

А. К. Голубев

НАБОР ДЛЯ ПРОГРАММИРОВАНИЯ LITTLEBITS

МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ



МОСКВА
2021

УДК 372.8:004
ББК 32.816
М36

Голубев А. К.

М36 Набор для программирования littleBits: методические рекомендации /
А. К. Голубев. — М. : Издательство «Экзамен», 2021. — 64 с.
ISBN 978-5-377-16847-8

Базовый комплект модульной электроники «Набор для программирования» представляет собой набор электронных и структурных компонентов, предназначенных для получения базовых знаний в отрасли электроники и программирования, путем создания различных устройств с электрическими схемами, с помощью комбинирования элементов (модулей) электроники, входящих в комплект, и последующего их программирования.

Комплект предназначен для занятий на уроках робототехники в основной и средней школе. Он реализует требования образовательного стандарта к реализации системно-деятельностного подхода в обучении, развития мотивации к учению, интеллектуальной и творческой деятельности обучающихся, а также обеспечивает возможности групповой деятельности обучающихся, поддержку коммуникации между участниками образовательных отношений.

УДК 372.8:004
ББК 32.816

Подписано в печать с диапозитивов 09.02.2021.
Формат 60х90/8. Гарнитура «Calibri». Бумага офсетная.
Усл. печ. л. 8. Тираж 250 экз. Заказ №

ISBN 978-5-377-16847-8

© Голубев А. К., 2021
© Издательство «**ЭКЗАМЕН**», 2021
© «**ЭКЗАМЕН-ТЕХНОЛАБ**», 2021

СОДЕРЖАНИЕ

1. Базовый образовательный комплект модульной электроники «Набор для программирования» и комплекты расширений к нему	стр. 4
2. Знакомство	стр. 14
3. Ввод-вывод	стр. 20
4. Циклы	стр. 26
5. Логика	стр. 34
6. Переменные	стр. 40
7. Функции	стр. 48
8. Устранение неполадок	стр. 54
9. Решение проблем во время эксплуатации компонентов littleBits	стр. 58
10. Спросите трех раньше меня	стр. 62

1. БАЗОВЫЙ КОМПЛЕКТ МОДУЛЬНОЙ ЭЛЕКТРОНИКИ «НАБОР ДЛЯ ПРОГРАММИРОВАНИЯ» И КОМПЛЕКТЫ РАСШИРЕНИЙ К НЕМУ

ОБРАЗОВАТЕЛЬНЫЙ КОМПЛЕКТ МОДУЛЬНОЙ ЭЛЕКТРОНИКИ «НАБОР ДЛЯ ПРОГРАММИРОВАНИЯ»



Базовый образовательный комплект модульной электроники «Набор для программирования» представляет собой набор электронных и структурных компонентов, предназначенных для получения базовых знаний в отрасли электроники и программирования путем создания различных устройств с электрическими схемами, с помощью комбинирования элементов (модулей) электроники, входящих в комплект, и последующего их программирования. Программирование осуществляется в блочно-модульной графической среде Blockly, с возможностью представления кода в текстовой среде JavaScript.

Комплект позволяет создавать и программировать электрические схемы без их сборки на макетной плате и без пайки.

Комплект предусматривает магнитное соединение электронных модулей между собой, а также возможность крепления модулей на монтажной панели. Магнитное соединение электронных модулей между собой исключает возможность размещения модулей в неправильном порядке.

Электронные модули имеют обозначения, указывающие направление движения тока через них.

Электронные модули сгруппированы в соответствии с их функционалом. Модули одной группы имеют одинаковый цвет.

Комплект включает в себя 16 электронных модулей:

Наименование (англ.)	Наименование (рус.)	Кол-во в наборе	Фото	Описание
Bargraph	Гистограмма	1		Показывает, какое количество сигнала получает модуль с дисплеем из пяти светодиодов разных цветов
Branch	Разветвитель	1		Обеспечивает подключение одного модуля к 3 другим в разных направлениях от разветвителя
Button	Кнопки включения\выключения	2		Выполняют функцию включения\выключения
codeBit™	Программируемый модуль codeBit™	1		Позволяет программировать созданные схемы, служить для создания уникальных звуков, движений и анимации, обеспечивать прием\передачу данных по беспроводной сети с компьютера или другого устройства, используемого для написания программы
Dimmer	Диммер	1		Позволяет регулировать силу сигнала, передаваемого по цепи

Наименование (англ.)	Наименование (рус.)	Кол-во в наборе	Фото	Описание
Slide dimmer	Ползунковый диммер	1		Позволяет регулировать силу сигнала, передаваемого по цепи, при помощи соответствующего ползунка
Speaker	Динамик	1		Позволяет воспроизводить сигналы, поступающие от программируемого модуля или плеера
Wire	Проводные модули	3		Позволяют размещать соединяемые модули на расстоянии друг от друга, изменять углы соединения, а также вращать соединенные модули
LED matrix	Светодиодная матрица	1		Представляет из себя цветной дисплей, которым можно управлять, используя написанную программу либо другие модули, а также имеет 2 режима воспроизведения сигнала
USB power	Модуль питания USB	1		Позволяет обеспечивать питанием собранную электрическую схему путем подключения его через USB-кабель к источнику питания

Наименование (англ.)	Наименование (рус.)	Кол-во в наборе	Фото	Описание
Pressure sensor	Датчик силы нажатия	1		Позволяет регулировать силу передаваемого сигнала в зависимости от силы нажатия на его подушку
Servo	Servo	1		Представляет собой управляемый двигатель, который может вращаться вперед и назад или поворачиваться в определенное положение, имеет вал поперечной оси, который совместим с любым аксессуаром данного комплекта, имеющим соответствующее отверстие
Sound trigger	Звуковой триггер	1		Позволяет определять уровень шума в помещении и передавать сигнал включения по цепи при достижении определенного уровня шума

Также в комплекте представлены дополнительные аксессуары, среди которых:

- USB кабели – 2 шт.;
- аккумуляторная батарея – 1 шт.;
- беспроводной передатчик данных – 1 шт.;
- полоски для крепления модулей – 2 шт.;
- дополнительные аксессуары для крепления модулей – 12 шт.;
- устройства подачи питания – 3 шт.;
- механический захват – 1 шт.;
- монтажная панель (175 мм × 200 мм) – 1 шт.

РЕСУРСНЫЙ НАБОР МОДУЛЬНОЙ ЭЛЕКТРОНИКИ «ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ»



Ресурсный набор модульной электроники «Информационные технологии» является расширением к комплекту «Набор для программирования». Он представляет собой набор электронных и структурных компонентов, предназначенных для расширения полученных навыков программирования путем программирования устройств с электрическими схемами, созданных с помощью комбинирования элементов (модулей) электроники. Программирование осуществляется в блочно-модульной графической среде Blockly, с возможностью представления кода в текстовой среде JavaScript.

Комплект включает в себя 9 электронных модулей:

Наименование (англ.)	Наименование (рус.)	Кол-во в наборе	Фото	Описание
Accelerometer	Акселерометр	1		Определяет величину его ускорения в созданной цепи
Fork	Вилка	1		Позволяет одновременно подключить один модуль к трем другим

Наименование (англ.)	Наименование (рус.)	Кол-во в наборе	Фото	Описание
Light sensor	Датчик освещенности	1		Обеспечивает управление созданной цепи при помощи определения изменения освещенности вокруг датчика
Long LED	Длинные светодиоды	2		Представляют собой светодиоды, подключенные к плате с помощью кабеля
Number	Числовой модуль	1		Отображает информацию в числовом виде, полученную от модулей, расположенных до него в цепи
Proximity sensor	Датчик приближения	1		Определяет расстояние до объекта перед датчиком
Random	Модуль случайности	1		Выдает случайные величины напряжения или уровня шума
Wire	Проводной модуль	1		Позволяет размещать соединяемые модули на расстоянии друг от друга, изменять углы соединения, а также позволяет вращать соединенные модули

Также в комплекте представлены дополнительные аксессуары, среди которых:

- монтажная панель (175 мм × 200 мм) – 1 шт.;
- дополнительные аксессуары для крепления модулей – 12 шт.;
- полоски для крепления модулей – 2 шт.

РЕСУРСНЫЙ НАБОР МОДУЛЬНОЙ ЭЛЕКТРОНИКИ «ТЕХНОЛОГИЯ»



Ресурсный набор модульной электроники «Технология» является расширением к комплекту «Набор для программирования». Он представляет собой набор электронных и структурных компонентов, предназначенных для расширения полученных навыков в области технологии путем программирования устройств с электрическими схемами, созданных с помощью комбинирования элементов (модулей) электроники.

Программирование осуществляется в блочно-модульной графической среде Blockly, с возможностью представления кода в текстовой среде JavaScript.

