заднее подвижное на шарнире. Спереди по центру расположен датчик освещенности, направленный строго вниз и находящийся на расстоянии 5-10 мм от пола.



Робот ставится в центре и при старте должен двигаться внутри круга, не выходя за его пределы.

Алгоритм действий "Танец в круге" таков:

1. ехать вперед, пока показания датчика не понизятся на 5 пунктов (лучше 10);
2. отъехать немного назад (полсекунды);
3. развернуться примерно на 120-150 градусов (тоже по времени);
4. повторять пункты 1-3 бесконечно.



Параметры, указанные в модификаторах, подберите самостоятельно: степень понижения освещенности на черной линии, время отъезда назад и время поворота.

2. Вытолкнуть все банки (кегельринг)

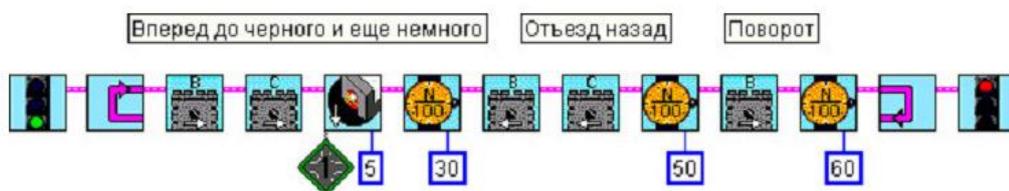
Несколько пластиковых стаканчиков или пустых жестяных банок расставлены внири круга, за черной линией на расстоянии 12-15 см от нее - это мусор, от которого необходимо очистить круг за кратчайшее время. Первые попытки запуска робота покажут несколько **недостатков**:

1. стаканчики попадают под колеса, падают и плохо выталкиваются;
2. даже вытолкнутые стаканчики остаются частично внутри круга, поскольку, увидев край, робот сразу устремляется назад;
3. робот ведет себя как слон в посудной лавке;
4. робот делает много движений впустую.

Избавимся от первого недостатка. Для этого построим бампер шириной 20-25 см рядом с датчиком освещенности.



Второй и третий недостатки устраняются программно. Пусть увидев край, тележка еще немного двигается вперед, выталкивая стаканчик, и только после этого объезжает внутрь круга.

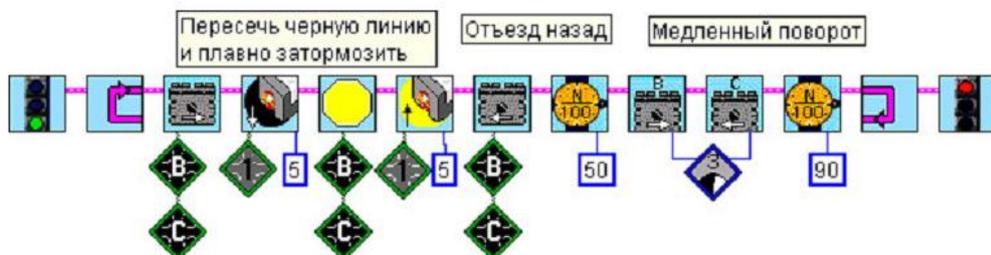


Самый надежный способ заехать точно на пределы черной линии - это дождаться значения белого на датчике освещенности. Поэтому время можно заменить на "ожидание белого". Для экономии места стоит сгруппировать команды управления моторами, а также использовать "реверс" при смене направления на работающих моторах.



Теперь стоит поработать над точностью движения, по возможности не теряя скорости. В зависимости от конструкции робота, при резкой смене направления он может потерять

равновесие или просто "встать на дыбы" на передние колеса. Поэтому последние несколько сантиметров можно проехать на торможении по инерции, то есть полностью освободив моторы. И второе. Точность поворота будет зависеть от того, какие команды подаются на моторы и по какому принцип рассчитываются длительности поворота. К сожалению, таймер - не надежный помощник. По инерции на малых промежутках времени робот может поворачиваться на различные углы. Мы можем пожертвовать реверсом в последней команде управления мотором В, для того чтобы достичь неторопливого движения обоими моторами. Длительность поворота при этом немного возрастет.



Для точности управления моторами необходимо использовать другой тип команд: с контролируемым вращением. Эти команды находятся в разделе "AdvancedOutputControl" и позволяют задавать мощность моторов от -100 до 100. В новом примере для компактности разместим все числовые параметры сверху, а модификаторы портов снизу.



