



## ScratchDuino.Лаборатория



Санкт-Петербург  
2015



ЗАО «Тырнет»

## ScratchDuino.Лаборатория

Санкт-Петербург  
ЗАО «Тырнет»  
2015

Рецензент — доктор педагогических наук, профессор  
заведующий кафедрой методики технологического образования  
РГПУ им. А. И. Герцена И. Б. Готская

**Вострикова Е. А.**

ScratchDuino.Лаборатория: руководство пользователя / Е. А. Вострикова,  
Л. С. Захаров, Е. А. Львова. — Санкт-Петербург : Множительный центр ЗАО  
«Тырнет», 2015. — 54 с.

Руководство пользователя предназначено для эксплуатации ScratchDuino. Лаборатория — электронного комплекса со встроенными датчиками звука и света, а также переменным резистором и интерфейсом для подключения дополнительных датчиков.

В данном руководстве приводится следующая информация:

1. Инструкции по установке программного обеспечения для трёх распространенных операционных систем.
2. Сведения о составе коробочной версии, методика тестирования и калибровки датчиков (сенсоров). Необходимые для работы сведения о языке программирования Scratch.
3. Примеры готовых скриптов по использованию сенсоров.
4. Инструкции по регистрации и размещению собственных проектов на портале [wiki.scratchduino.ru](http://wiki.scratchduino.ru), а также возможности сотрудничества детей и взрослых в рамках фестиваля «Свободная робототехника ScratchDuino».

## Оглавление

Установка программного обеспечения для ScratchDuino.Лаборатория .....	4
Операционные системы семейства Windows .....	4
Операционные системы семейства Linux .....	8
Операционные системы Macintosh Operating System.....	12
Состав комплекта .....	13
Тестирование и калибровка сенсоров.....	15
Основные понятия Scratch .....	19
Типы алгоритмов Scratch .....	22
Проекты применения элементов платы ScratchDuino.Лаборатория.....	28
Кнопки .....	28
Рычажок.....	29
Звуковой сенсор .....	30
Выносной температурный датчик.....	33
Внешние датчики .....	35
Развиваем сообщество единомышленников .....	38
Регистрация на wiki-портале проекта ScratchDuino .....	38
Размещение фестивальной работы на <a href="http://wiki.scratchduino.ru">wiki.scratchduino.ru</a> .....	44
Правила обсуждения фестивальных проектов на портале <a href="http://wiki.scratchduino.ru">wiki.scratchduino.ru</a> .....	47
Использованные источники.....	50
Приложение .....	51
Положение о фестивале «Свободная робототехника ScratchDuino» .....	51

## Установка программного обеспечения для ScratchDuino.Лаборатория

### Операционные системы семейства Windows



Для успешной работы с Лабораторией под ОС Windows необходимо, как минимум, установить визуальную среду программирования Scratch и драйвер Arduino UNO. Кроме этого, продвинутые пользователи могут установить среду для программирования Arduino вместе с Arduino IDE.

Данное программное обеспечение (далее — ПО) можно взять из трёх источников:

- с диска, который идет в комплекте со ScratchDuino.Лаборатория;
- с файл-сервера проекта <http://files.scratchduino.ru/>;
- скачать с сайта официальных разработчиков.

### Установка с диска или с файл-сервера проекта

*Установка Scratch:* запустите инсталляционный файл с диска `../windows/Scratchduino.exe` либо скачайте и запустите инсталляционный файл с сервера —

<http://files.scratchduino.ru/Software/Windows/Scratchduino.exe>

Мастер установки предложит выбрать язык. Выберите Русский; затем нажмите Ок, Далее и Установить. Через несколько секунд на рабочем столе (или в списке программ в кнопке Пуск, это зависит от индивидуальных настроек) появятся две иконки — Scratch и Scratchduino Robot (рис. 1).

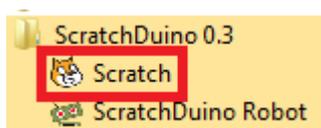


Рис. 1. Вид иконок.



Для работы со Scratchduino.Лаборатория предназначено ПО Scratch! Чтобы запустить его, щёлкните по иконке Кота.



*Установка драйвера:* запустите исполняемый файл с диска `../windows/ScratchDuino_drivers.exe` либо скачайте и запустите исполняемый файл с сервера —

[http://files.scratchduino.ru/Software/Windows/ScratchDuino\\_drivers.exe](http://files.scratchduino.ru/Software/Windows/ScratchDuino_drivers.exe).

В диалоговых окнах мастера установки нажмите Yes, Далее, Установить и Готово.

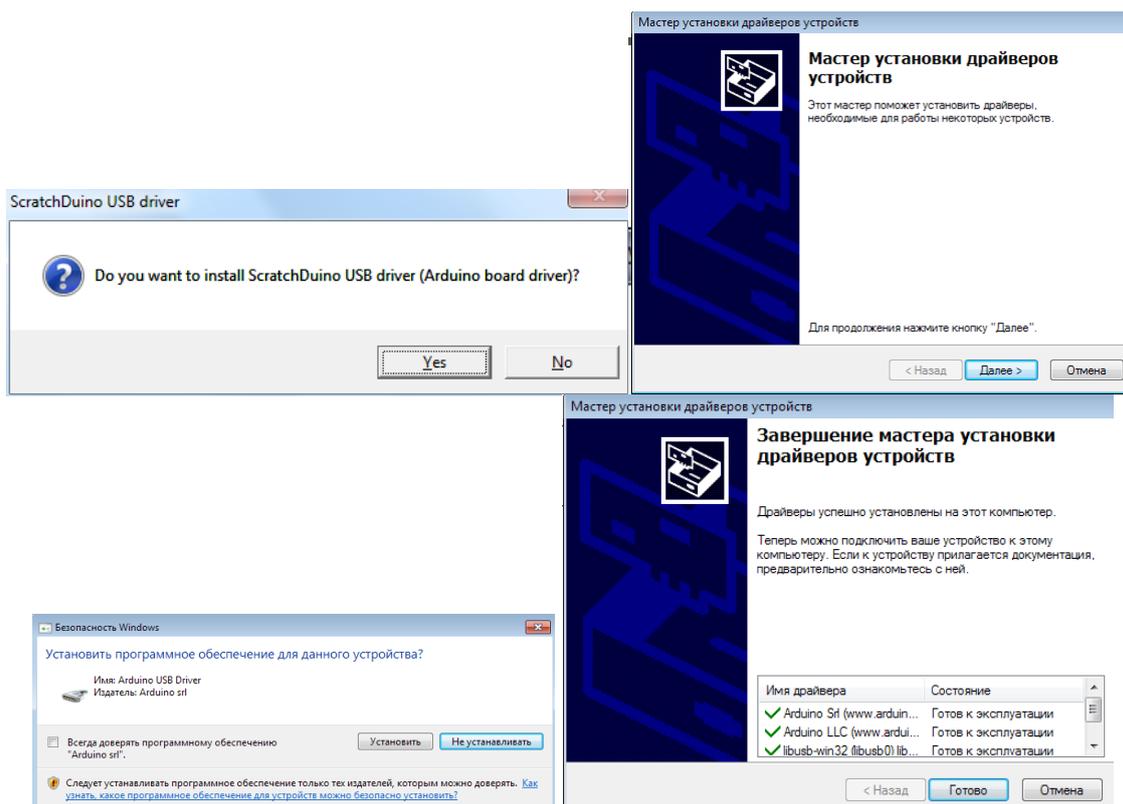


Рис. 2. Вид диалоговых окон.

