
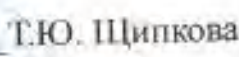


Рассмотрено
На заседании Управляющего
Совета Протокол № 1
От 30.08 2017
Председатель

 Ю.В. Владимиров

Утверждено
Приказом № 3
От 04.09 2017
Директор

 Е.Ю. Щипкова



Согласовано
С профсоюзным комитетом
Протокол № 1
От 01.09 2017
Председатель

 Н.Б. Стуловская

**Государственное бюджетное общеобразовательное учреждение города Москвы
«Романовская школа»**

Дополнительная общеобразовательная (общеразвивающая) программа

Направленность программы – техническая

Форма организации образовательной деятельности – кружок

Название кружка – Леготехника, 4 класс

Уровень программы – ознакомительный

Возраст учащихся – 9-12 лет

Срок реализации программы – 1 год

Автор-составитель программы:
Новосельский Алексей Кириллович,
педагог дополнительного образования

Пояснительная записка

Настоящий курс предлагает использование образовательных конструкторов LEGO и аппаратно-программного обеспечения как инструмента для обучения школьников конструированию, моделированию и компьютерному управлению во время внеурочной деятельности в 4-х классах.

Основными задачами курса являются:

- ✓ ознакомление с основными принципами механики;
- ✓ ознакомление с основами программирования в компьютерной среде моделирования ПЕРВОРОБОТ/ROBOLAB 2.5.4;
- ✓ развитие умения работать по предложенным инструкциям;
- ✓ развитие умения творчески подходить к решению задачи;
- ✓ развитие умения довести решение задачи до работающей модели;
- ✓ развитие умения излагать мысли в четкой логической последовательности, отстаивать свою точку зрения, анализировать ситуацию и самостоятельно находить ответы на вопросы путем логических рассуждений.

Обоснование курса

Работа с образовательными конструкторами LEGO позволяет школьникам в форме познавательной игры узнать многие важные идеи и развить необходимые в дальнейшей жизни навыки. При построении модели затрагивается множество проблем из разных областей знания – от теории механики до психологии, – что является вполне естественным.

Очень важным представляется тренировка работы в коллективе и развитие самостоятельного технического творчества. Простота в построении модели в сочетании с большими конструктивными возможностями конструктора позволяют детям в конце урока увидеть сделанную своими руками модель, которая выполняет поставленную ими же самими задачу.

Изучая простые механизмы, ребята учатся работать руками (развитие мелких и точных движений), развивают элементарное конструкторское мышление, фантазию, изучают принципы работы многих механизмов. *Цель* первой части курса заключается в том, чтобы перевести уровень общения ребят с техникой «на ты», познакомить с профессией инженера; изучение понятий конструкции и ее основных свойствах (жесткости, прочности и устойчивости), элементов черчения.

Вторая часть курса предполагает использование компьютеров и специальных интерфейсных блоков совместно с конструкторами. Важно отметить, что компьютер используется как средство управления моделью; его использование направлено на составление управляющих алгоритмов для собранных моделей. Учащиеся получают представление об особенностях составления программ управления, автоматизации механизмов, моделировании работы систем. *Цель* второй половины курса состоит в том, чтобы научить ребят грамотно выразить свою идею, спроектировать ее техническое и программное решение, реализовать ее в виде модели, способной к функционированию.

Предлагаемый курс – это интегрированный курс, в котором помимо информационных

технологий задействованы:

- ✓ материальная технология (конструктор Лего),
- ✓ физика (механизм, оптика),
- ✓ биология,
- ✓ ОБЖ и многое другое.

Конструктор Лего и программное обеспечение к нему предоставляет прекрасную возможность учиться ребенку на собственном опыте. Такие знания вызывают у детей желание двигаться по пути открытий и исследований, а любой признанный и оцененный успех добавляет уверенности в себе. Учение происходит особенно успешно, когда ребенок вовлечен в процесс создания значимого и осмысленного продукта, который представляет для него интерес.

Важно, что при этом ребенок сам *строит свои знания*, а учитель лишь консультирует работу.

В окружающем нас мире очень много роботов: от лифта в вашем доме до производства автомобилей, они повсюду. Конструктор ПервоРобот приглашает ребят войти в увлекательный мир роботов, погрузиться в сложную среду информационных технологий. Программное обеспечение **Robolab** отличается дружественным интерфейсом, позволяющим ребенку постепенно превращаться из новичка в опытного пользователя.

Лего позволяет учащимся:

1. Совместно обучаться школьникам в рамках одной бригады;
2. Распределять обязанности в своей бригаде;
3. Проявлять повышенное внимание культуре и этике общения;
4. Проявлять творческий подход к решению поставленной задачи;
5. Создавать модели реальных объектов и процессов;

Лего-технологии позволяют выйти на новые образовательные результаты

- ✓ Умение работать в группе;
- ✓ Решать задачи практического содержания
- ✓ Моделировать и исследовать процессы;
- ✓ Переходить от обучения к учению
- ✓ Роль учителя меняется от наставника-тренера, к союзнику-помощнику.

Литература

1. Индустрия развлечений: ПервоРобот. Книга для учителя и сборник проектов. int.
2. Автоматизированные устройства: ПервоРобот. Книга для учителя. int.
3. MindStorms for schools. Educational division.
4. Наука. Энциклопедия. – М., «РОСМЭН», 2001. – 125 с.
5. Энциклопедический словарь юного техника. – М., «Педагогика», 1988.
6. www.school.edu.ru/int
7. <http://www.int-edu.ru>
8. CD ПервоРобот/RoboLab 2.5.4. Руководство пользователя. Int
9. CD ПервоРобот/RoboLab 2.5.4. Программное обеспечение. Int

Учебно-тематический план занятий.