Комплект включает в себя 13 электронных модулей:

Наименование (англ)	Наименование (рус)	Кол-во в наборе	Фото	Описание
Power	Модуль питания	1		Использует щелочную батарею для обеспечения током цепи

Наименование (англ)	Наименование (рус)	Кол-во в наборе	Фото	Описание
Bargraph	Гистограмма	1		Показывает, какое количество сигнала получает модуль с дисплеем из 5 светодиодов разных цветов
Buzzer	Зуммер	1		Выполняет функцию сигнального устройства
DC Motor	Двигатели постоянного тока	2		Обеспечивают вращение поперечного вала
Light sensor	Датчик освещенности	1		Обеспечивает управление созданной цепи при помощи определения изменения освещенности вокруг датчика
Long LED	Длинный светодиод	1		Представляет собой светодиод, подключенный к плате с помощью кабеля
Number	Числовой модуль	1		Отображает информацию в числовом виде, полученную от модулей, расположенных до него в цепи
Proximity sensor	Датчик приближения	1		Определяет расстояние до объекта перед датчиком

Наименование (англ)	Наименование (рус)	Кол-во в наборе	Фото	Описание
Split	Модуль разделения	1		Обеспечивает подключение одного модуля к 2 другим с использованием проводов, что позволяет размещать соединяемые модули на расстоянии друг от друга и изменять углы соединения
Wire	Проводные модули	3		Позволяют размещать соединяемые модули на расстоянии друг от друга, изменять углы соединения, а также вращать соединенные модули

Также в комплекте представлены дополнительные аксессуары, среди которых:

- батарея питания 9В – 1 шт.;
- кабель для батареи питания – 1 шт.;
- зажим для батареи – 1 шт.;
- механический захват – 1 шт.;
- колеса – 2 шт.;
- крепление к двигателю – 1 шт.;
- ролик – 1 шт.;
- ось – 1 шт.;
- монтажная панель (175 мм × 200 мм) – 1 шт.

2. ЗНАКОМСТВО





ОБЗОР УРОКА

Класс обучения: 3-5 классы

Продолжительность занятия: 60 минут

Изучаем основы базового набора (Образовательный комплект модульной электроники «Набор для программирования»).



ОСНОВНЫЕ МОМЕНТЫ

СТУПЕНЬ ОБРАЗОВАНИЯ	УЧЕБНЫЕ ДИСЦИПЛИНЫ	СЛОЖНОСТЬ	ПРОДОЛЖИТЕЛЬ- НОСТЬ
Начальная, средняя	Информатика, проектная деятельность	Ознакомительная	60 минут

ТРЕБУЕМЫЕ НАВЫКИ

Набирать текст на
клавиатуре,
использовать
манипулятор типа
«мышь»



НЕОБХОДИМЫЕ МАТЕРИАЛЫ

МОДУЛИ

Набор для программирования

ДРУГОЕ

Компьютер с установленным
приложением Code Kit App, USB-порт

ХОД ЗАНЯТИЯ



ШАГ 1: НАСТРОЙКА

Продолжительность: 10 минут

Загрузите и установите приложение Code Kit App на компьютеры учеников. Ученики смогут разделиться на группы по 2-3 человека и использовать 1 набор на группу.



ШАГ 2: ВВЕДЕНИЕ

Продолжительность: 10 минут

Программируем робота-учителя.

Представьте, что ваш учитель сегодня превратился в компьютер и вам, ученикам, предстоит помочь ему выполнить задание с помощью четких указаний (например, выключи свет или отпусти всех домой пораньше).

Напишите подробную инструкцию, как робот-учитель должен будет выполнить поставленную вами задачу.

Поделитесь программой с классом и запрограммируйте робота-учителя на выполнение вашей задачи. Помните, что компьютер будет делать именно то, что вы ему скажете.



Обсудите:

- Есть ли разница в манере объяснения задачи компьютеру и человеку?
- Какого рода инструкции наиболее полезны?

Компьютер не может догадаться или домыслить, что вы имели в виду. Когда мы даем команду компьютеру, мы должны делать это максимально четко,

иначе, получив команду «вперед», компьютер будет идти вперед, пока не обойдет земной шар или не упрется в стенку.



ШАГ 3: ТВОРЧЕСТВО

Продолжительность: 15 минут

Разбейтесь на группы по 2-3 ученика и возьмите по одному набору Code Kit и компьютеру на группу.

Познаем основы

Самостоятельно начните изучать состав наборов и базовые функции модулей. Назовите особенности LittleBits.

Делимся знаниями

Обсудите со всем классом, что нового вы узнали о LittleBits.

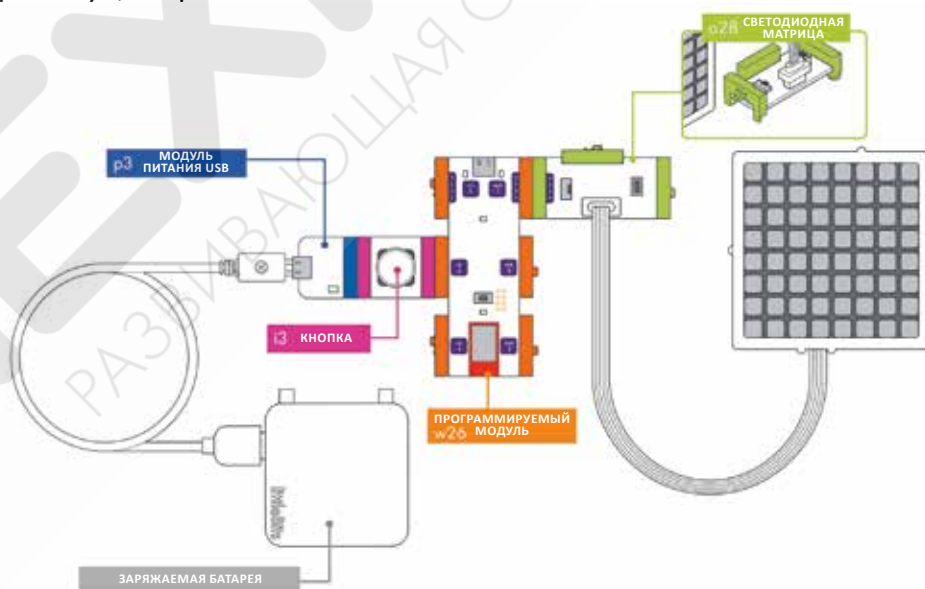
Смотрим видео (дополнительно)

Видеоролик про базовые возможности набора LittleBits.

Знакомство с модулем codeBit™

Настало время познакомиться с программированием LittleBits. Воспользуйтесь предложенной ниже схемой и соберите собственную цепь. Затем запрограммируйте ее так, чтобы при нажатии на кнопку светодиодная матрица высвечивала улыбку. При повторном нажатии будет светиться слово «Hello, World!». Помните, что светодиодная матрица небольшая и одновременно может вместить только часть слова.

«Здравствуй, Мир!»





ШАГ 4: УЧИМСЯ ИГРАЯ

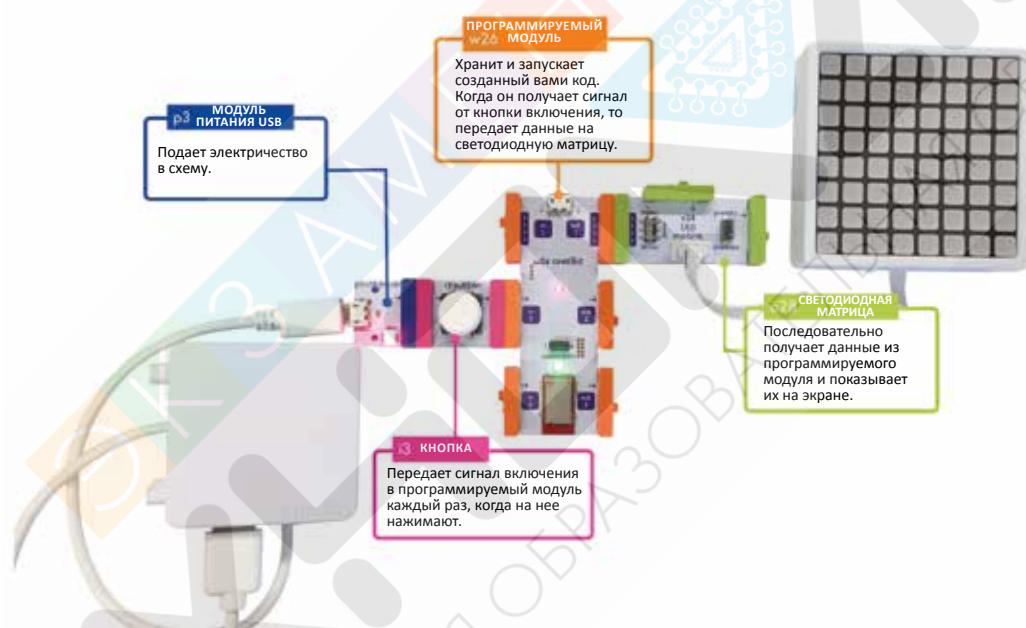
Продолжительность: 10 минут

Программируем

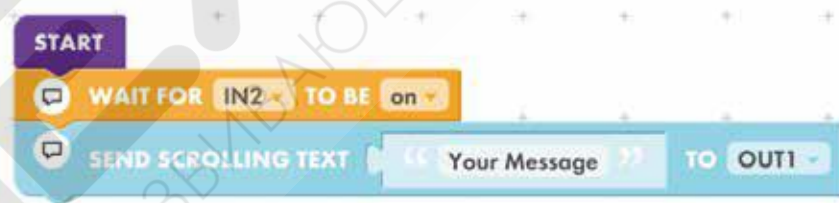
Попросите учеников запустить приложение Code Kit App и найти в нем «Hello, World!» («Здравствуй, Мир!»). Вам необходимо будет проследовать инструкциям и запрограммировать свою схему, загрузив в нее код из приложения. Это упражнение позволит ученикам понять принцип программирования схем с помощью приложения.