**Замечание для пользователей:** ПО Arduino IDE не размещено ни на диске, ни на файл-сервере проекта, так как при установке Arduino IDE пользователь может перепрограммировать заранее запрограммированный картридж ArduinoUNO. Неопытным пользователям во избежание недоразумений не рекомендуется устанавливать Arduino IDE. Опытные пользователи могут установить ПО Arduino IDE с сайта разработчика (см. следующий пункт).

## Установка ПО с сайтов официальных разработчиков

1. Скачайте инсталлятор Scratch версии 1.4 с сайта разработчика:

<http://download.scratch.mit.edu/ScratchInstaller1.4.exe>.

Запустите инсталлятор, выберите Русский в качестве языка установки и следуйте указаниям Мастера установки. После установки появится иконка Scratch (Кот).



2. Скачайте последнюю версию ПО Arduino IDE с сайта разработчика

<https://www.arduino.cc/en/Main/Software> и следуйте инструкции по уста-

новке — <https://www.arduino.cc/en/Guide/Windows>.

## Подсоединение ScratchDuino.Лаборатория через USB-кабель

Подключите USB-кабель, идущий в комплекте со ScratchDuino.Лаборатория, в USB-разъём компьютера и к картриджу Arduino. Для обеспечения связи Scratch и ScratchDuino.Лаборатория необходимо узнать последовательный порт подключения ScratchDuino.Лаборатория к компьютеру. Для этого необходимо попасть в Диспетчер устройств, который, как правило, находится в Панели управления.

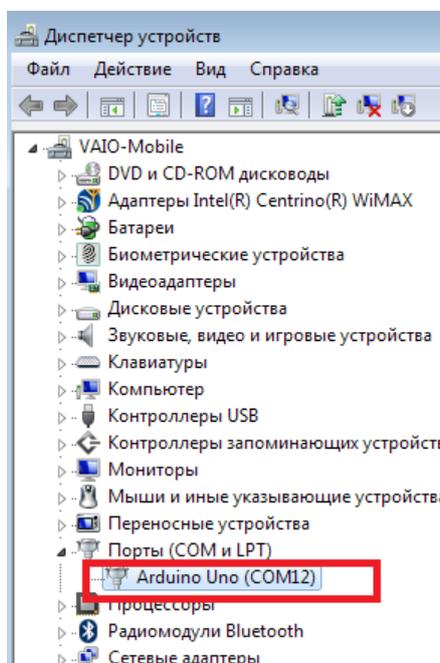


Рис. 3. Фрагмент окна Диспетчер устройств: выводит на экран номер порта, к которому автоматически подключено устройство.



Запомните номер COM-порта! Именно этот порт необходимо будет выбирать в Scratch. На рис. 3 ArduinoUno соответствует порт COM12.

### Подключение Scratchduino.Лаборатория и выбор порта в Scratch

Запустите Scratch. В открывшемся окне Scratch выберите синий кирпичик Сенсоры, далее щёлкните правой кнопкой мыши по блоку Значение сенсора и выберите показать данные ScratchBoard (рис. 4).

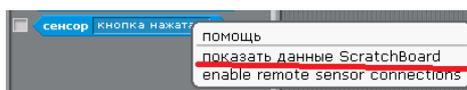


Рис. 4. Контекстное меню при нажатии правой кнопкой мыши на блок Сенсор.

Щёлкните правой кнопкой мыши по серому полю и выберите Выбрать серийный/usb порт (рис. 5).

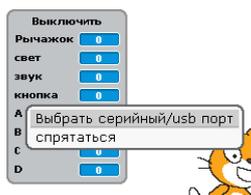


Рис. 5. Контекстное меню при нажатии правой кнопкой мыши на ScratchBoard.

В открывшемся перечне портов выберите тот, к которому подключилась Scratchduino.Лаборатория.

## Операционные системы семейства Linux



Существует три способа установки ПО для ScratchDuino. Лаборатория под ОС семейства GNU/Linux:

- воспользоваться диском, идущим в комплекте;
- скачать пакеты с файл-сервера проекта —

<http://files.scratchduino.ru/>;

— произвести установку из специально созданного репозитория, содержащего установочные пакеты для ПО ScratchDuino. Лаборатория: <http://download.opensuse.org/repositories/home:/scratchduino/>.

**!** Не следует устанавливать `scratch_1.4` и виртуальную машину `squeak-vm` через официальные репозитории вашего дистрибутива или скачивать с сайта разработчика. Такой Scratch не сможет работать с последовательным портом, а значит, не будет возможности программировать ScratchDuino.Лаборатория через Scratch. Виртуальная машина `squeak-vm`, на базе которой работает Scratch 1.4, должна быть версии 4.0.3-2202.

### Установка с диска

Пакеты для установки ПО под ОС семейства Linux находятся на диске в папке `../linux`. Установочные пакеты, собранные под ряд наиболее распространённых дистрибутивов (RHEL/CentOS, Fedora, Ubuntu, Debian, openSUSE, SLED), находятся в папке с соответствующим названием и версией дистрибутива. Если необходимого дистрибутива нет, то выберите дистрибутив, наиболее родственному требуемому. Так, для установки Scratch на Linux Mint 17 воспользуйтесь пакетами из папки `../linux/xUbuntu14.04/`.

Для Deb-дистрибутивов установите пакеты `scratch 1.4.0.7`, `squeak-vm 4.0.3`, `squeak-plugins-scratch`.

Для RPM-дистрибутивов установите пакеты `squeak-vm 4.0.3`, `scratch 1.4.0.7`.

После установки в графическом интерфейсе дистрибутива появится иконка запуска Scratch (Кот).



Кроме того, вы можете запустить Scratch из терминала — для этого в командной строке наберите `scratch`.

## Установка с файл-сервера проекта

Установка аналогична установке с диска. Скачайте пакеты, соответствующие вашему дистрибутиву, с сайта

<http://files.scratchduino.ru/Software/Linux/>, и установите их.