№ занятия	Название темы	Количество часов (теория/практика)
1	Техника безопасности Роботы вокруг нас. Среда конструирования – повторение знаний о деталях конструктора, а также типовых соединениях.	1 час(1/0)
2-5	Управление 4 • Использование Датчика Освещенности в команде Жди • Создание многоступенчатых программ	4 часа(1/3)
6-9	Движение по траектории. Модель с одним и двумя датчиками света. Программирование.	4 часа(1/3)
10-13	Соревнования «Движение по линии» между группами, обсуждение проектов и программ	4 часа(2/6)
14-17	Проект Карусель. Использование автоматического управления. Программирование вращения на заданное количество времени, автоматическое освещение, срабатывающее на уменьшение освещенности объекта. Разбиение на задачи. Прыжки.	4 часа(1/3)
18-21	Понятие о простых механизмах и их разновидностях. Рычаг и его применение. Конструирование рычажных механизмов.	4 часа(1/3)
22-25	Рычаги: правило равновесия рычага. Основные определения. Правило равновесия рычага. Построение сложных моделей по теме «Рычаги».	4 часа(1/3)
26-29	Модель «шлагбаум». Использование зубчатой передачи для уменьшения скорости модели.	4 часа(1/3)
30-34	Датчики – органы чувств Робота. Шлагбаум-автомат. Использование датчика касания.	5 часов(1/4)
Всего за год 34 часа		

Занятие 1

Тема: Роботы вокруг нас. Среда конструирования - знакомство с деталями конструктора.

Оборудование: Конструктор Лего «Индустрия развлечений», компьютер учителя, подключенный к нему проектор.

В первой части урока детям демонстрируется презентация **Роботы вокруг нас**.

Занятие 6-9

Тема: Движение по траектории. Модель с одним и двумя датчиками света. Программирование.

Оборудование:

- конструктор 9786;
- для движения с 2 датчиками освещенности необходим второй датчик освещенности (есть в конструкторе 9794);
- ПК с установленной средой программирования Robolab 2.5.4.

Содержание:

Для преподавателя:

Занятие, на котором занимаемся почти одним программированием. Не так увлекательно, трудно, но очень интересно заставить робота двигаться вдоль черной линии – траектории. Можно заинтересовать детей тем, что заранее собрать и запрограммировать модель, затем показать её в действии. Ещё эффектнее это будет, если купить черный скотч, и создавать различные траектории прямо на глазах у детей. Когда робот в каждой ситуации найдет дорогу, то ученики ваши!

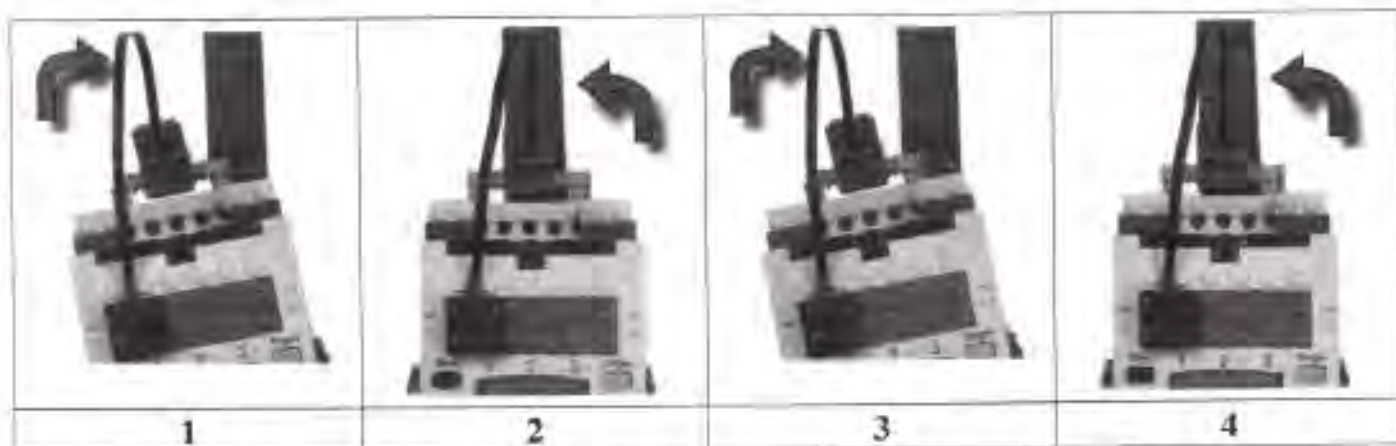
Для детей:

Один датчик освещенности:

На этом занятии мы будем собирать и программировать модель, которая может двигаться по черной линии. Существует несколько алгоритмов, по которым модель, отслеживая линию световыми датчиками, может двигаться по ней, не сбиваясь. Самый простой алгоритм называется "Отслеживание черного края".

Алгоритм, по которому будет двигаться модель с одним датчиком света, называется "Отслеживание черного края".



1. Сначала модель расположена так, что датчик света, который смотрит вниз находится на белой области рядом с черной линией, пусть например, с левой стороны.
2. Затем модель начинает двигаться вправо, до тех пор, пока не наедет датчиком на черную линию (т.е. пока датчик света не "увидит" темноту)
3. Потом модель начинает поворачивать влево, до тех пор, пока не наедет датчиком на белую область (т.е. пока датчик света не "увидит" свет)
4. Затем программа начинает работать с начала, т.е. в цикле.



Такой алгоритм позволит модели с одним датчиком света двигаться по черной линии.