13. Движение вдоль линии. Один датчик.

На базовую тележку надо прикрепить датчик расстояния и подключить его на 1 порт. Датчик расстояния слегка выносится вперед:

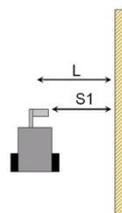


Ориентацию датчика можно варьировать:



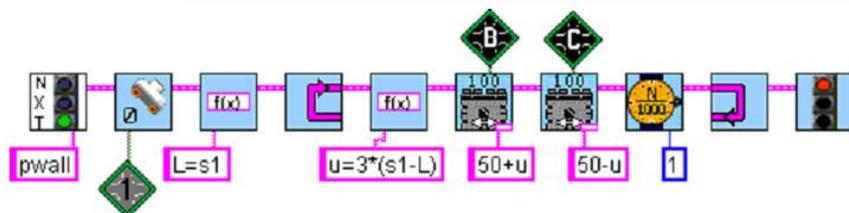
Движение вдоль стенки на П-регуляторе

Решим такую задачу. Робот должен двигаться вдоль стенки на заданном расстоянии L . Предположим, что левое колесо робота управляется мотором В, правое - мотором С, а датчик расстояния, подключенный к порту 1, закреплен несколько впереди корпуса тележки (это



важно!) и направлен на стену справа по ходу движения.

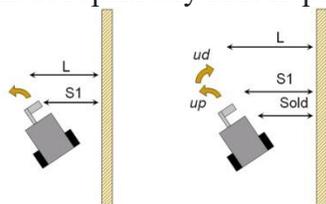
Расстояние до стенки в настоящий момент времени, которое показывает датчик, обозначим $S1$. Измеряется она в сантиметрах. Моторы двигаются со средней скоростью 50% от максимума, но при отклонении от заданного курса на них осуществляется управляющее воздействие u (на мотор В $50+u$, на мотор С $50-u$): $u=k*(S1-L)$, где k - некий усиливающий коэффициент, определяющий воздействие регулятора на систему. Таким образом, при $S1=L$ робот не меняет курса и едет прямо. В случае отклонения его курс корректируется. Для робота NXT средних размеров коэффициент k может колебаться от 1 до 10 в зависимости от многих факторов. Подберите его самостоятельно.



В данном случае П-регулятор будет эффективно работать только при малых углах отклонения. Кроме то, движение практически всегда будет происходить по волнообразной траектории. Сделать регулирование более точным позволит введение новых принципов, учитывающих отклонение робота от курса.

Пропорционально-дифференциальный регулятор

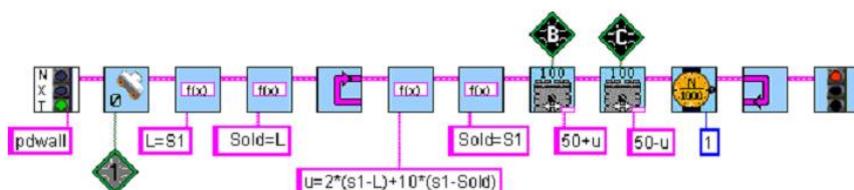
В некоторых случаях П-регулятор может вывести систему из устойчивого состояния.



Например, если робот направлен от стенки, но находится по отношению к ней ближе заданного расстояния, на моторы поступит команда еще сильнее повернуть от стенки, в результате чего с ней может быть утерян контакт (датчик расстояния получает отраженный сигнал практически только от перпендикулярной поверхности).

Для защиты от подобных ситуаций добавим в регулятор дифференциальную составляющую, которая будет следить за направлением движения робота.

$u=k1*(S-L)+k2*(s1-Sold)$, где $Sold$ - расстояние на предыдущем шаге.



Необходимо подобрать подходящие значения коэффициентов $k1$ и $k2$. Обычно подбор начинается с пропорционального коэффициента ($k1$) при нулевом дифференциальном ($k2=0$). Когда достигнута некоторая стабильность на небольших отклонениях, добавляется дифференциальная составляющая.

Конструкция робота с датчиком расстояния, расположенным под углом

Описанный выше робот может объезжать стены только при малых отклонениях от прямой линии. Рассмотрим вариант, при котором на пути движения будут возникать серьезные повороты, вплоть до прямых углом. Потребуется внести модификации и в конструкцию, и в программу.