## Установка из репозитория

Подключите репозиторий для вашего дистрибутива, как описано ниже:

— для openSUSE и SLE введите в консоли команду

```
$ sudo zypper addrepo  
http://download.opensuse.org/repositories/home:/scratchduino/XXXXXXXX/  
ScratchDuino
```

— для CentOS, RHEL, Fedora и ScientificLinux добавьте файл `/etc/yum.repos.d/scratchduino.repo`, со следующим содержимым:

```
[scratchduino]  
name=ScratchDuino  
type=rpm-md  
baseurl=http://download.opensuse.org/repositories/home:/scratchduino  
/XXXXXXXX/  
gpgcheck=1  
gpgkey=http://download.opensuse.org/repositories/home:/scratchduino/  
XXXXXXXX/repdata/repomd.xml.key  
enabled=1
```

— для Debian и Ubuntu добавьте в файл `/etc/apt/sources.list` строчку  
deb

```
http://download.opensuse.org/repositories/home:/scratchduino/XXXXXXXX/ /
```



Во всех случаях XXXXXXXX следует заменить на название и версию дистрибутива.

Для Deb-дистрибутивов установите из подключенного репозитория пакеты `scratch 1.4.0.7`, `squeak-vm 4.0.3`, `squeak-plugins-scratch`.

Для RPM-дистрибутивов установите из подключённого репозитория пакеты `squeak-vm 4.0.3`, `scratch 1.4.0.7`.

## Выполнение пост-установочных настроек

После успешной установки ПО для работы со ScratchDuino.Лаборатория необходимо провести ряд пост-установочных манипуляций.

— Доступ к Arduino UNO осуществляется через `/dev/ttyACM<номер>`. По умолчанию к устройствам `/dev/tty*` имеют доступ пользователи из группы `dialout`. Поэтому пользователь, от имени которого будет осуществляться работа со ScratchDuino.Лаборатория, должен входить в группу `dialout`.



Добавьте пользователя в группу `dialout`!

— Если используется дистрибутив Ubuntu 12.04 или новее, либо дистрибутив, основанный на Ubuntu (например, Mint), то необходимо обратить внимание на то, какой используется менеджер окон (WM, windows manager). По умолчанию используется `Compiz`. К сожалению, использование `Compiz` может вызвать конфликт со Scratch. Поэтому рекомендуется установить пакет `gnome-session-fallback` (для Ubuntu 14.04 и новее — `gnome-session-flashback`), после чего на экране логина (рис. 6) будет доступна кнопка смены WM.

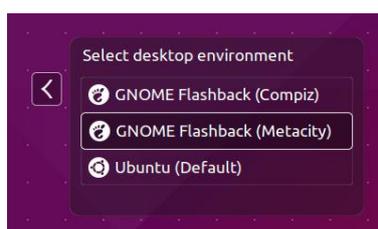


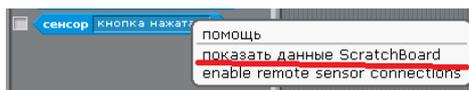
Рис. 6. Экран логина.



Выбирайте при логине менеджер окон `Metacity`!

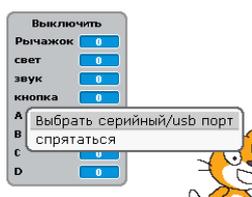
## Подключение Scratchduino. Лаборатория и выбор порта в Scratch

Запустите Scratch из графического интерфейса или из консоли. В открывшемся окне Scratch выберите синий кирпичик Сенсоры, далее щёлкните правой кнопкой мыши по блоку значение сенсора и выберите показать данные ScratchBoard (рис. 7).



*Рис. 7. Контекстное меню при нажатии правой кнопкой мыши на блок Сенсор.*

Далее щёлкните правой кнопкой мыши по серому полю и выберите Выбрать серийный/usb порт (рис. 8).



*Рис. 8. Контекстное меню при нажатии правой кнопкой мыши на ScratchBoard.*

В появившемся списке выберите `/dev/ttyACM0`.

## Операционные системы Macintosh Operating System



Mac

Для установки ПО под MAC ОС можно воспользоваться диском, идущим в комплекте со ScratchDuino.Лаборатория, либо скачать установочные файлы с файл-сервера проекта. На диске установочный файл находится в директории ../mac/Scratch.dmg.

Ссылка для скачивания с файл-сервера проекта <http://files.scratchduino.ru/Software/Mac/Scratch.dmg>.

Для установки ПО просто откройте Scratch.dmg и перетащите папку Scratch в папку Application (рис. 9).

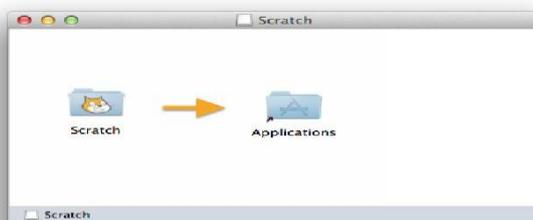


Рис. 9. Вид окна.

Откройте Application файл-менеджером, зайдите в папку Scratch и запустите исполняемый файл (рис. 10).

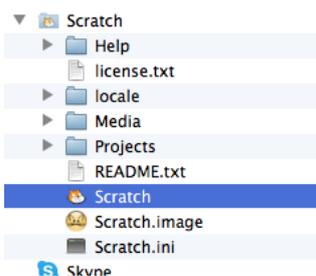


Рис. 10. Запуск исполняемого файла.

### Подключение Scratchduino.Лаборатория и выбор порта в Scratch

В открывшемся окне Scratch выберите синий кирпичик Сенсоры, далее щёлкните правой кнопкой мыши по блоку значение сенсора и выберите показать данные ScratchBoard (см. рис. 7). Далее щёлкните правой кнопкой мыши по серому полю и выберите select serial/USB port (Выбрать серийный/usb порт) (см. рис. 8). В появившемся списке выберите /dev/cu.usbmodem1411.

## Состав комплекта

ScratchDuino.Лаборатория («Цифровая лаборатория») предназначена для проведения лабораторных и демонстрационных экспериментов и учебных исследовательских проектов и практикумов на уроках информатики, технологии и уроках естественнонаучного цикла начальной, основной и старшей школы, в 5–11 классах.

Оборудование включает:

- микроконтроллер Arduino Uno с возможностью автономной работы и подключения к компьютеру с экраном;
- датчики, встроенные на плату (см. рис. 13);
- датчики подключаемые (рис. 11), предназначенные для снятия показаний параметров объектов исследования внешними исполнительными устройствами.