Программа "Отслеживание черного края", написанная на уровне "Inventor 4" выглядит вот так:



Две пиктограммы  - красная метка и  - красный прыжок, в данной программе используются для создания бесконечного цикла, т.е. для того чтобы пиктограммы находящиеся между ними выполнялись достаточно долго, чтобы модель могла пройти всю линию до конца. Когда программа доходит до пиктограммы красный прыжок она начинает выполняться с пиктограммы красная метка.

Если бы в нашем алгоритме не использовались пиктограммы "красная метка и прыжок", наш алгоритм назывался бы линейным т.к. в этом случае пиктограммы-команды выполнялись одна за другой один раз.

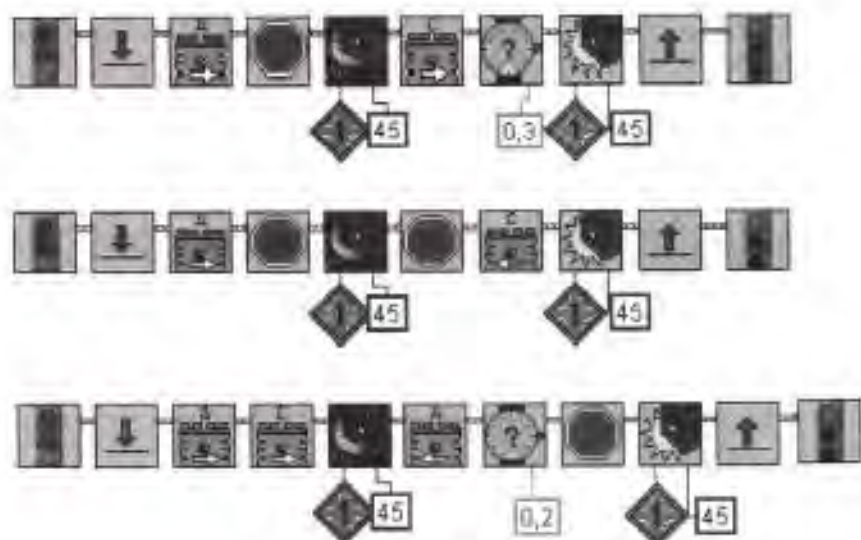
Алгоритм "Отслеживание черного края" работает так:

- включается двигатель, подключенный к порту А, при этом двигатель, подключенный к порту С выключен;
 - модель начинает поворачивать вираво; программа ждет срабатывания датчика света на темноту (меньше 45);
 - после чего двигатель А отключается,
 - включается двигатель С; модель начинает поворачивать влево, до срабатывания датчика света (больше 45);
 - красный прыжок на красную метку, программа начинает выполняться с начала.
- Значение освещенности 45 взято условно, в вашей программе оно должно соответствовать вашему полю и вашей освещенности.

А как можно сделать так, чтобы модель двигалась быстрее? Один из вариантов, это изменение программы.

Ниже предлагается 4 варианта измененной программы. Необходимо попробовать реализовать каждый из этих вариантов или все и решить, какой из вариантов эффективнее.





Два датчика освещенности:

В модели с двумя датчиками света возможны 4 варианта расположения датчиков относительно черной линии:

1. Оба датчика на белом, модель движется вперед.
2. Правый датчик на черном, левый на белом, модель поворачивает вправо.
3. Левый датчик на черном, правый на белом, модель поворачивает влево.
4. Оба датчика на черном (перекресток), модель едет прямо.

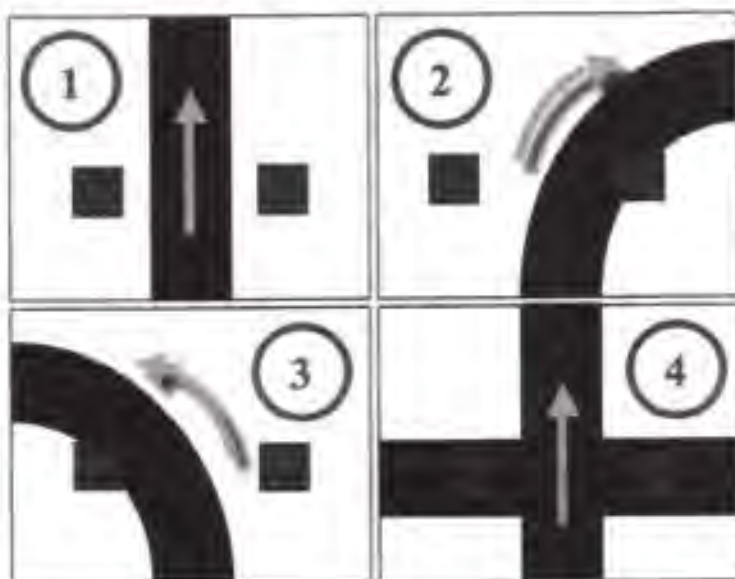
Для того чтобы программа могла проверить все 4 варианта и в зависимости от показаний датчиков изменить движение модели, ей необходимо проверять значения обоих датчиков одновременно.

Для этих целей лучше всего использовать такие структуры программирования как ветвления.

После того как собрана модель, ее нужно запрограммировать.

Запускаем **Robolab – Программист – Конструирование – Inventor 4.**



Ветвление - это структура, позволяющая выбрать одну из ветвей выполнения программы, в зависимости от выполнения или невыполнения какого-либо условия. Т.е. она позволяет выбрать один из вариантов, в зависимости от условий.