Принцип работы:

- Ниже представлены принципы работы каждого типа модулей:



- Программный код:



В соответствии с кодом, представленным выше, программа будет выполнять определенное действие только после нажатия кнопки.

После нажатия кнопки модуль программирования получит сигнал запустить следующий программный блок и направит на светодиодную матрицу текст, который вы укажете.

Это однократная программа, если вы захотите заново отправить текст на матрицу, вам необходимо будет начать программу заново.



ШАГ 5: ДОПОЛНЯЕМ

Продолжительность: 5 минут

Не стесняйтесь экспериментировать и пробовать новое. Ниже вы найдете ряд дополнительных заданий:

- Какие еще изображения вы можете направить на светодиодную матрицу? Сможете нарисовать своего любимого персонажа или животное?
- С помощью каких других модулей набора мы сможем вывести изображение или текст?
- Вам удалось запустить изображение на светодиодной матрице с помощью нажатия кнопки. Сможете ли вы создать более длительную анимацию из нескольких изображений? (Скорее всего для решения этой задачи вам понадобится блок [WAIT 1 SECOND])



ШАГ 6: ДЕЛИМСЯ РЕЗУЛЬТАТАМИ

Продолжительность: 5 минут

Расскажите всему классу о том, какое изображение вы вывели на светодиодную матрицу. С какими сложностями вам пришлось столкнуться и как вы их решили? Обсудите всем классом принцип работы схемы «Здравствуй, Мир!».



ШАГ 7: ЗАВЕРШЕНИЕ

Продолжительность: 5 минут

Чистота и порядок на рабочем месте – залог успеха. Разберем все собранные схемы и уберем все детали на свои места.

3. ВВОД-ВЫВОД





ОБЗОР УРОКА

Класс обучения: 3-5 классы

Продолжительность занятия: 85 минут

Ученики выполняют ряд упражнений для изучения основ ввода-вывода в приложении Code Kit App и смогут узнать, как управлять изображениями, звуком и движениями.



ОСНОВНЫЕ МОМЕНТЫ

СТУПЕНЬ ОБРАЗОВАНИЯ

Начальная,
средняя

УЧЕБНЫЕ ДИСЦИПЛИНЫ

Информатика,
проектная
деятельность

СЛОЖНОСТЬ

Ознакомительная

ПРОДОЛЖИТЕЛЬ- НОСТЬ

85 минут

ТРЕБУЕМЫЕ НАВЫКИ

Понимание рабо-
ты модулей набора
LittleBits,
умение програм-
мировать с помо-
щью блоков,
умение загружать
программу в мо-
дуль codeBit™



НЕОБХОДИМЫЕ МАТЕРИАЛЫ

МОДУЛИ

Набор для программирования

ДРУГОЕ

Компьютер с установленным
приложением Code Kit App, USB-порт

ХОД ЗАНЯТИЯ



ШАГ 1: НАСТРОЙКА

Продолжительность: 30 минут

Разделитесь на группы по 2-3 человека.

Каждой группе понадобится один набор для программирования, компьютер с установленным приложением Code Kit App и рабочая тетрадь на каждого ученика.



ШАГ 2: ВВЕДЕНИЕ

Продолжительность: 10 минут

Что такое ввод? Что такое вывод?

Воспользуйтесь приложением Code Kit App и посмотрите ролик про вводы-выводы.

Обсудите:

- Попросите каждую группу выбрать любой элемент вывода из набора и рассказать о нем.
- Попросите каждую группу выбрать любой элемент ввода из набора и рассказать о нем.

Каким образом модуль codeBit™ работает с элементами ввода-вывода? Почему мы можем называть модуль codeBit™ мозгом всей схемы?



ШАГ 3: ТВОРЧЕСТВО

Продолжительность: 30 минут

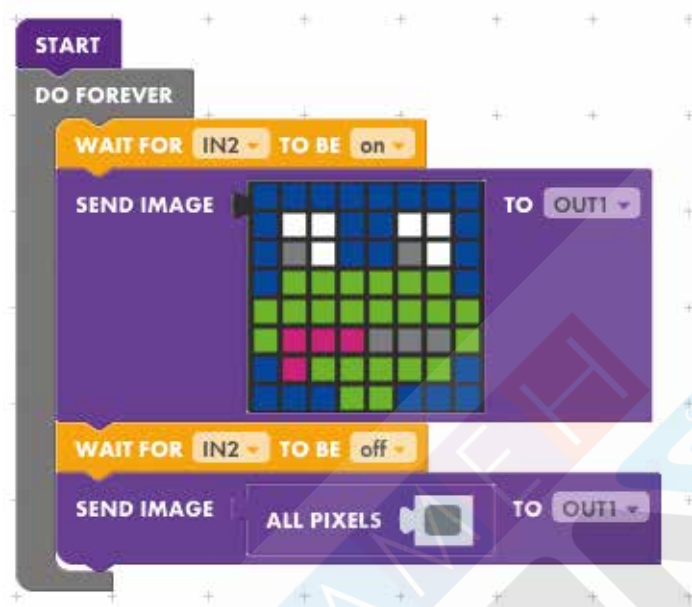
Вводный урок – давайте создадим игровые контроллеры!

Воспользуйтесь приложением Code Kit App и пройдите ознакомительные уроки в следующем порядке:

- 1.0 Ваш первый игровой контроллер
- 1.1 Звуковые эффекты из игр
- 1.2 Смешиваем звуки [дополнительно]

1.0 Ваш первый игровой контроллер

Результат



Принцип работы

Блок цикла [DO FOREVER] заставляет написанный внутри него код повторяться бесконечное количество раз.

Блок ожидания [WAIT FOR IN 2 TO BE ON] заставляет программу ожидать до тех пор, пока не поступит сигнал с блока ввода, который подключен к IN 2 модуля codeBit™.

В случае если значения codeBit™ – «ВКЛ», блок [SEND IMAGE TO OUT 1] отправляет изображение на LED-матрицу.

После отправки изображения блок ожидания [WAIT FOR IN 2 TO BE OFF] заставляет программу ожидать до тех пор, пока блок, подсоединенный к IN 2, перестанет посылать сигнал. Как только сигнал перестанет поступать, запустится блок [SEND IMAGE ALL PIXELS [] TO OUT 1] и выключит LED-матрицу.

1.1 Звуковые эффекты из игр

Результат



Принцип работы

Блок цикла [DO FOREVER] заставляет написанный внутри него код повторяться бесконечное количество раз.

Блок ожидания [WAIT FOR IN 1 TO BE > 50%] заставляет программу ждать до тех пор, пока значение сигнала, поступающего с датчика на IN 1, будет больше 50%.

Как только значение превысит 50%, блок [SEND TONE TO OUT 2 FOR 250 MS] направит звуковой сигнал определенной тональности с определенной продолжительностью на динамики.

1.2. Смешиваем звуки

Результат



Принцип работы

Блок цикла [DO FOREVER] заставляет написанный внутри него код повторяться бесконечное количество раз.

Блок [SEND TONE TO OUT 2 FOR 250 MS] направляет звуковой сигнал на динамик, присоединенный к OUT 2. Далее нам необходимо, чтобы модуль, присоединенный к IN 2, определил тональность звука, но проблема в том, что звуковой сигнал находится в частоте 0 – 100, а это очень низкие частоты. Для изменения частот мы воспользовались блоком [SIGNAL FROM IN 2 CONVERTED TO A TONE BETWEEN [50] Hz AND [1000] Hz] внутри блока [SEND TONE]. Таким образом мы меняем частоту поступающего сигнала. Частота сигнала измеряется в Hz (Гц).



ШАГ 4: ДЕЛИМСЯ РЕЗУЛЬТАТАМИ

Продолжительность: 10 минут

Что мы узнали, чему научились?

Попросите учеников выполнить в рабочей тетради задания по теме ввод-вывод.



ШАГ 5: ЗАВЕРШЕНИЕ

Продолжительность: 5 минут

Попросите учеников разобрать все собранные схемы и убрать элементы обратно в коробки.

4. ЦИКЛЫ





ОБЗОР УРОКА

Класс обучения: 3-5 классы

Продолжительность занятия: 60 минут

Ученики выполняют ряд упражнений для изучения основ программируемого цикла в приложении Code Kit App и научатся создавать анимацию.



ОСНОВНЫЕ МОМЕНТЫ

СТУПЕНЬ ОБРАЗОВАНИЯ	УЧЕБНЫЕ ДИСЦИПЛИНЫ	СЛОЖНОСТЬ	ПРОДОЛЖИТЕЛЬНОСТЬ
Начальная, средняя	Информатика, проектная деятельность	Ознакомительная	60 минут

ТРЕБУЕМЫЕ НАВЫКИ

Понимание работы модулей набора LittleBits, умение программировать с помощью блоков, умение загружать программу в модуль codeBit™, пройденный урок «Вводы-выводы»



НЕОБХОДИМЫЕ МАТЕРИАЛЫ

МОДУЛИ

Набор для программирования

ДРУГОЕ

Компьютер с установленным приложением Code Kit App, USB-порт

ХОД ЗАНЯТИЯ



ШАГ 1: НАСТРОЙКА

Продолжительность: 10 минут

Разделите учеников на группы по 2-3 человека.