*Рис. 11. Выносной датчик температуры и соединительный шнур с тремя зажимами-«крокодилами».*



*Рис. 12. Кабель длиной 180 см для подключения Лаборатории к компьютеру.*

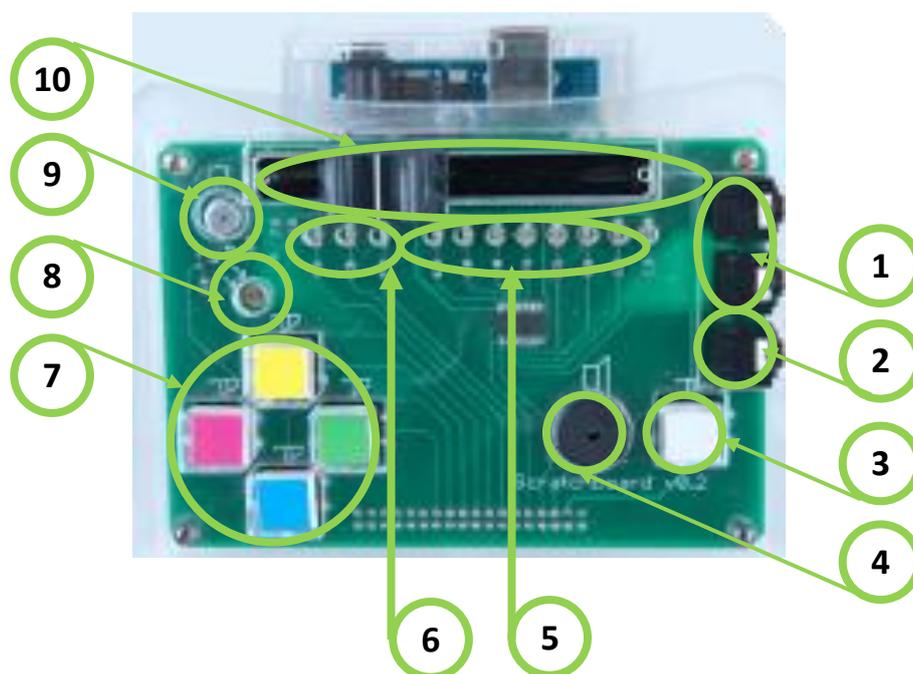


Рис. 13. Элементы платы ScratchDuino.Лаборатория:

- 1 — Разъёмы для дополнительных аналоговых датчиков (переменные<sup>1</sup> «Сопротивление А» и «Сопротивление В»); 2 — Цифровой вход для внешнего устройства\*; 3 — Кнопка (переменная «кнопка»); 4 — Динамик; 5 — Восемь светодиодов, управляемых матрично\*; 6 — Три светодиода на отдельных цифровых выходах (красный, жёлтый, зелёный)\*; 7 — Четыре кнопки (переменные: «Сопротивление А», «Сопротивление В», «Сопротивление С», «Сопротивление D»); 8 — Датчик света (переменная «Свет»); 9 — Датчик звука (переменная «Звук»); 10 — Переменный резистор (переменная «Рычажок»).

\* Дополнительные элементы, требующие модифицированной версии ScratchBoard.

В комплект также входят:

- кабель для подключения к компьютеру (см. рис. 12);
- диск с программным обеспечением;
- руководство пользователя;
- гарантийный талон.

<sup>1</sup> Данные переменные используются в языке программирования *Scratch*, о котором пойдет речь ниже.

## Тестирование и калибровка сенсоров

Для составления программ важно исследовать чувствительность датчиков и значение кнопок ScratchDuino.Лаборатории. Несмотря на стандартную комплектацию, датчики в каждой ScratchDuino.Лаборатории имеют свои диапазоны чувствительности.

Таблица 1. Чувствительность датчиков и значение кнопок ScratchDuino.Лаборатории

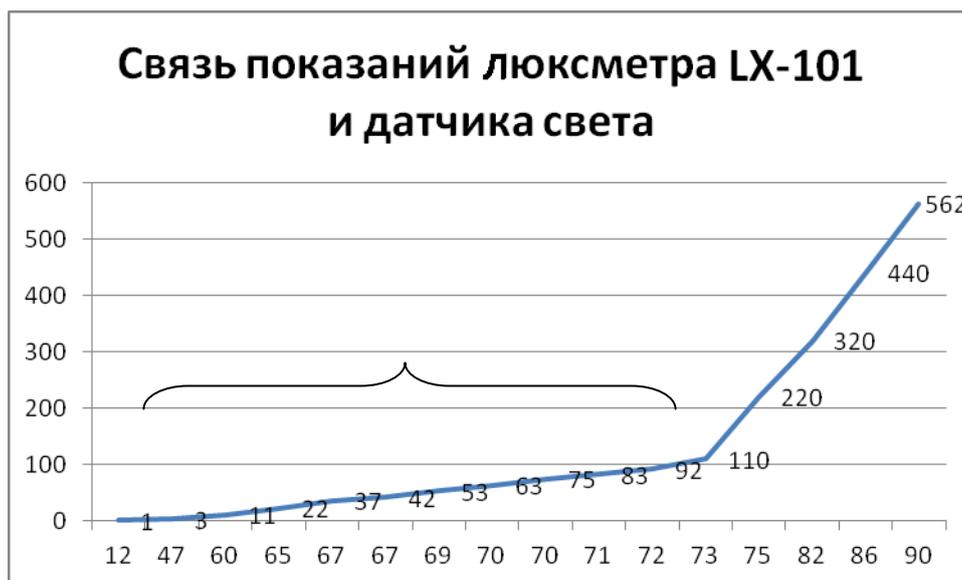
Название датчика или кнопки (имя переменной)	Min	Max
Датчик света (переменная «Свет»)	12	98
Датчик звука (переменная «Звук»)	23	100
Переменный резистор (переменная «Рычажок»)	0	100
Кнопка чёрная (переменная «кнопка»)	false	true
Четыре кнопки (переменные: «Сопротивление А», «Сопротивление В», «Сопротивление С», «Сопротивление D»)	0	100
Датчик температуры (после подключения используется переменная «Сопротивление А»)	0,6	22,5

### Калибровка датчика света

Для калибровки датчика света используем измеритель освещённости. В данном случае использовался цифровой люксметр LX101. Результаты измерений приведены в таблице 2 и на рис. 14.