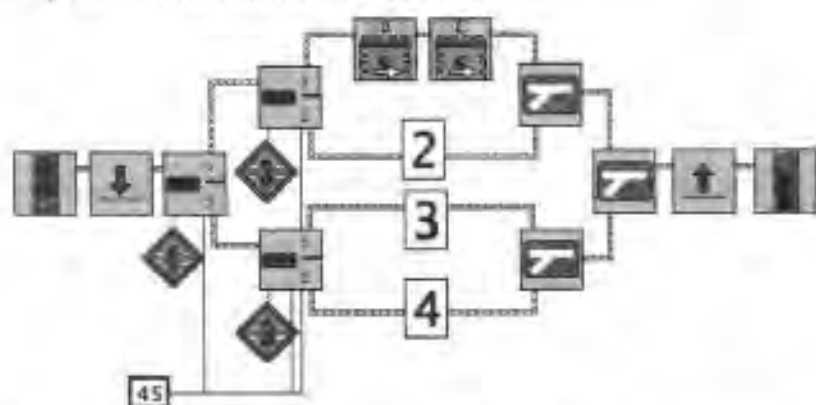


Для реализации ветвления, в палитре команд нажмите кнопку - структуры.

Затем нажмите кнопку . Перед вами появится окно, с вариантами ветвлений:



В котором вам нужно выбрать  - освещенность, т.к. к нашей модели подсоединены только датчики света. Ветки ветвления должны обязательно сливаться в одну. Это осуществляется размещением пиктограммы  - слияние ветвлений, после условий реализованных этим ветвлением.



После запуска этой программы, будет проверяться условие: если значение на датчике света, подключенного к порту 1, будет меньше 45, то выполнение программы пойдет по верхней ветви. Если же значение на датчике будет больше либо равно 45, то выполнится нижняя ветвь (значение датчика света можно посмотреть, нажав кнопку **View** на RCX, после хотя бы однократного запуска программы содержащей датчик света). Если выполнение программы пошло по верхней ветке то, проверяется значение освещенности на датчике, подключенном к порту №3, если значение освещенности и на нем больше 45 (т.е. оба датчика на белом), то начинают выполняться пиктограммы верхней ветви - двигатели включаются вперед (модель едет вперед). После этого ветвление закрывается (сливается) и программа начинает выполняться заново.

Попробуйте самостоятельно вместо цифр 2, 3, 4 вставить пиктограммы, символизирующие команды, которые должны выполняться в зависимости от расположения датчиков относительно черной полосы.

Варианты 2 и 3 должны осуществлять поворот, соответственно, в правую и в левую стороны, а вариант 4 будет точно таким же, как и 1 – движение прямо.



Хотя, конечно, здесь не все так просто. Поворот можно осуществлять самыми разными способами, и какой их них подойдет к конкретной конструкции и к конкретному освещению, сказать заранее с гарантией нельзя, поэтому здесь обязательно нужна отладка программы. Разобраться с этим способны дети, имеющие вкус к программированию, либо

Модель работает следующим образом:

- платформа вращается заданное количество времени, обеспечивая массовое удовольствие и радость участников аттракциона согласно купленным билетам.
- в вечернее время для безопасного получения приобретенного удовольствия площадка с аттракционом автоматически освещается. При повышении освещенности до стандартного состояния при помощи естественных источников (солнце, в Египте «Гелиос») искусственное освещение автоматически отключается (в целях экономии государственных ресурсов).
- наиболее одаренным учащимся можно предложить дополнительный вариант оснащения карусели музыкальным сопровождением

При программировании может использоваться режим Программирование, уровень Inventor 4.

Для детей:

Существует 4 уровня "Inventor"(Конструирование), они отличаются своими возможностями. Чтобы использовать все доступные возможности необходимо выбрать





уровень "Inventor 4".

После запуска уровня "Inventor 4" перед вами появится 3 окна: информационное окно, окно программы и окно "палитра команд". Для удобства работы необходимо




развернуть окно программы на полный экран.





В режиме "Inventor 4" в области программы находиться только две пиктограммы,

 -начало и  -конец программы. Остальные пиктограммы необходимые для работы программы необходимо брать из палитры команд (функции) и перетаскивать в рабочее окно программы. Для этого:




- щелкните указателем мыши на пиктограмме команды, которую вы хотите использовать.
- передвиньте курсор мыши в Окно программы и щелкните кнопкой мыши, чтобы поместить сюда выбранную команду. При этом возможно, помещенная пиктограмма автоматически свяжется проводом с рядом стоящей.

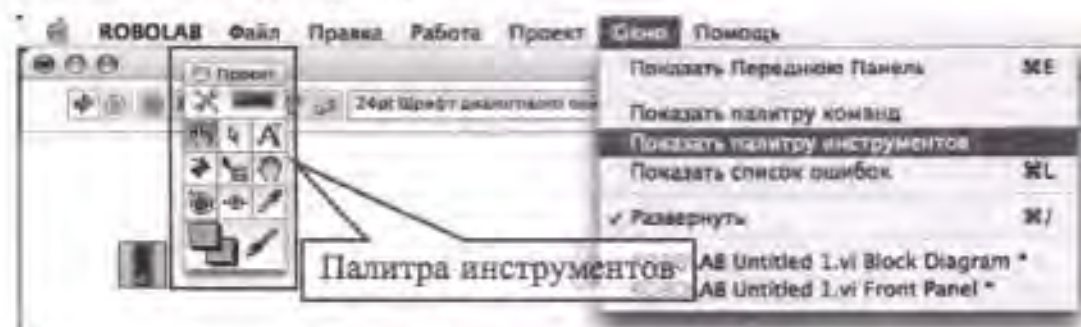
Если провод не соединился или соединился неправильно его необходимо создать, для этого нужно поменять вид курсора на "моток провода" , Проще всего это сделать, нажав 2 раза клавишу пробел клавиатуры компьютера.