Каждой группе понадобится один набор для программирования, компьютер с установленным приложением Code Kit App и рабочая тетрадь на каждого ученика.



ШАГ 2: ВВЕДЕНИЕ

Продолжительность: 10 минут

Что такое цикл?

Всем классом посмотрите вступительный ролик про циклы в приложении Code Kit App.

Сможете ли вы вспомнить что-либо из повседневной жизни, похожее на цикл?



ШАГ 3: ТВОРЧЕСТВО

Продолжительность: 30 минут

Делаем анимацию!

Воспользуйтесь приложением Code Kit App и пройдите ознакомительные уроки в следующем порядке:

2.0 Создайте анимацию для придуманного персонажа

2.1 Секрет

2.2 Спецэффекты [дополнительно]

2.0 Создайте анимацию для придуманного персонажа

Результат



Принцип работы

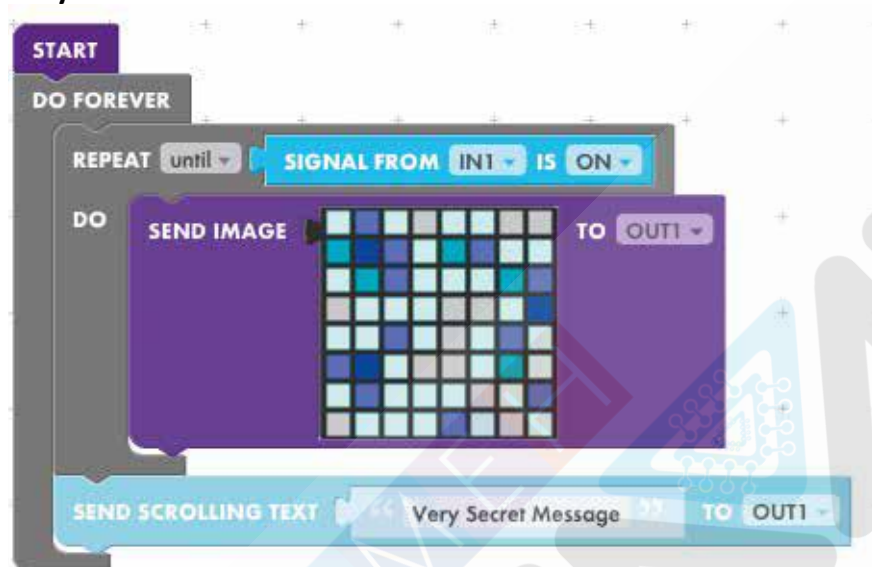
Блок цикла [DO FOREVER] заставляет написанный внутри него код повторяться бесконечное количество раз.

Блок [SEND IMAGE TO OUT 1] создает кадры анимации.

Блок ожидания [WAIT 1 SECONDS] устанавливает время между сменой кадров. Если мы не установим этот блок и нужное нам значение, кадры будут показаны слишком быстро и мы не увидим анимацию. После исполнения последнего блока [DO FOREVER] цикл запускается снова в порядке расположения блоков сверху вниз.

2.1 Секрет

Результат



Принцип работы

Блок цикла [DO FOREVER] заставляет написанный внутри него код повторяться бесконечное количество раз.

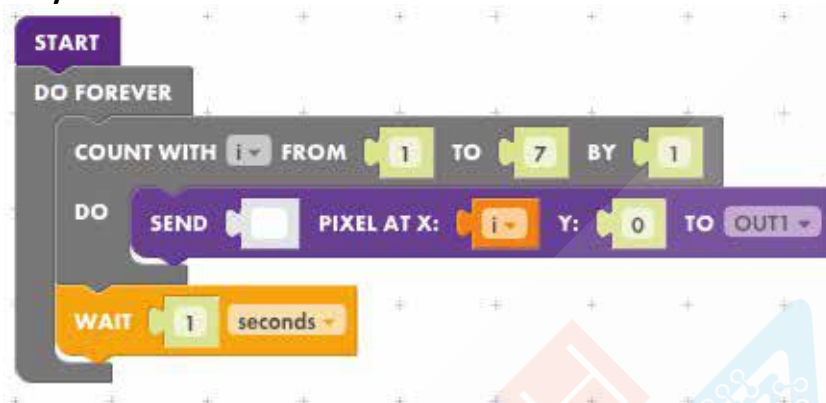
Блок цикла [REPEAT UNTIL, DO], который расположен внутри блока цикла [DO FOREVER], заставляет написанную внутри него часть программы повторяться до тех пор, пока не будут выполнены определенные условия. В нашем случае повтор будет происходить до тех пор, пока мы не нажмем на кнопку, присоединенную к IN 1 [SIGNAL FROM IN 1 IS ON].

Блок [SEND IMAGE] будет работать до тех пор, пока мы не нажмем на кнопку. Когда мы нажмем на кнопку, блок цикла [REPEAT UNTIL] прекратит повторять часть программы, и далее начнется выполнение блока [SEND SCROLLING TEXT]. Как раз этот блок и покажет наше секретное сообщение.

После отправки сообщения на LED-матрицу вся программа начнется заново в порядке расположения блоков сверху вниз.

2.2 Спецэффекты

Результат



Принцип работы

Блок цикла [DO FOREVER] заставляет написанный внутри него код повторяться бесконечное количество раз.

Блок цикла [COUNT WITH [i] FROM [1] TO [7] BY [1], DO] находится внутри блока цикла [DO FOREVER].

Этот блок цикла использует переменную [i], которая содержит часть кода (например – число). В нашем случае каждый раз, когда повторяется цикл, переменная [i] увеличивается на 1.

Мы выбрали интервал значений от 0 до 7, потому что LED-матрица содержит всего 8 пикселей и они пронумерованы от 0 до 7.

Блок [SEND [] PIXEL AT X: [i] Y:0 TO OUT1] отправляет изображение на один из пикселей LED-матрицы с координатами X, Y. Переменная [i] будет увеличиваться на 1 каждый раз, когда блок цикла [COUNT WITH] будет срабатывать, что, в свою очередь, будет увеличивать значение X координаты, передвигая изображение на один пиксель вправо по оси X. Как только переменная [i] достигнет значения 7, блок цикла [COUNT WITH] остановится и программа перейдет к выполнению следующего блока.

Блок ожидания [WAIT 1 seconds] добавляет паузу перед запуском блока цикла [DO FOREVER] и запуском блока цикла [COUNT WITH].



ШАГ 4: ДЕЛИМСЯ РЕЗУЛЬТАТАМИ

Продолжительность: 5 минут

Что нового вы узнали на этом уроке? С чем испытали трудности?

Попросите учеников открыть рабочую тетрадь и выполнить упражнения, связанные с циклами.

[Дополнительно] Обсудите трудности, с которыми ученики столкнулись в процессе выполнения задач этого урока.



ШАГ 5: ЗАВЕРШЕНИЕ

Продолжительность: 5 минут

Попросите учеников разобрать все собранные схемы и убрать элементы обратно в коробки.

5. ЛОГИКА





ОБЗОР УРОКА

Класс обучения: 3-10 классы

Продолжительность занятия: 60 минут

Ученики выполняют ряд упражнений для изучения основ логики в программировании в приложении Code Kit app и научатся использовать выражения [IF/DO] и [IF/DO/ELSE] для программирования.



ОСНОВНЫЕ МОМЕНТЫ

СТУПЕНЬ ОБРАЗОВАНИЯ

Начальная,
средняя

УЧЕБНЫЕ ДИСЦИПЛИНЫ

Информатика,
проектная
деятельность

СЛОЖНОСТЬ

Ознакомительная

ПРОДОЛЖИТЕЛЬ- НОСТЬ

60 минут

ТРЕБУЕМЫЕ НАВЫКИ

Понимание работы модулей набора LittleBits, умение программировать с помощью блоков, умение загружать программу в модуль codeBit™, пройденные уроки «Вводы-выводы» и «циклы»



НЕОБХОДИМЫЕ МАТЕРИАЛЫ

МОДУЛИ

Набор для программирования

ДРУГОЕ

Компьютер с установленным приложением Code Kit App, USB-порт

ХОД ЗАНЯТИЯ



ШАГ 1: НАСТРОЙКА

Продолжительность: 5 минут

Предварительная настройка до начала занятия.

Разделите учеников на группы по 2-3 человека. Каждой группе понадобится один набор для программирования, компьютер с установленным приложением Fusebox App и рабочая тетрадь для каждого ученика.



ШАГ 2: ВВЕДЕНИЕ

Продолжительность: 15 минут

Что такое логика?

Посмотрите ознакомительный ролик про логику в программировании в приложении Fusebox App.