Во-первых, робот должен будет смотреть не только направо, но и вперед. Ставить второй дальномер довольно затратно. Однако можно воспользоваться эффектом того, что ультразвуковой датчик имеет расширяющуюся область видимости.

Это напоминает угловое зрение человека: кое-что он может увидеть краем глаза. Стоит воспользоваться таким свойством и разместить датчик расстояния не перпендикулярно курсу движения, а под острым углом. Так можно убить сразу двух зайцев. Во-первых, робот будет

видеть препятствия спереди, во-вторых, более стабильно будет придерживаться курса вдоль стены, постоянно находясь на грани видимости. Таким образом, без добавления новых устройств будет получено более эффективное использование возможностей датчика.

Важное замечание. При старте робота его надо будет направлять датчиком строго на стену, чтобы процесс считывания начального значения прошел без помех. Крепление для датчика размещается на левой стороне:



Как и впервой конструкции, датчик располагается вертикально:



Увеличенное за счет корпуса робота расстояние до стены способствует расширению области обзора:



Очевидно, что изменение конструкции влечет изменение коэффициентов регулятора k_1 и k_2 . Обычно подбор начинается с пропорционального коэффициента при нулевом дифференциальном. Когда достигнута некоторая стабильность на небольших отклонениях, добавляется дифференциальная составляющая.

14. Творческое конструирование собственной модели. Программирование.

15. Защита модели.

КАЛЕНДАРНЫЙ УЧЕБНЫЙ ГРАФИК 3 год обучения

Содержание		Месяцы обучения								
		Сен- тябрь	Ок- тябрь	Но- ябрь	Де- кабрь	Ян- варь	Фев- раль	Март	Ап- рель	Май
	Вводное занятие. Инструктаж по технике безопасности.	2								
	Повторение второго года обучения.	4								
	Способы крепления деталей. Высокая башня.	16								
	Механический	2	14							

	манипулятор (хваталка).									
	Механическая передача. Волчок.		10	6						
	Механическая передача. Ручной миксер.			16						
	Тележки. Одномоторная тележка.			2	14					
	Знакомство со средой Robolab 2.9. Шагающий робот.				10	6				
	Приводная платформа EV3 на гусеничном ходу					16				
	Типы команд. Команды действия. Базовые команды.					2	14			
	Моторы NXT.						10	6		
	Кегельринг. Танец в круге.							16		
	Движение вдоль линии. Один датчик.							2	14	
	Творческое конструирование собственной модели. Программирование.								10	6
	Защита модели.									14
	Экскурсии в объединения.									2
	Итоговое занятие									2
	Итого:	24	24	24	24	24	24	24	24	24

СФОРМИРОВАННЫЕ УМЕНИЯ И НАВЫКИ:

- работать с литературой, с журналами, с каталогами, в интернете (изучать и обрабатывать информацию);
- самостоятельно решать технические задачи в процессе конструирования роботов (планирование предстоящих действий, самоконтроль, применять полученные знания, приемы и опыт конструирования с использованием специальных элементов и т.д.);
- создавать действующие модели роботов на основе конструктора ЛЕГО; создавать программы на компьютере на основе компьютерной программы Robolab;
- передавать (загружать) программы в RCX и NXT;
- корректировать программы при необходимости;
- демонстрировать технические возможности роботов;
- излагать логически правильно действие своей модели (проекта).

Методическое обеспечение дополнительной образовательной программы

Основные формы занятий:

- теоретическая часть занятий;
- практическая часть занятий

ФОРМЫ РАБОТЫ С УЧАЩИМИСЯ:

- игры (дидактические, словесные, театрализованные, сюжетно-ролевые, творческие, подвижные);
- экскурсии в объединения;

- викторины, конкурсы, соревнования;
- беседы познавательные, обобщающие, сопровождающие;

УСЛОВИЯ РЕАЛИЗАЦИИ ПРОГРАММЫ

Возраст обучающихся в реализации данной дополнительной общеобразовательной программы 7 - 12 лет.

Программа рассчитана на обучение и воспитание детей и подростков 7-12 лет. Количество воспитанников в группах – до 15 человек.

При необходимости могут формироваться разновозрастные группы. Для занятий объединения требуется просторное, светлое помещение, отвечающее санитарно-гигиеническим нормам.