Таблица 2. Связь показаний датчика света ScratchDuino.Лаборатории и люксметра LX-101

Люксметр LX-101, лк	1	3	11	22	37	42	53	63	75	83	92	110	220	320	440	562
Датчик света, ед.	12	47	60	65	67	67	69	70	70	71	72	73	75	82	86	90



*Рис. 14. Зона линейной зависимости (зона достоверности) на графике показаний люксметра LX-101 и датчика света ScratchDuino.Лаборатории.*

Если использовать справочник по рекомендованному уровню освещённости рабочих мест из инструкции к люксметру LX-101 (рис. 15), то можно сделать вывод о том, что встроенный датчик света ScratchDuino.Лаборатории с большой степенью достоверности может быть использован как датчик освещённости до 100 лк по формуле 1. Данная формула получена в результате построения линии тренда (аппроксимация функции) для значений от 12 до 73 датчика света ScratchDuino.Лаборатории.

$$y = x - 15,62 \quad (1)$$

зона достоверности датчика

ЛЮКСЫ МЕСТОРАСПОЛОЖЕНИЕ	1000	5000	3000	2000	1500	1000	750	500	300	200	150	100	75	50	30	20
ОФИС				Печать на клавиатуре Чертежные работы	Офисная работа		Конференц-зал Столовая Приемная		Коридор Лестницы	Вход Склад	Внутренние пожарные лестницы					
ДОМ				Шитье	Чтение Обучение	Макияж	Обеденный стол	Все виды отдыха	Стирка							
БОЛЬНИЦА	Проверка глаз			Операционная Скорая помощь		Смотровая Столовая	Приемная	Кабинет врача Склад	Лестницы	Пожарные лестницы						
ШКОЛА				Чертежный кабинет Библиотека	Классные комнаты		Спортзал Уборная Аудитории		Пожарные лестницы							

*Рис. 15. Требования к уровню освещённости рабочих мест в лк, а также зона достоверности датчика света ScratchDuino.Лаборатории.*

### Калибровка датчика звука

Измерения проводятся в комнате при полной тишине. В этом случае сенсор показывает значение 23 ед. Естественные звуки (голос, музыка и т.д.) соответствуют диапазону 60–80 ед. Максимум в 100 ед. можно добиться, энергично дунув в датчик. Для выявления функциональной зависимости показаний датчика звука в децибелах (дБ) необходимо использовать цифровой шумомер. Методика проведения измерений и выведение формулы описаны в калибровке датчиков света и температуры.

### Значения переменного резистора Рычажок

В крайнем левом положении — 0, в крайнем правом — 100.

### Значения чёрной кнопки

При нажатии значение меняется с false на true.

### Значения цветных кнопок

При нажатии значение меняется с 0 на 100. Контактная группа у всех кнопок обычная: пара контактов неподвижного типа, между которыми расположен подвижный контакт. Подвижный контакт связан со штоком, выходящим наружу из корпуса платы.

### Калибровка датчика температуры

Для калибровки температурного датчика проведём измерения температуры воды в процессе непрерывного нагревания и занесем значения, взятые с термометра и с сенсорной панели, в таблицу 3. Построим на основе табличных данных график (рис. 16).

*Таблица 3. Связь показаний выносного датчика температуры и бытового термометра*

Термометр, градусы	0	5	10	15	20	25	30	35	40	45	50
Датчик температуры, ед.	0,6	2,2	4,3	6,7	9	11,1	13,7	15,8	17,9	20,2	22,5

Построив линию тренда (аппроксимация функции), можно увидеть, что зависимость показаний выносного датчика от температуры воды, измеряемой бытовым термометром, *линейная* и выражается формулой 2:

$$y = 0,4497x - 0,4165 \quad (2)$$



*Рис. 16. График зависимости показаний выносного датчика от температуры, измеряемой бытовым термометром.*

Для целого ряда проектов может потребоваться не только прямая, но и обратная зависимость показаний температуры, измеряемой бытовым термометром, от показаний выносного датчика. Для этого сразу выведем обратную зависимость, выразив  $x$  через  $y$ :

$$x = 2,2237y + 0,9261 \quad (3)$$

**!** Калибровка датчиков является одним из самостоятельных направлений учебных исследований, проводимых с использованием ScratchDuino.Лаборатории. Измерения могут проводиться разными измерительными приборами и по разным методикам. Погрешности измерения могут различаться. В рамках сообщества ScratchDuino, о котором речь пойдет ниже, можно представить свои измерения и формулы. В сообществе также можно познакомиться с опытом других участников по калибровке датчиков, входящих в комплект и дополнительных.

## Основные понятия Scratch

*Scratch* — компьютерная модель реального мира. Окно программы с элементами интерфейса показано на рис. 17.

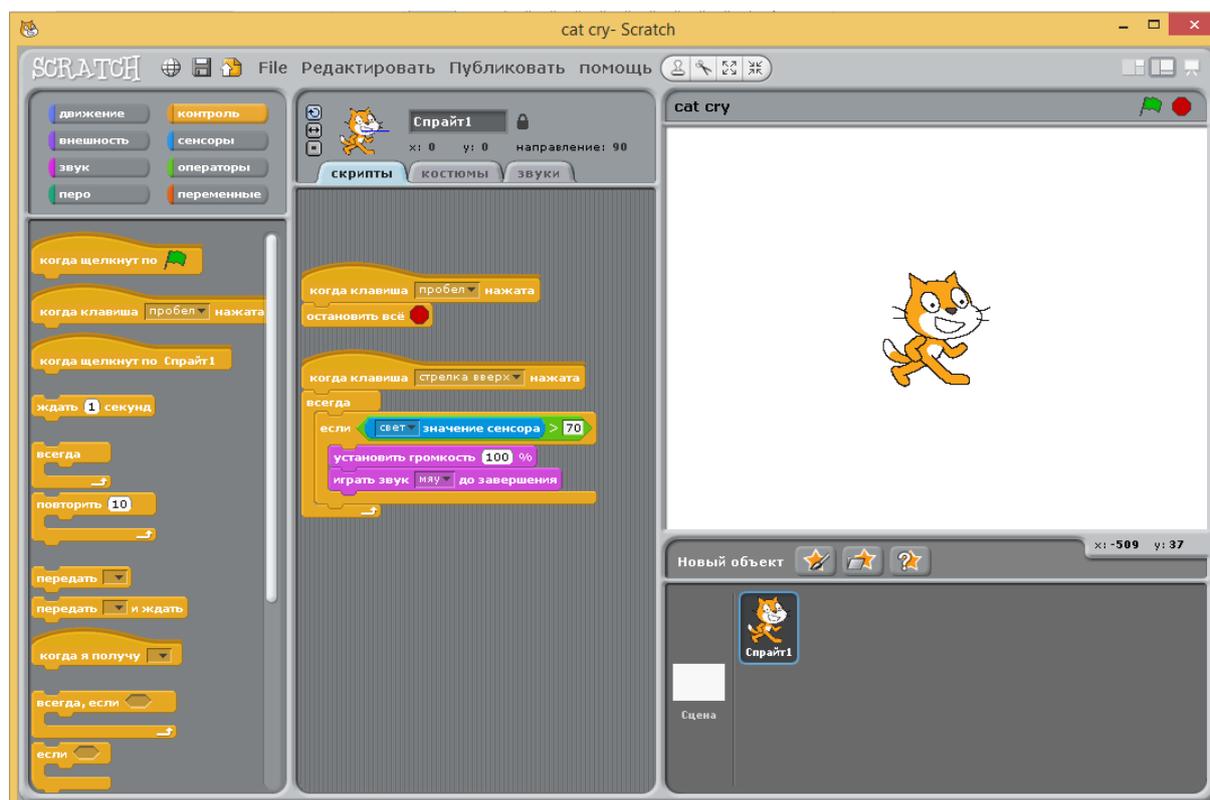


Рис. 17. Окно Scratch: 1 — информационная панель; 2 — группы команд; 3 — зона написания скриптов; 4 — группа Контроль; 5 — Исполнитель (спрайт); 6 — фон текущей сцены; 7 — зона спрайтов; 8 — зона сцены.