Действия:

- Запишите выражение: IF _____ DO _____ ELSE _____. По очереди ученики подходят к доске и записывают недостающие слова. Например, IF (Сегодня пойдет дождь) DO (Взять зонтик) ELSE (Если сегодня не будет дождя, останься дома).
- Обсудите – как выбор действий в повседневной жизни похож на логику в программировании.



ШАГ 3: ТВОРЧЕСТВО

Продолжительность: 15 минут

Программируем!

Воспользуйтесь приложением Fusebox App и пройдите ознакомительные уроки в следующем порядке:

3.0 Делаем Эмодзи!

3.1 Волшебный меч

3.0 Делаем Эмодзи!

Результат



Принцип работы

Блок цикла [DO FOREVER] заставляет написанный внутри него код повторяться бесконечное количество раз.

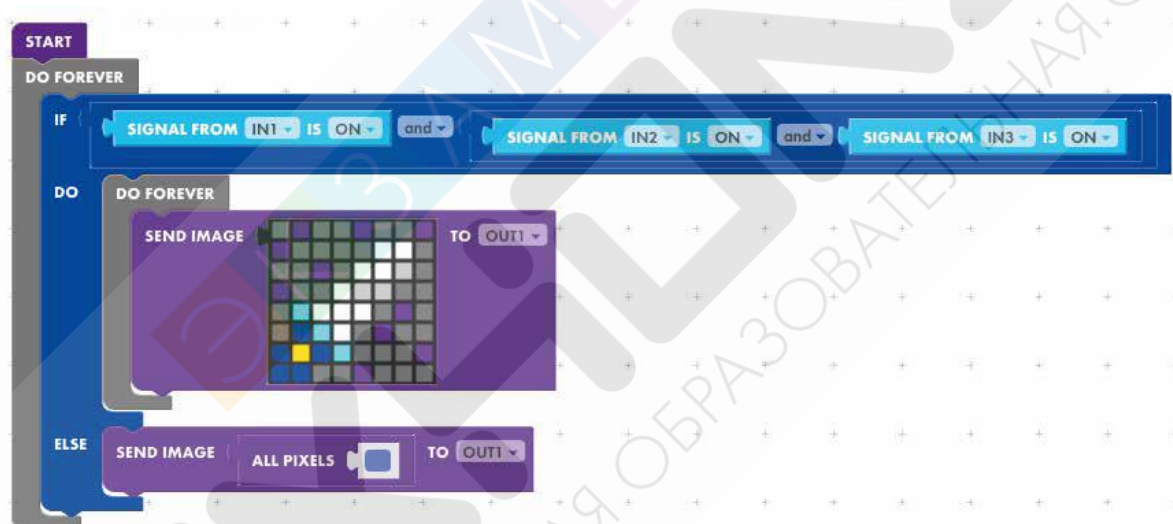
Внутри основного блока цикла расположены 4 похожих кода. Каждый из этих кодов находится внутри блока [IF/DO], который заставляет схему показывать

то или иное Эмодзи, в зависимости от положения ползунка.

Блок [SIGNAL FROM IN 2 CONVERTED TO NUMBER BETWEEN 1 AND 4] меняет поступающий сигнал с ползунка со значениями 0-100 на значения 1 – 4. Эти 4 значения соответствуют четырем типам Эмодзи, из которых программа выбирает. Блок [ROUND] округляет поступающий сигнал, и на выходе мы получаем целые числа, соответствующие изображениям Эмодзи. Блок [=] позволяет программе понять, какое именно из значений получилось (1, 2, 3, или 4).

Если все работает верно, то блок [SEND IMAGE (IMAGE) TO OUT 1] направляет изображение 1 на OUT 1, куда присоединена LED-матрица.

3.1 Волшебный меч



Принцип работы

Блок цикла [DO FOREVER] заставляет написанный внутри него код повторяться бесконечное количество раз.

Блок [IF/DO/ELSE] заставляет программу проверять, поступает ли (IF) сигнал со всех трех вводов (ON). При значении «истина» включается следующая команда блока (DO) и LED-матрица покажет изображение меча при помощи блоков [DO FOREVER [SEND IMAGE TO OUT 1]]. При значении «ложь» начнет работать иная команда блока (ELSE) и LED-матрица станет вся фиолетовой при помощи блока [SEND IMAGE [ALL PIXELS] TO OUT 1].

Обратите внимание на то, где расположена команда DO – внутри блока цик-

ла [DO FOREVER], это позволит нам вывести постоянное изображение на LED-матрицу. В противном случае изображение будет гореть, только пока мы нажимаем на кнопки.



ШАГ 4: ДЕЛИМСЯ РЕЗУЛЬТАТАМИ

Продолжительность: 5 минут

Что мы узнали, чему научились?

Попросите учеников выполнить задания по логике в рабочей тетради.



ШАГ 5: ЗАВЕРШЕНИЕ

Продолжительность: 5 минут

Попросите учеников разобрать все собранные схемы и убрать элементы обратно в коробки.

6. ПЕРЕМЕННЫЕ





ОБЗОР УРОКА

Класс обучения: 3-5 классы

Продолжительность занятия: 50 минут

Ученики выполняют ряд упражнений для изучения основ программируемого цикла в приложении Fusebox App и узнают, как пользоваться переменными «изображение», «время» и «координата».



ОСНОВНЫЕ МОМЕНТЫ

СТУПЕНЬ ОБРАЗОВАНИЯ

Начальная,
средняя

УЧЕБНЫЕ ДИСЦИПЛИНЫ

Информатика,
проектная
деятельность

СЛОЖНОСТЬ

Ознакомительная

ПРОДОЛЖИТЕЛЬ- НОСТЬ

50 минут

ТРЕБУЕМЫЕ НАВЫКИ

Понимание работы модулей набора LittleBits, умение программировать с помощью блоков, умение загружать программу в модуль codeBit™, пройденные уроки «Вводы-выводы», «Циклы» и «Логика»



НЕОБХОДИМЫЕ МАТЕРИАЛЫ

МОДУЛИ

Набор для программирования

ДРУГОЕ

Компьютер с установленным приложением Code Kit App, USB-порт

ХОД ЗАНЯТИЯ



ШАГ 1: НАСТРОЙКА

Продолжительность: 10 минут

Предварительная настройка до начала занятия.

Разделите учеников на группы по 2-3 человека. Каждой группе понадобится один набор для программирования, компьютер с установленным приложением Fusebox App и рабочая тетрадь для каждого ученика.

Каждому ученику понадобятся стикеры для записи.



ШАГ 2: ВВЕДЕНИЕ

Продолжительность: 10 минут

Посмотрите ознакомительный ролик про переменные в приложении Fusebox App.

Действия:

- Каждый ученик кладет свой рюкзак на стол и клеит на него стикер. На стикере пишется имя рюкзака - он и будет нашей переменной.
- Несколько учеников покажут, что лежит в их рюкзаках (учебник, ручка и т.д.).
- Обратите внимание, как рюкзак выступает в роли переменной, как он хранит разнообразные предметы, которые мы можем достать в любой момент. Приведите примеры других переменных из повседневной жизни (например ведро, шкаф, коробка).



ШАГ 3: ТВОРЧЕСТВО

Продолжительность: 25 минут

Программируем!

Воспользуйтесь приложением Fusebox App и пройдите ознакомительные уроки в следующем порядке:

4.0 День рождения

4.1 Спрайт «Прыгающий монстр»

4.2 Гонка [дополнительно]

4.0 День рождения

Результат



Принцип работы

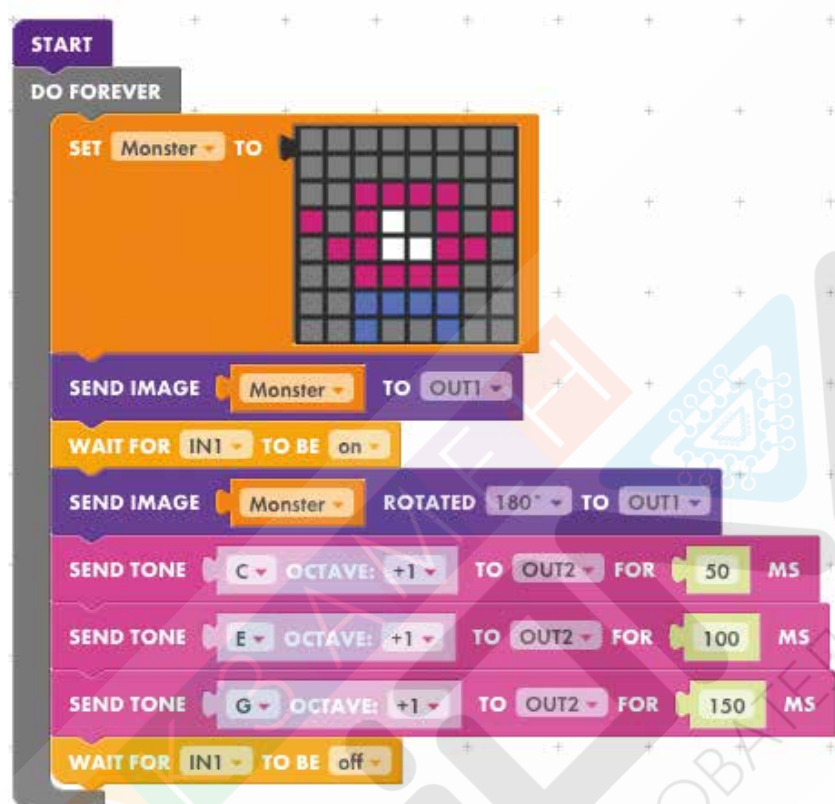
Блок цикла [DO FOREVER] заставляет написанный внутри него код повторяться бесконечное количество раз.