Далее:

1	щелкните в правом верхнем углу левой пиктограммы	
2	переместите курсор к левому верхнему углу левой пиктограммы	
3	Получившееся соединение имеет вид розовой линии или черно-белого пунктира. Если соединение имеет вид пунктирной линии, выберите пункт "Удалить разорванные провода в меню "Правка", после этого попробуйте еще раз соединить пиктограммы.	
		

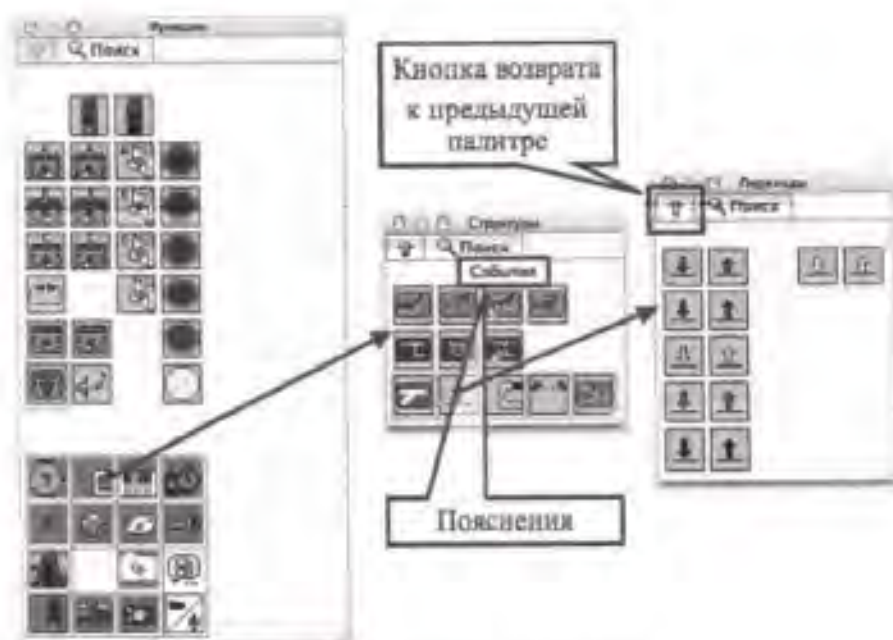
Одна пиктограмма может иметь до 5 мест соединения проводов, все они отображаются, когда курсор "моток провода" оказывается на пиктограмме. Провода определенного цвета соединяются с пиктограммами, имеющими обод такого же цвета.

В начале работы курсор мышки выглядит как указательный палец , но для перетаскивания и выделения пиктограмм необходимо, чтобы курсор выглядел как стрелочка (или крестик) . Для изменения вида курсора необходимо в меню "окно" выбрать пункт "показать палитру инструментов", в появившемся окне можно менять вид курсора. Выбираем курсор "стрелочка" .

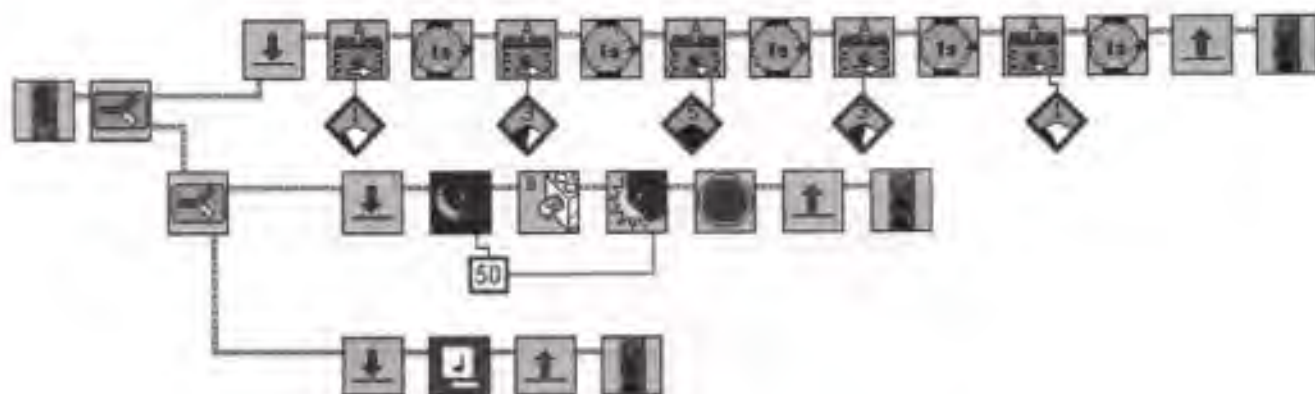


Еще один простой способ менять вид курсора - это использовать клавишу "пробел" клавиатуры, при ее нажатии вид курсора меняется на два наиболее часто используемых "стрелочка  " и "моток провода" ".

Некоторые кнопки в окне функций позволяют перейти к использованию новых пиктограмм, при этом окно меняется. Для возврата к предыдущей палитре используется кнопка "стрелка вверх".



Вариант программы движения карусели



Как альтернатива, на этом занятии можно попробовать собрать модель "Карусель" (см. Приложение), она очень хороша для знакомства с различными видами передач. Но, увы, у нас пока нет достаточного количества конструкторов, чтобы выполнить эту модель полностью. Да и, возможно, трудновато для младших школьников сразу столько нового материала.

Занятие 26-29

Тема: Проект Шлагбаум. Знакомство с червячной передачей. Сбор и программирование.

Для преподавателя:

В Приложении дана также схема сборки светофора, а также автоматическое открытие шлагбаума по датчику света. Это можно использовать дополнительно, для тех, кто справится с основным заданием.