Мир *Scratch* состоит из множества объектов (*лат.* objectum — предмет), обитающих в общем пространстве. *Объекты* — это всё, что существует в природе: люди, животные, ветер, снег, дерево, солнце, буквы, мороженое, конфеты и т.д.

Объекты могут также быть *Исполнителями* алгоритмов.

*Алгоритм* — это точное предписание, которое определяет действия Исполнителя, ведущие от начальных данных (*например, взятых с датчиков Scratch-Duino.Лаборатория*) к требуемому результату. Разработка алгоритма — это творческий процесс. Алгоритм может быть представлен в виде *скрипта*.

*Скрипты* в Scratch состояются из готовых блоков-команд, похожих на блоки конструктора Лего. Синтаксис Scratch интуитивно понятен. *Скрипт* — это несколько соединённых блоков. Сами блоки и их порядок очень важен, так как они определяют, что будет делать Исполнитель.

В Scratch всего 2 вида исполнителей алгоритмов: *Сцена* (пространство, где происходят события) и *спрайты* (спрайт —англ. Sprite — вымышленный герой, эльф). Описание сюжетов в Scratch основано на использовании алгоритмов.

*Спрайты*, созданные пользователями, загруженные или найденные в библиотеке спрайтов, являются объектами (Исполнителями), которые выполняют действия в проекте. Во многих проектах есть как минимум один спрайт, который в отличие от сцены может передвигаться по экрану проекта.

Помимо выполнения команд, спрайт может менять свой *костюм*. Менять вид спрайта можно напрямую или с помощью команд в области редактирования скриптов. Для смены костюма нужно перейти во вкладку *Костюмы*, которая находится рядом с вкладками *Скрипты* и *Звуки*. Во вкладке «Костюмы» находится весь список костюмов, которые можно редактировать, импортировать из библиотеки спрайтов или с компьютера. Можно создать новый костюм.

*Сцена* содержит набор изображений, на фоне которых действуют спрайты. При включении программы автоматически создается фон1: белый прямоугольник, имеющий 480 точек по ширине и 360 точек по высоте (рис. 18). В компьютерной графике минимальную точку изображения и экрана называют *пиксель*.

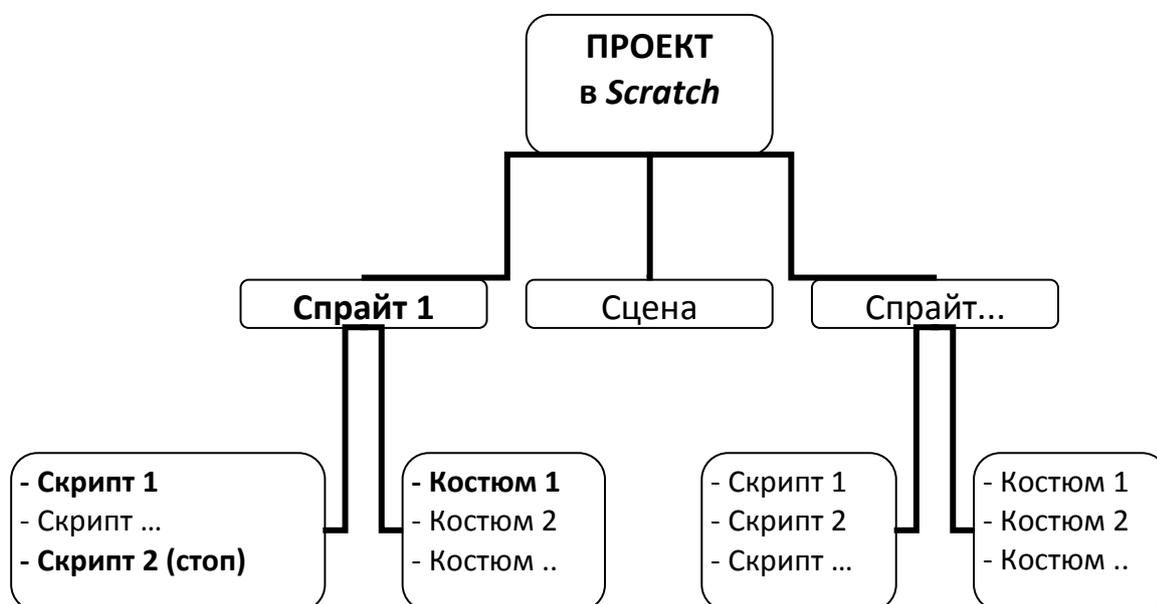


Рис. 18. Вкладка *Фоны*.

*Система команд* спрайтов состоит из 125 команд, а сцены — из 85. Этот набор позволяет спрайтам выполнять множество разнообразных алгоритмов. Все команды находятся в левом окне программы и объединены в 8 групп, визуально выделенных разными цветами: движение, внешность, звук, перо, контроль, сенсоры, операторы, переменные.

*Скрипт* — это несколько соединённых блоков в Scratch.

*Проекты* в Scratch состоят из нескольких скриптов для спрайтов, выполняемых одновременно (параллельно) с использованием одного или нескольких костюмов (рис. 19).



*Рис. 19. Структура проекта в Scratch. Выделена обязательная часть. Остальное зависит от авторского замысла.*

Для описания проектов в Scratch будем применять следующий план:

- тема проекта;
- описание требований к выполнению проекта;
- описание хода выполнения проекта и/или пояснения к скрипту;
- рисунок скрипта с изображением спрайта, для которого скрипт составлен.

## Типы алгоритмов Scratch

По структуре выполнения алгоритмы делятся на три вида: *линейные*, *ветвящиеся* и *циклические*.

**Линейный алгоритм** — это алгоритм, в котором все действия выполняются последовательно друг за другом и только один раз. Схема представляет собой последовательность блоков, которые располагаются сверху вниз в порядке их выполнения.

*Тема проекта:* «Краб рисует лесенку».

*Условия выполнения проекта:*

- исполнитель — Краб;
- процесс рисования лесенки выполняется как в мультфильме;
- реализуется простая возможность менять число ступенек и их размер;
- выполнение скрипта должно проходить, если значение резистора (переменная «Рычажок») больше 50.