Блок переменной [SET Tone Length TO 500] позволит установить переменную, которой мы дали название «Tone Length» («Длина звука»). Эта переменная хранит число, в нашем случае 500. Вы можете изменить название переменной и ее значение на любые другие.

Блоки звуков [SEND TONE TO OUT 2 FOR [Tone Length] MS] отвечают за воспроизведение звуков из песни продолжительностью в 500 миллисекунд. Если вы поменяете значение переменной «Tone Length» («Длина звука»), то установленное значение появится во всех блоках звуков (как и в любом блоке, который вы включите в эту связку).

4.1 Спрайт «Прыгающий монстр»

Результат



Принцип работы

Блок цикла [DO FOREVER] заставляет написанный внутри него код повторяться бесконечное количество раз.

Блок переменной [SET MONSTER] позволит нам хранить изображение монстра в переменной, которая называется «Monster».

Блок изображения [SEND IMAGE Monster to OUT1] позволит вывести изображение нашего монстра на LED-матрицу (или экран компьютера).

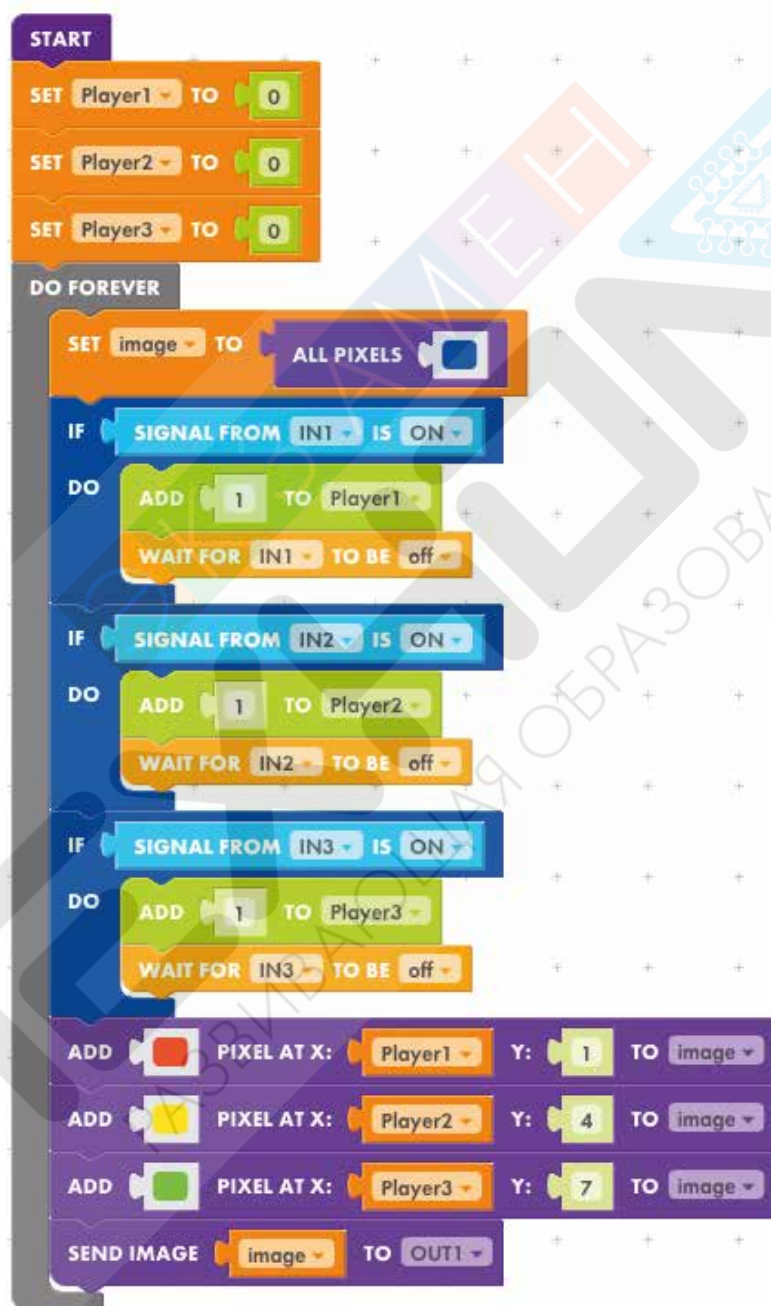
Блок ожидания [WAIT FOR IN1 TO BE on] заставляет программу ожидать, пока мы не нажмем на кнопку (ВКЛ). Когда мы нажмем на кнопку, блок изображения [SEND IMAGE Monster ROTATED 180 TO OUT 1] заставит изображение монстра повернуться на 180 градусов (вверх ногами).

Блоки звуков [SEND TONE TO OUT 2 FOR [Tone Length] MS] заставят программу воспроизвести звуковой сигнал, когда мы нажмем на кнопку и изображение монстра перевернется.

Блок ожидания [WAIT FOR IN1 TO BE on] говорит программе, что во время нажатия кнопки воспроизведение звуков и поворот изображения должны произойти только один раз. В противном случае поворот и воспроизведение звуков будут работать все время, пока нажата кнопка.

4.2 Гонка

Результат



Принцип работы

Начнем с того, что установим стартовые позиции участников гонки на оси X со значением 0 с помощью переменных.

Каждый из блоков переменных [SET Player 1 TO [0]] соответствует определенному участнику.

Блок цикла [DO FOREVER] заставляет написанный внутри него код повторяться бесконечное количество раз.

Установим переменную с изображением, используя блок переменной [SET image TO ALL PIXELS [■]]. Это изображение послужит фоном.

Затем воспользуемся тремя блоками [IF SIGNAL FROM IN 1 IS ON] (IF/DO), которые будут проверять, поступают ли сигналы (IF) с вводов IN1, IN 2 и IN 3 (значение ИСТИНА). Если сигнал поступает с ввода (значение ИСТИНА), то блок [ADD 1 TO Player1] будет увеличивать значение переменной с шагом 1 (DO).

Блок ожидания [WAIT FOR IN 1 TO BE off] заставит программу прекратить увеличивать значение переменной с шагом 1 при остановке поступающего сигнала с ввода (значение ЛОЖЬ).

Блок изображения [ADD [■] PIXEL AT X: Player 1 Y: [1] TO image] заставит программу выводить изображение на ось X. Позиция пикселей на LED-матрице будет равна значению переменной из предыдущего кода. Это создаст впечатление движущегося объекта на матрице. Далее стартовая позиция игрока перемещается на ось Y.

Блок изображения [SEND IMAGE TO OUT1] заставляет программу вывести фоновое изображение, которое мы установили ранее в качестве переменной image.



ШАГ 4: ДЕЛИМСЯ РЕЗУЛЬТАТАМИ

Продолжительность: 5 минут

Что мы узнали, чему научились?

Попросите учеников выполнить задания по переменным в рабочей тетради.



ШАГ 5: ЗАВЕРШЕНИЕ

Продолжительность: 5 минут

Попросите учеников разобрать все собранные схемы и убрать элементы обратно в коробки.



7. ФУНКЦИИ





ОБЗОР УРОКА

Класс обучения: 3-5 классы

Продолжительность занятия: 60 минут

Ученики выполняют ряд упражнений для изучения основ применения функций в программировании в приложении Fusebox app и узнают, как создать свой собственный блок для программирования.



ОСНОВНЫЕ МОМЕНТЫ

СТУПЕНЬ ОБРАЗОВАНИЯ

Начальная,
средняя

УЧЕБНЫЕ ДИСЦИПЛИНЫ

Информатика,
проектная
деятельность

СЛОЖНОСТЬ

Ознакомительная

ПРОДОЛЖИТЕЛЬ- НОСТЬ

60 минут

ТРЕБУЕМЫЕ НАВЫКИ

Понимание работы модулей набора LittleBits, умение программировать с помощью блоков, умение загружать программу в модуль codeBit™, пройденные уроки «Вводы-выводы», «Циклы» и «Логика»



НЕОБХОДИМЫЕ МАТЕРИАЛЫ

МОДУЛИ

Набор для программирования

ДРУГОЕ

Компьютер с установленным приложением Code Kit App, USB-порт

ХОД ЗАНЯТИЯ



ШАГ 1: НАСТРОЙКА

Продолжительность: 10 минут

Разделите учеников на группы по 2-3 человека. Каждой группе понадобится один набор для программирования, компьютер с установленным приложением Fusebox App и рабочая тетрадь для каждого ученика.



ШАГ 2: ВВЕДЕНИЕ

Продолжительность: 20 минут

Посмотрите ознакомительный ролик про использование функций в программировании в приложении Fusebox App.