Для детей:

Одной из самых медленных передач является червячная передача. (Своим названием она обязана основному её элементу, "червяку".)

Собираем по предложенному образцу (см. Приложение), если Вы всё сделано правильно, то дети сами в этом убеждаются. Особое внимание обращаем на то, насколько надёжен зацеп между "червяком" и шестернёй.

Еще важное замечание: шестерня превосходно вращается "червяком" и наоборот, вращая шестерню, невозможно повернуть ось, на которую насажен "червяк"? Это можно использовать в самых разных ситуациях, и нам это свойство очень пригодится. Запомним это!

Сегодня мы будем собирать по готовой схеме, и в конце занятия у нас получится модель шлагбаума. (См. Приложение)

Мозговой штурм:

- Как выглядит шлагбаум?
- Для чего нужен шлагбаум?
- Где **обязательно** устанавливается шлагбаум?
- Как и кем управляется шлагбаум?

Программирование сегодня также будет использоваться.

Вариант программы:



Тема: Модель «шлагбаум». Использование зубчатой передачи для уменьшения скорости модели.

Оборудование: Конструктор Лего «Индустрия развлечений», компьютер учителя, подключенный к нему проектор.

Цель этого занятия:

- ✓ Построить модель реального объекта,
- ✓ Обсудить необходимость этого объекта для жизнедеятельности человека, его безопасности,
- ✓ Узнать какие бывают шлагбаумы и разные его определения со времен его появления,
- ✓ В процессе исследования модели понять, что в построенном объекте не все так, как в реальном мире—понять потребность изменения скорости подъема стрелы шлагбаума,
- ✓ Применит для построения новую деталь—зубчатое колесо—шестеренки,
- ✓ Изучить новый вид соединения деталей—шестеренок,
- ✓ Запрограммировать работу шлагбаума,
- ✓ Представить свою модель классу—описать её, показать в работе.

В начале урока традиционная Лего-разминка, сегодня из 7 деталей.

Следующий этап—мозговая атака—«Как выглядит шлагбаум?» Учащиеся приводят описание, рассказывают о том, что это такое, Где они встречали это устройство. В форме беседы учитель рассказывает, для чего нужен шлагбаум, если этого не рассказали дети, или обобщает их рассказы.

Экскурс в историю—обращение к словарям со слайда позволяет информировать детей откуда взялось это название, слово шлагбаум заимствованное—здесь можно провести интеграцию с предметом «русский язык», вспомнить какие еще заимствованные слова знают учащиеся. Не более 2 минут.

После теоретического вступления учащимся предлагается первая часть практической работы.

Построение шлагбаума по представлению учащихся, они достаточно быстро справляются с заданием.

Умея уже работать с программой RoboLab на уровне Invertor 1, учащиеся могут запрограммировать работу шлагбаума. В таком случае можно не представлять им на этом уроке уровень Invertor 3. Этот уровень потребуется для работы на следующем уроке, но я знакомлю учащихся с этим уровнем на 4 уроке, так как есть время познакомиться с новым меню.

Слайд 13 можно оставить на экране на протяжении всего времени работы учеников за компьютерами.

Во время испытаний включите слайд 14, у детей должен возникнуть вопрос: «Как замедлить движение стрелы шлагбаума?», не подсказывайте им этот вопрос, опытным путем запуская на выполнение программу, устанавливая маленькое время подъема стрелы, они сами придут к этому.

Вот здесь, появится необходимость изучения новой темы: «Виды передач. Зубчатая передача».

Тема изучается экспериментальным путем. Нужно собрать соединение как на рисунке. Предварительно соберите его, чтобы показать детям, как закрепить оси сзади. Лучше, чтобы спереди оси тоже выступали.

Эксперимент, о котором идет речь на слайде—попросить одного ребенка держать ось с маленькой шестеренкой, а другого вращать ось, на которую насажена большая шестеренка—лишних деталей—типа рычагов быть не должно.

Затем пусть ребята поменяются—крутят ось с маленькой шестеренкой, а пытаются удержать ось, на которую насажена большая шестеренка.

Вопрос смогли ли они вращать ось, на которую насажена большая шестеренка, при удерживании оси с маленькой шестеренкой. (Нет или с трудом), в случае же когда ребенок удерживает ось с большой шестеренкой, а вращает другой маленькую сопротивление бесполезно—маленькую вращать легче.

Здесь необходимо объяснить детям, почему так происходит. Ввести понятие передаточное число, предложить посчитать зубчики на шестеренках—записать результаты в тетради. Произвести расчет передаточного числа.

Практическая работа—часть 2. Построение шлагбаума с использованием зубчатой передачи. По гиперссылке со слайда 18 можно в увеличенном виде показать вариант крепления деталей. Лучше будет, если вы соберете эту модель и покажете детям.

Еще лучше, если на данном уроке вы полностью соберете шлагбаум, соответственно без зубчатой передачи, и с зубчатой передачей. В первой части занятия покажите одну модель, а во второй—обе. Сделайте сравнительный анализ работы моделей.

Помогите детям, кто не справится с данным соединением.

Программа в микропроцессорах уже есть, можно построив модель испытать её. Уменьшается скорость, уменьшается расстояние, проходимое стрелой шлагбаума. Пусть ребята самостоятельно найдут параметр в программе, который нужно изменить, чтобы подъем стрелы шлагбаума был полноценный.



Пример построения шлагбаума с использованием червячной передачи. Если в вашем распоряжении есть конструктор 9723, или просто детали червячной передачи, полезно продемонстрировать детям такую модель. В такой интерпретации стрела шлагбаума поднимается еще медленнее.