*Выполнение проекта «Краб рисует лесенку» с использованием линейного алгоритма.*

Выберем в качестве исполнителя алгоритма (спрайта) Краба и научим его рисовать лесенку. Краб живет на прямоугольном поле размером 480×360 пикселей. Он умеет находить место на поле *по его координатам*<sup>2</sup>. Краб умеет: перемещаться на указанное количество шагов; ждать; поворачиваться влево и вправо на 0, 90 и 180 градусов. У Краба есть перо, которое оставляет след. Пример скрипта для Краба приведен на рис. 20. Согласно этому скрипту, Краб рисует 3 ступеньки (рис. 21). Начинается исполнение скрипта при нажатой на клавиатуре клавише «Стрелка вверх».

При использовании только линейного алгоритма:

- процесс построения лесенки не виден, но рисунок появляется мгновенно;

---

<sup>2</sup> Методически целесообразно начинать освоение Scratch с 5-го класса, так как необходимо знание учащимися таких математических понятий, как «координатная плоскость», «положительные и отрицательные числа», «десятичная дробь», «градусная мера угла» и т.д. Однако простые проекты могут быть реализованы и в начальной школе.

- изменение количества ступенек требует увеличения объёма скрипта и редактирования числовых значений в каждом блоке, поэтому данный способ нельзя назвать простым;
- требования по использованию условия с Рычажком реализовать невозможно.



Рис. 20. Линейный алгоритм рисования лестницы.

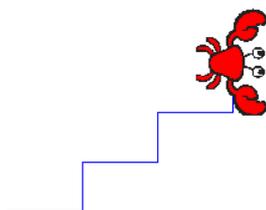


Рис. 21. Результат выполнения линейного скрипта для Спрайта 1 в костюме Краба на сцене с белым фоном (рисунок из коллекции crab1-a).

**Вывод 1.** При использовании линейного алгоритма невозможно выполнить все требования проекта.

**Алгоритмы циклической структуры.** *Цикл* — последовательность команд, повторяемая до тех пор, пока не будет выполнено заданное условие. Циклическое описание многократно повторяемых процессов значительно снижает трудоемкость написания программ. В *Scratch* есть блоки для 4-х видов циклов: безусловный, со счётчиком, с предусловием и с пост-условием (рис. 22).



Рис. 22. Операторы цикла в Scratch.

Выполнение проекта «Краб рисует лесенку» с использованием линейного и циклического алгоритмов.

Легко заметить, что на рис. 20 несколько раз повторяется один набор команд: идти 60 шагов, повернуть вверх (0 градусов), идти 40 шагов, повернуть направо. Применим цикл со счётчиком.

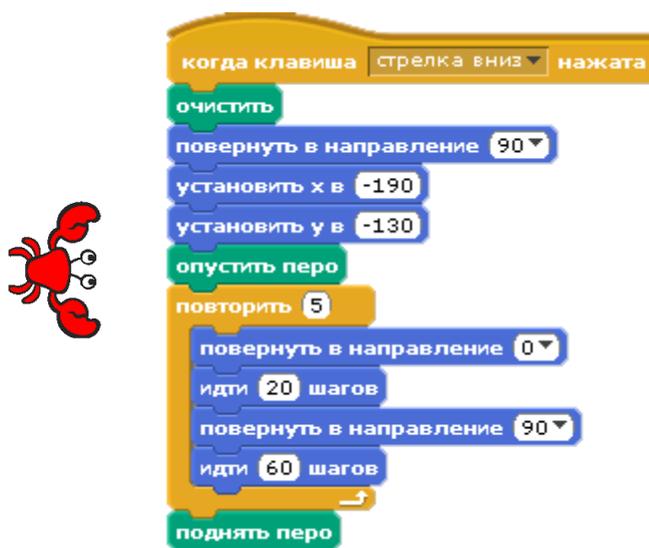


Рис. 23. Циклический алгоритм.

Обратите внимание: на рис. 23 текст скрипта стал короче, чем на рис. 20, а Краб нарисовал 5 ступенек лесенки (рис. 24). Теперь достаточно изменить значение в поле Повторить, и ступенек будет сколько угодно много. Изменим количество шагов (высоту ступеньки) только в одном блоке со значения 40 на 20. Длина всех ступенек изменится.

При использовании циклического алгоритма:

— процесс построения лесенки по-прежнему не виден;

- удалось реализовать требование о простом изменении количества и размера ступеней;
- требования по использованию условия с Рычажком реализовать по-прежнему невозможно.

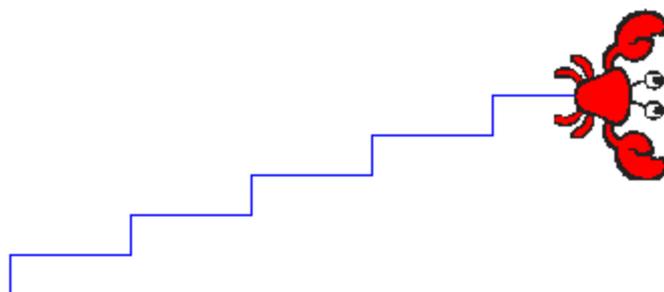


Рис. 24. Результат выполнения циклического скрипта для Спрайта 1 в костюме Краба на сцене с белым фоном (из коллекции *crab1-a*).

**Вывод 2.** При использовании только линейного и циклического алгоритмов нельзя выполнить все требования проекта «Краб рисует лесенку».

**Алгоритмы ветвления.** Ветвящийся или разветвляющийся алгоритм — это алгоритм, в котором имеется несколько альтернативных ветвей действий. Выбор называют простым, когда есть только два варианта, и сложным, когда вариантов больше двух (рис. 25).

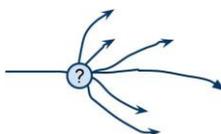


Рис. 25. Сложный выбор.

Момент выбора называется точкой ветвления. Ветвление — одна из трех (наряду с линейным выполнением команд и циклом) базовых конструкций алгоритмов. Все языки программирования имеют специальные операторы (команды) ветвления (условные операторы) программы в зависимости от того, выполнено указанное условие или нет. В *Scratch* в группе Контроль имеются 3 условных оператора: полное и неполное ветвление и пауза (рис. 26).



Рис. 26. Условные операторы в *Scratch*.

Выполнение проекта «Краб рисует лесенку» с использованием линейного, циклического и разветвляющегося алгоритмов.

Как видно на рис. 27, в готовый скрипт (рис. 23) были добавлены:

1) два блока ожидания , чтобы можно было видеть (как в мультфильме<sup>3</sup>) процесс рисования лесенки;

2) неполное ветвление , в условие проверки которого было введено значение Рычажка из подключенной ScratchDuino.Лаборатории. Согласно указанному условию, скрипт будет выполняться при условии, если Рычажок стоит на правой половине своей «дорожки», т.е. значение переменной Рычажок больше 50.