Действия:

- Попросите 3-х добровольцев придумать простой танец (например, руки вверх, руки в стороны, шаг вправо). Придумайте название этому танцу.
- Затем эти 3 добровольца должны обучить остальной класс придуманному танцу. Инструкции могут быть только голосовые, то есть – ничего не показываем!
- После нескольких попыток воспроизвести танец попросите авторов танца показать оригинальную задумку. Сравните с результатами класса. Обсудите, насколько подробными должны быть инструкции.
- Играем в танец функций:
 1. Встаньте в круг и включите музыку.
 2. Каждый ученик должен задать танец своему соседу по очереди (руки вверх, либо руки в стороны, либо шаг вправо). В свою очередь сосед должен повторять заданное движение много раз. Повторяйте до тех пор, пока не останется не танцующих учеников.
 3. Перед окончанием песни попросите учеников произнести название танца. Все ученики прекратят повторять заданные движения и начнут танцевать весь танец целиком.

Обсудите разницу исполнения конкретного движения и исполнения танца целиком.



ШАГ 3: ТВОРЧЕСТВО

Продолжительность: 20 минут

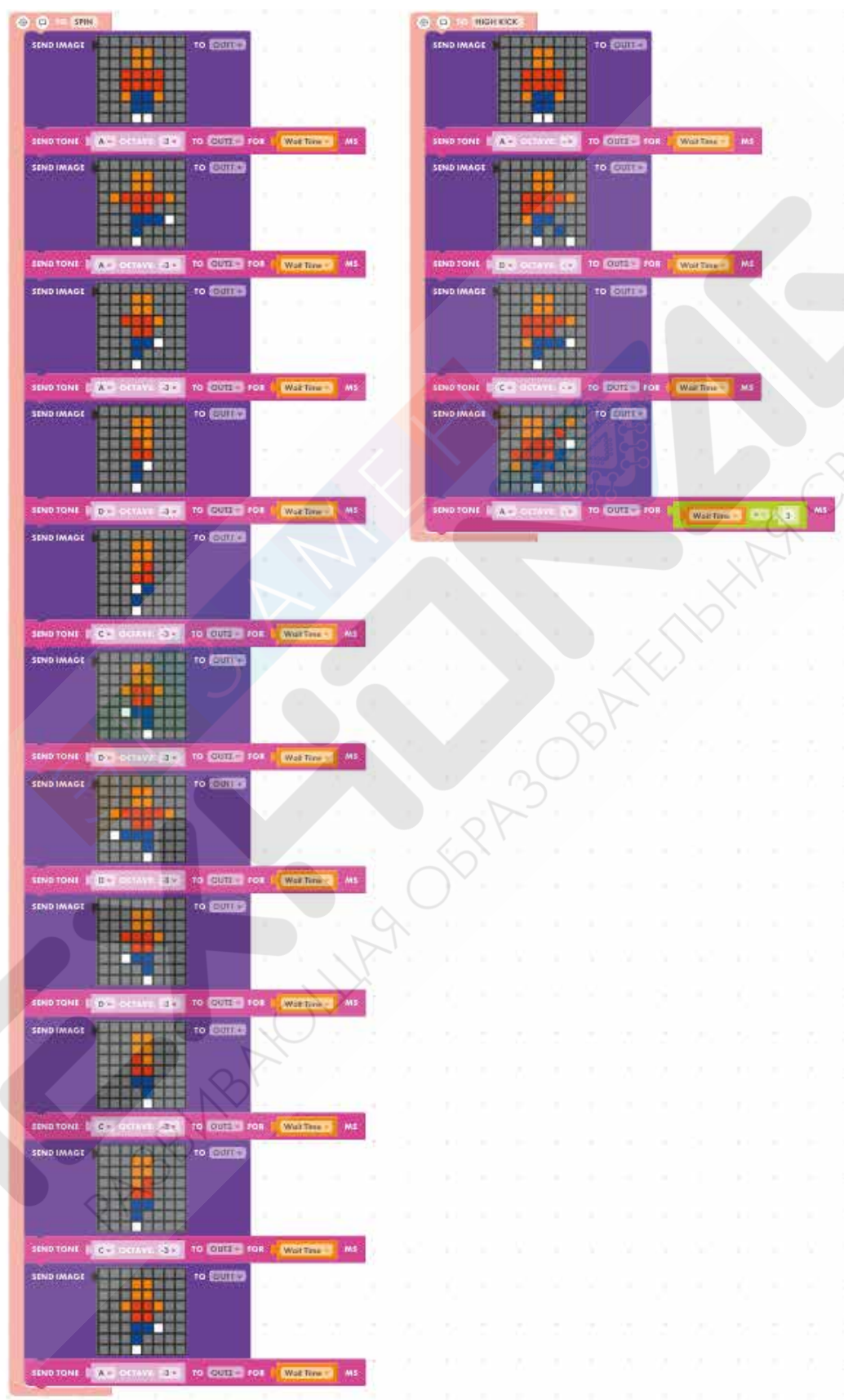
Создаем собственные блоки кода!

Воспользуйтесь приложением Fusebox App и пройдите ознакомительный

урок:

5.0 Супертанец





Если посмотреть на картинку выше, можно увидеть два программных кода в блоках. Слева располагается основной код, где мы задаем значение переменной [Wait Time to 125], которая расположена в блоке переменной [SET Wait Time TO 125]. С помощью этой переменной мы определяем продолжительность воспроизведения каждой ноты между кадрами анимации (более подробно их можно разглядеть на коде справа).

Блок цикла [DO FOREVER] заставляет написанный внутри него код повторяться бесконечное количество раз.

Внутри блока цикла располагаются блоки функции [SPIN], [HIGH KICK] и [SPIN, SPLIT].

Каждый из блоков функции подробно описан на изображении наверху. Внутри блока функции [SPLIT] находится блок изображения [SEND IMAGE TO OUT1] и блок звука [SEND TONE TO OUT 2 FOR Wait Time MS]. Последний блок звука использует дополнительный блок ожидания [Wait Time x 4], который увеличивает продолжительность соответствующего звукового сигнала в 4 раза.



ШАГ 4: ДЕЛИМСЯ РЕЗУЛЬТАТАМИ

Продолжительность: 5 минут

Что мы узнали, чему научились?

Попросите учеников выполнить задания по использованию функций в рабочей тетради.



ШАГ 5: ЗАВЕРШЕНИЕ

Продолжительность: 5 минут

Попросите учеников разобрать все собранные схемы и убрать элементы обратно в коробки.

8. УСТРАНЕНИЕ НЕПОЛАДОК



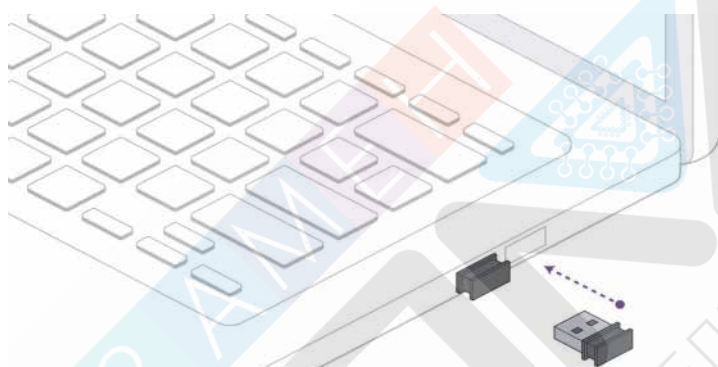
НЕПОЛАДКИ

Если в ходе работы с оборудованием возникли технические неполадки, ознакомьтесь с данным разделом, в котором вы можете найти решение возникшей проблемы.

1. Проверьте соединение

- **Вы точно подсоединили USB-адаптер?**

Если нет, подсоедините его. Возможно, вам потребуется перезагрузить компьютер.



- **Вы подключили модуль codeBit™ к приложению?**

Если это так, то вы увидите индикатор соединения в приложении. Если модуль не подключен, нажмите на кнопку «CONNECT YOUR CODEBIT», которая расположена в самом низу экрана.

- **Вы нажали кнопку Вкл/Выкл?**

Попробуйте отсоединить, затем присоединить r3 power Bit (Модуль питания) к своей цепи. Снова загрузите код. Если это не помогло, попробуйте перезагрузить приложение.

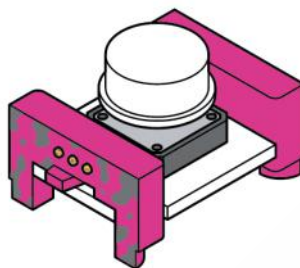
- **Напряжение?**

Вашей цепи требуется напряжение для работы. Убедитесь, что вы присоединили r3 power Bit (Модуль питания) к своей цепи.

2. Проверьте цепь

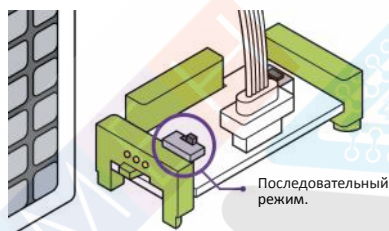
- **Чистка**

Время от времени на платы модулей попадает пыль. Попробуйте почистить модули салфеткой. Не следует делать это с помощью острых предметов.