Проверка уровня усвоения: По вашему усмотрению вы можете провести на уроке проверочную работу по материалам из папки «Дидактика» - задание к уроку 3.

Занятие 30-34

Тема: Датчики – органы чувств Робота. Шлагбаум-автомат. Использование датчика касания.

Оборудование: Конструктор Лего «Индустрия развлечений», компьютер учителя, подключенный к нему проектор.

Цель этого занятия:

- ✓ Построить модель реального управляемого объекта,
- ✓ Научиться работать с датчиком касания—правильно его крепить,
- ✓ Научиться включать в программу датчик касания, использовать меню «Модификаторы».

Лего-разминка из 8 деталей. Можно уже сюда включить датчик касания, как простой кирпич Лего. Но детей это уже заинтересует.

После разминки, изучим этот «кирпичик», нужно найти на нём кнопку сработки, кнопки подключения провода. Подключить провод. Учитель проверяет, все ли подключили правильно.

В программировании важно показать детям, что для кирпичика «датчик касания» пиктограмм существует две:

- ✓ «жди нажато»,
- ✓ «жди отпущено».

Кроме того, здесь впервые вводится необходимость подключения модификатора—

указывающего номер порта, к которому подключен датчик касания. (На уровне Invertor 1 нет такой необходимости, но датчик должен быть подключен к порту 1)

Ввиду 9-10 летнего возраста детей, я называю эту пиктограмму как «пиктограмма с ножкой» или «пиктограмма в одном ботинке». Название прижилось, и мы с детьми понимаем друг друга. В дальнейшем кто-то скажет: «Ты ботинок забыл одеть, поэтому программа не выполняется.» И одноклассник его поймет.

Для успешного составления программы покажите детям новые вложенные палитры: «Жди», «Прыжок-переход», «Модификатор».

Для чего нужны прыжки?, программируем повторяющиеся действия. Необходимо выделить «шаг цикла», научить видеть циклы вокруг себя, вывести детей на диалог.

Практика на уроке: Построение автоматического шлагбаума, вернее полуавтоматического, ведь нажатие на датчик касания будет осуществляться человеком. В полном смысле автоматический шлагбаум строиться с использованием датчика освящения, но с ним в 4 классе мы не будем работать.

Достаточно распространенной на этапе конструирования ошибкой является неправильное подключение датчика касания к проводу. Хотя в начале урока речь об этом шла. Проверьте, не случилось ли это с одной из ваших пар.

На этом уроке конструирование почти полностью повторяет конструирование прошлого урока, поэтому дайте возможность и время детям заняться программой. Пусть это будет главной заслугой их на уроке—создание рабочей программы.











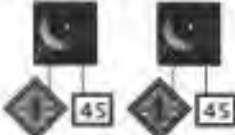


На этом уроке уместно поговорить о возможных ошибках в программе, и как посмотреть, какие ошибки, как считает сама программа RoboLab, присутствуют в алгоритме, составленном ребенком. Расскажите им о «сломанной серой стрелке», и как она подсказывает, где забыли соединить провода, а где подключили ни к тому полюсу, и на многие другие неточности соединений. Но эта «сломанная серая стрелка» не подскажет о потере «ботинка» - пиктограммы с модификатором—указателем порта подключения датчика касания. Здесь нужно внимание ученика.



На этом уроке нужно провести сопоставительный анализ шлагбаума с прошлого урока и нового. Вспомнить, что нового узнали за урок. Подвести итоги.

Сфотографировать модели.


































Завершение урока — запись домашнего задания, разбор моделей, сдача конструкторов дежурным.









Памятка


Пиктограммы	Пояснение	Где найти
	начало программы	Основное окно палитры инструментов
	красная метка (аналогична команде "label" других языков программирования)	 Структуры -> Переходы
	Включаем двигатель, подключенный к порту А и отключаем порт С. (поворот направо)	Основное окно палитры инструментов
	Ждать темнее чем, т.е. пока модель не наедет на черное	
	Модификатор, порт №1, обозначает к какому порту подключен тот или иной датчик	
	Модификатор, числовая константа (подсоединяется к пиктограммам датчиков снизу), числовое значение необходимо вводить сразу, либо использовать курсор \bar{A} \bar{A} .	 Модификаторы
	Пример: программа ждет когда на датчике света подключенном к порту №1 значение освещенности не станет меньше 45	
	Включаем двигатель, подключенный к порту С и отключаем порт А. (поворот налево)	Основное окно палитры инструментов
	Пример: программа ждет когда на датчике света подключенном к порту № 1 значение освещенности не станет больше 45	


	Красный прыжок (аналогичен команде "goto" других языков программирования)	^{->} Структуры -> Переходы
	Конец программы	


Перечень деталей


балка 1x2		4x	минишкив/блок		8x
балка 1x2		4x	универсальная втулка		20x
балка 1x4		4x	короткий штифт с кнопкой		4x
балка 1x6		4x	укороченный штифт 1,5 мм		8x
балка 1x12		2x	штифт гладкий		12x
балка 1x16 голубая		2x	штифт-полуось		8x
балка 1x16 черная		2x	черный штифт с выступами		12x
пластина 1x2		8x	фиксатор		2x
пластина 1x8		2x	захват с одним промежуточным отверстием		2x
опора скользящая черная 2x2		2x	захват		4x
пластина 2x4 с отверстиями		4x	втулка-удлиннитель оси		2x
пластина 2x6 с отверстиями		4x	кирпич 2x2 желтый		6x
пластина 2x8 с отверстиями		4x	кирпич 2x4 красный		6x
пластина 2x10 с отверстиями		2x	угловая балка со скруглением		2x
пластина 6x14		2x	8-зубое зубчатое колесо		3x
пластина угловая 2x2		2x	24-зубое зубчатое колесо		2x
			40-зубое зубчатое колесо		2x