Занятия в 1 года обучения организуются 2 раза в неделю по 2 часа. Занятия 2, 3 и последующих лет обучения организуются – 3 раза в неделю по 2 часа.

Занятия, предусмотренные программой, включают теоретические и практические формы работы с обучающимися.

На занятии используются все формы обучения: коллективные, групповые, индивидуальные, что обеспечивает дифференцированный подход с учетом степени освоения обучающимися программы.

Методы организации процесса обучения: беседы, практические занятия, создание ситуации успеха, работа над проектами.

Программа предусматривает участие в соревнованиях и выставках по робототехнике различного уровня, от соревнований внутри объединения до всероссийских.

ДОКУМЕНТЫ И МАТЕРИАЛЫ, С УЧЕТОМ КОТОРЫХ СОСТАВЛЕНА ПРОГРАММА

1. Федеральный Закон «Об образовании в Российской Федерации» от 29.12.2012 №273-ФЗ;
2. Закон Республики Татарстан «Об образовании» от 22.07.2013 №68-ЗРТ;
3. Модельный стандарт качества муниципальной услуги по организации предоставления дополнительного образования детей в многопрофильных организациях дополнительного образования от 20.03.14 №1465/14;
4. Санитарно–эпидемиологические правила и нормативы СанПиН 2.4.4.3172-14;
5. Устав МБОУДО «Станция юных техников»; Должностная инструкция педагога дополнительного образования.

ОЖИДАЕМЫЕ РЕЗУЛЬТАТЫ РАБОТЫ ПО ПРОГРАММЕ

1. ПРОГНОЗИРУЕМЫЕ РЕЗУЛЬТАТЫ И СПОСОБЫ ИХ ПРОВЕРКИ

Результаты обучающихся

1-й год обучения

принципы и технологию сборки конструктора LEGO WeDo.

названия деталей из LEGO® WeDo™ (LEGO EducationWeDoSoftware):

принципы работы датчиков (коммутатор LEGO® USB Hub, мотор, датчик наклона, датчик движения.)

Должны уметь:

- оживление LEGO моделей.
- работать с цифровыми инструментами.
- создавать простые программы для управления.

2. ПРОГНОЗИРУЕМЫЕ РЕЗУЛЬТАТЫ И СПОСОБЫ ИХ ПРОВЕРКИ

Результаты обучающихся

2-й год обучения

Должны знать:

- принципы и технологию сборки LEGO роботов;
- названия деталей из LEGO набора «Перворобот» и ресурсного набора;
- принципы работы датчиков;

- линейные программы, простые программы с ветвлением и циклами в среде программирования NXT – G;
- правила организации рабочего места и необходимые правила техники безопасности в процессе всех этапов конструирования.
- Должны уметь:
- самостоятельно строить LEGO роботов по технологическим картам;
- определять основные части изготавливаемых моделей и правильно произносить их названия;
- создавать простые программы для управления роботами.

3. ПРОГНОЗИРУЕМЫЕ РЕЗУЛЬТАТЫ И СПОСОБЫ ИХ ПРОВЕРКИ

Результаты обучающихся

3-й год обучения

Должны знать:

- правила безопасной работы;
- основные компоненты конструкторов;
- конструктивные особенности различных моделей, сооружений и механизмов;
- компьютерную среду, включающую в себя графический язык программирования;
- виды подвижных и неподвижных соединений в конструкторе;
- основные приемы конструирования роботов;
- конструктивные особенности различных роботов;
- как передавать программы в RCX и NXT;
- как использовать созданные программы;
- приемы и опыт конструирования с использованием специальных элементов, и других объектов и т.д.).

СПИСОК ИНФОРМАЦИОННЫХ РЕСУРСОВ ИНТЕРНЕТ-РЕСУРСЫ ДЛЯ ОРГАНИЗАЦИИ РАБОТЫ ПО РОБОТОТЕХНИКЕ В ОБРАЗОВАТЕЛЬНЫХ ОРГАНИЗАЦИЯХ

1. <http://www.prorobot.ru>
2. <http://www.nnxt.blogspot.ru>
3. <http://www.ielf.ucoz.ru>
4. <http://www.fiolet-korova.ru>
5. <http://www.mindstorms.ru>
6. <http://www.lego56.ru>
7. <http://www.robot-develop.org>
8. <http://www.lego.detmir.ru>