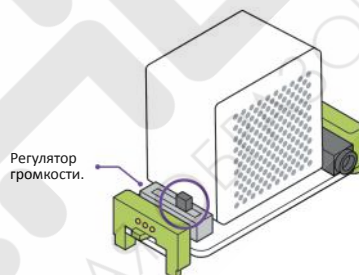
• Светодиодная матрица в порядке?

Убедитесь, что ваша матрица подключена и работает в правильном режиме.



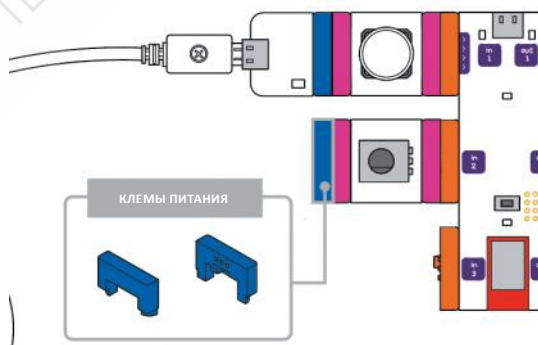
• Динамик в порядке?

Убедитесь, что громкость выставлена верно и ваш динамик подсоединен к OUT 2 или OUT 3.



• Везде ли вы использовали клеммы?

Используйте их!



3. Проверьте программный код

- **Блоки запрограммированы?**

Незаполненные блоки в программе не дадут программе запуститься.



- **Ваша программа начинается с блока [START]?**

Не забывайте соединять весь код с самым первым блоком, который запускает всю программу.

- **Вы верно указали входы и выходы в программе? Они должны совпадать с реальными значениями в цепи.**

Если вы использовали блок [WAIT FOR IN 3 TO BE ON], ваша цепь должна иметь вход IN 3.

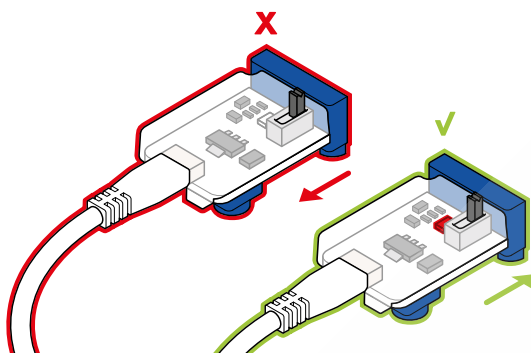
- **Запущен ли код?**

Вы можете убедиться в том, что программный код загружен и запущен, если взглянете на красную лампочку. Она должна гореть.



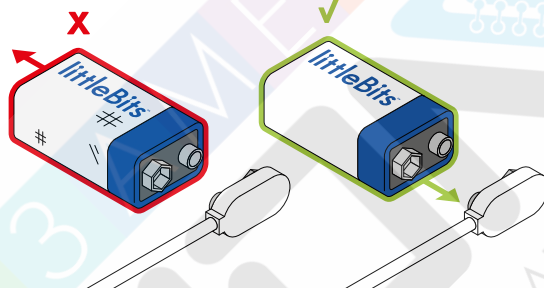
9. РЕШЕНИЕ ПРОБЛЕМ ВО ВРЕМЯ ЭКСПЛУАТАЦИИ КОМПОНЕНТОВ LITTLEBITS

1



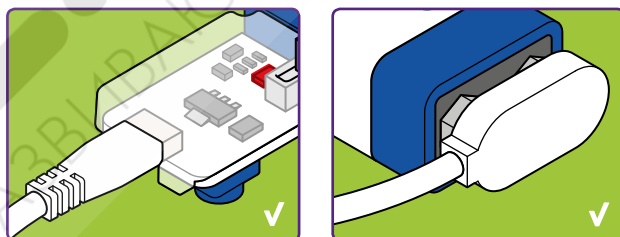
Убедитесь, что ваш модуль питания включен.
На плате должен загореться красный светодиод.

2



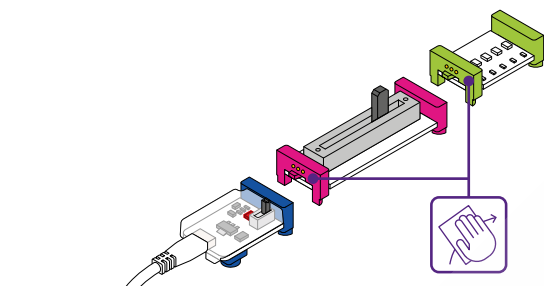
Попробуйте заменить батарею питания 9В на новую.
Низкий заряд батарей может привести к неустойчивой работе схемы. Компоненты littleBits имеют разные требования к мощности. Например: может оказаться, что двигатель постоянного тока не работает, в то время как свет все еще ярко светит.

3



Убедитесь, что кабель питания надежно прикреплен к модулю питания и к батарее.

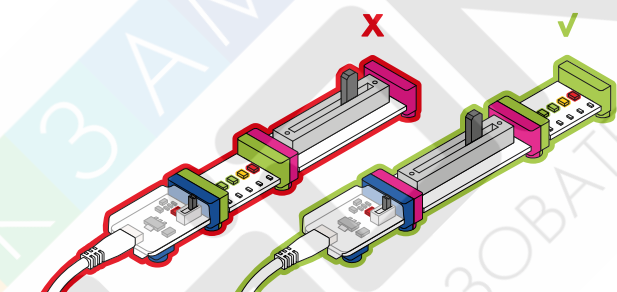
4



Проверьте соединение.

Проверьте, все ли компоненты littleBits надежно привязаны друг к другу. Иногда пыль мешает прочному соединению. Вы можете попробовать аккуратно протереть концы защелок мягкой тканью. Попробуйте разъединить, очистить компоненты littleBits и снова собрать все вместе.

5



Убедитесь, что компоненты littleBits расположены в правильном порядке.

Если выходной компонент littleBits в вашей цепочке стоит вначале, он не будет функционировать полноценно в вашей схеме.

УСТРАНЕНИЕ НЕПОЛАДОК

- Первым шагом является определение порядка действий, которому должны следовать ученики, если они столкнулись с проблемой. Некоторые преподаватели используют правило «спросите трех раньше меня», в котором ученики должны попросить помощи у трех сверстников, прежде чем спрашивать у преподавателя. Покажите учащимся, как использовать схемы сборки или другие доступные ресурсы для самостоятельного решения проблем.
- Другой вариант: совместно с учениками составьте контрольный список для устранения неполадок. Этот контрольный список может включать полезные советы. Например, отметьте все соединения, проверьте переключатели и винты, проверьте аккумулятор и т. д.
- Можно разместить плакаты с советами и рекомендациями по аудитории, чтобы помочь ученикам самостоятельно решать общие проблемы, и делиться креативными идеями для решения. Ученики могут предлагать любые решения, советы или приемы, которые они обнаружили, для создания коллективного ресурса, для использования littleBits.

МОЯ ЦЕПЬ НЕ РАБОТАЕТ

- Проверьте, включен ли источник питания. Переключатели маленькие, и иногда ученики совмещают контакты не до полного соединения. Проверьте, чтобы при включении светодиодный индикатор горел красным светом.
- Если вы используете аккумулятор, его нужно периодически заменять или заряжать. Для точного определения заряда батареи вам понадобится тестер батареи.
- Проверьте чистоту разъемов. Если на разъемах пыль, протрите их чистой сухой тканью.
- Проверьте, не окислились ли разъемы. Если какой-либо из трех электрических разъемов окислен, на нем будут темные пятна. В этом случае разъем можно протереть мягкой чистой тканью с небольшим количеством изопропилового спирта.

10. СПРОСИТЕ ТРЕХ РАНЬШЕ МЕНЯ







ФОРМА ОБРАТНОЙ СВЯЗИ

Для решения проблем и коллективного творчества бывает нужна четкая форма обратной связи. Предложите ученикам заполнить шаблон формы:

Имя:

Изобретение:

<p>ВАЖНОЕ</p> <p>Что вам действительно понравилось в изобретении?</p> 	<p>КОМПОНЕНТЫ</p> <p>Как вы думаете, что можно улучшить?</p> 
<p>ВОПРОС</p> <p>Какие у вас есть вопросы?</p> 	<p>ИДЕЯ</p> <p>Какие предложения у вас есть, чтобы сделать модель лучше?</p> 

Методические рекомендации

Голубев Алексей Константинович

Набор для программирования littleBits

Издательство **«ЭКЗАМЕН»**
«ЭКЗАМЕН-ТЕХНОЛАБ»

Гигиенический сертификат
№ РОСС RU C-RU.AK01.H.04670/19 с 23.07.2019 г.

Главный редактор *Л. Д. Лаппо*
Корректор *О. Ю. Казанцева*
Дизайн обложки
и компьютерная верстка *А. А. Винокуров*

107045, Россия, Москва, Луков пер., д. 8.
E-mail: по общим вопросам: robo@examen-technolab.ru;
www.examen-technolab.ru
www.littlebits.ru
по вопросам реализации: sale@examen-technolab.ru
тел. +7 (495) 641-00-23

Общероссийский классификатор продукции
ОК 034-2014; 58.11.1 – книги печатные

Отпечатано в соответствии с предоставленными материалами
в ООО «ИПК Парето-Принт», 170546, Россия, г. Тверь, www.pareto-print.ru