- ось 2-кнопочная 4x 
- ось 3-кнопочная 2x 
- ось 4-кнопочная 2x 
- ось 5-кнопочная 2x 
- ось 6-кнопочная 2x 
- ось 8-кнопочная 2x 
- ось 10-кнопочная 2x 
- ось 12-кнопочная 2x 
- полуось 2x 

ремень синий 26 мм 1x 

полусфера с зеркальной поверхностью 2x 


черный парик 1x 


коричневый парик 1x 


фигурка человека 1x 

винтовая трубка 2x 


провод 12,8 см с соединительными пластинами 2x2 3x 


провод 52 см с соединительными пластинами 2x2 3x 


колесный диск 4x 

средний шкив 2x 


колесный диск большой 2x 

гладкая шина 2x 


малая шина с протектором 2x 


средняя шина с протектором 2x 

шина 2x 

лампа 1x2 1x 

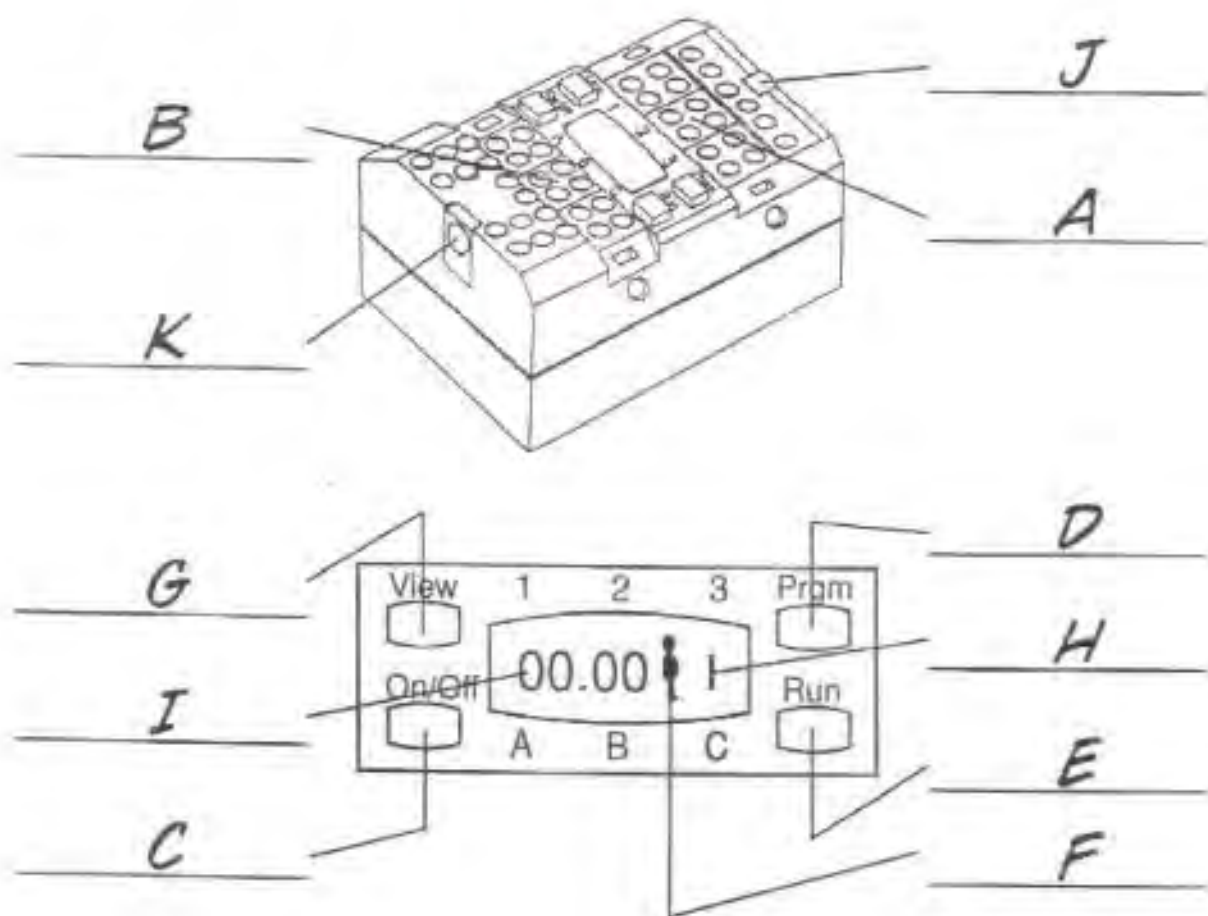
Датчик Освещенности 1x 

Датчик Касания 2x 

большой мотор 9 В 2x 

для соединения с USB-портом к набору 9786 1x 

Устройство RCX



- | | |
|---|---|
| <p>A. Порты ввода — к ним подключают датчики касания и освещенности</p> <p>B. Порты вывода — к ним подключают моторы и лампочки</p> <p>C. Кнопка включения/выключения блока RCX</p> <p>D. Кнопка выбора программного отсека, в который будет загружена программа</p> <p>E. Кнопка начала/остановки выполнения программы</p> | <p>F. Индикатор выполнения программы (показывает, выполняется ли программа в текущий момент)</p> <p>G. Кнопка просмотра текущего значения показаний датчика, подключенного к порту</p> <p>H. Индикатор номера текущего программного отсека</p> <p>I. Индикатор загрузки программного обеспечения</p> <p>J. Приемник/передатчик инфракрасных сигналов</p> <p>K. Гнездо сетевого адаптера</p> |
|---|---|