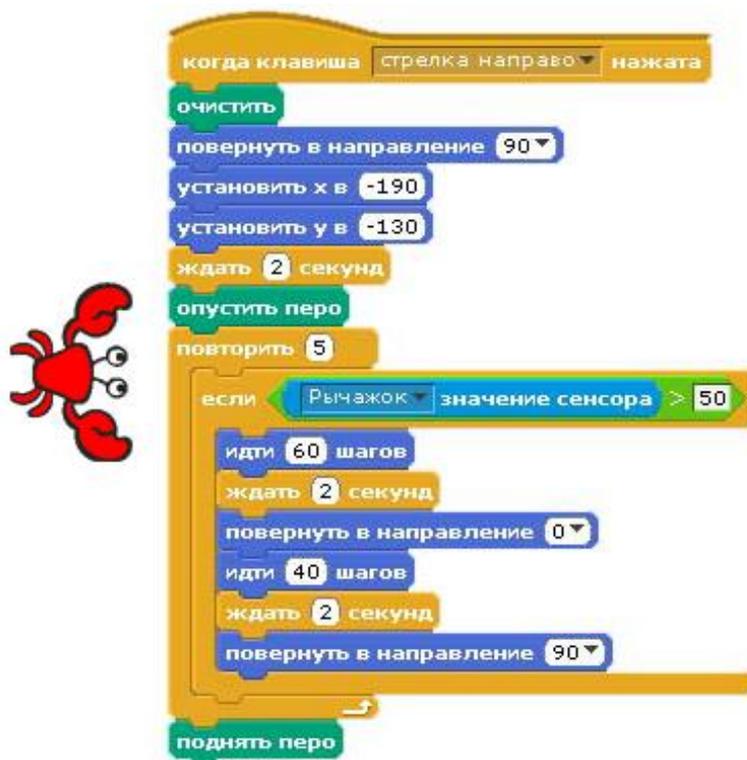


Рис. 27. Алгоритм с использованием ветвления.

**Вывод:** Проект может быть выполнен в полном объёме только при использовании всех трёх видов алгоритмов (линейного, циклического и разветвляющегося).

<sup>3</sup> Задержка изображения, удобная для человеческого глаза, входит в число методов разработки проектов с анимацией.

### Порядок « сборки » блока проверки условия

На рис. 27 виден конечный результат блока проверки условия. Его «сборка» происходит следующим образом: «перетягиваем» последовательно блоки неполного ветвления (группа Контроль); логического выражения (группа Операторы); значения сенсора (Группа Сенсоры) (рис. 28).



Рис. 28. Порядок « сборки » блока проверки условия.

Числовое значение можно менять в поле редактирования.

### Прием редактирования скрипта

Если нужно удалить блок из готового скрипта, используется правило «Отделяем снизу». Например, для удаления из скрипта (рис. 29) одного блока (выделен стрелкой) необходимо:

1. «Оторвать» часть, расположенную ниже.
2. «Оторвать» тем же приёмом лишний блок.
3. Вернуть «оторванную» часть на место снизу.



Рис. 29. Порядок удаления блока из скрипта.

## Проекты применения элементов платы ScratchDuino.Лаборатория

### Кнопки

ScratchDuino.Лаборатория позволяет управлять виртуальными Исполнителями. Рассмотрим применение кнопок (переменные «Соппротивление А», «Соппротивление В», «Соппротивление С», «Соппротивление D»).

*Тема проекта:* «Путешествие Кота».

*Условия выполнения проекта:*

- для «путешествия» Кота подобрана сюжетная сцена;
- при нажатии цветных кнопок на ScratchDuino.Лаборатории Кот должен двигаться влево, вправо, вверх и вниз;
- остановка скрипта происходит при нажатии на клавиатуре клавиши пробела.

### Выполнение проекта

Щёлкнуть по исполнителю Сцена, выбрать вкладку Фон. Выбрать Новый фон, кнопка Импорт. Откроется окно с готовыми изображениями. Выбрать папку Nature. В папке Nature выбрать файл desert.

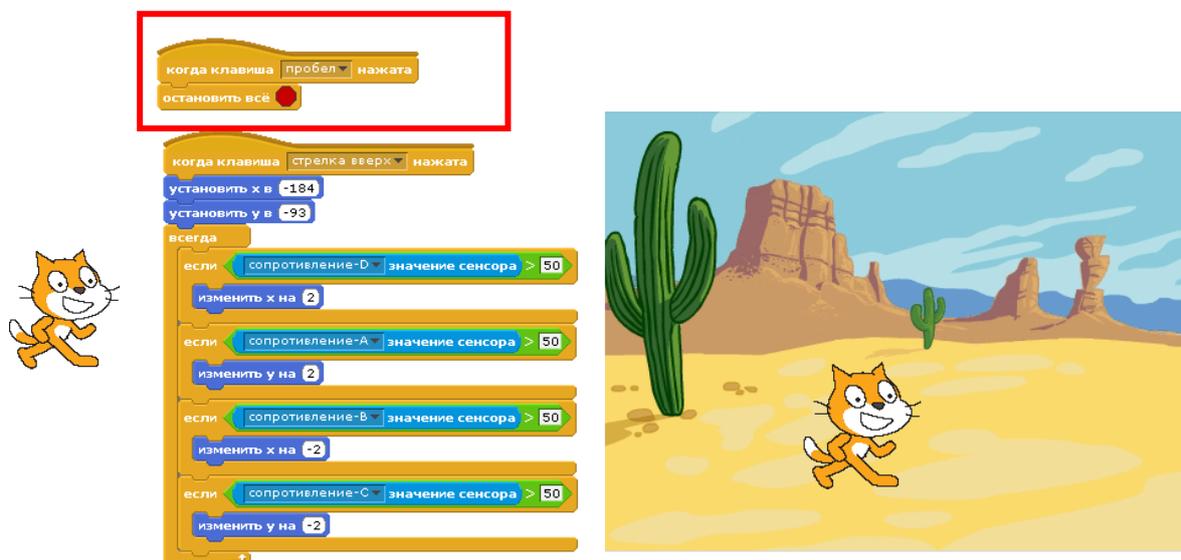


Рис. 30. Пример двух скриптов для управления спрайтом в костюме Котёнка (файл cat1-a) с помощью клавиш ScratchDuino.Лаборатории. Выделен первый скрипт, который приводит к остановке всех скриптов при нажатии клавиши пробела на клавиатуре.

## Рычажок

Тема проекта: «Задаем скорость мячику Рычажком».

Условия выполнения проекта:

- исполнитель — Мяч;
- Мяч при запуске скрипта начинает движение «Туда-сюда» по горизонтали, отталкиваясь от стенок в пределах сцены;
- скорость движения Мяча задается с помощью Рычажка;
- остановка скрипта происходит при нажатии на клавиатуре клавиши пробела.

Выполнение проекта

Ключевой идеей решения является изменение шага изменения координаты по оси x. Эффект отталкивания от края сцены реализован с помощью проверки текущей координаты Мяча (положение x) и координаты левой и правой границ Сцены: 200 и -200.

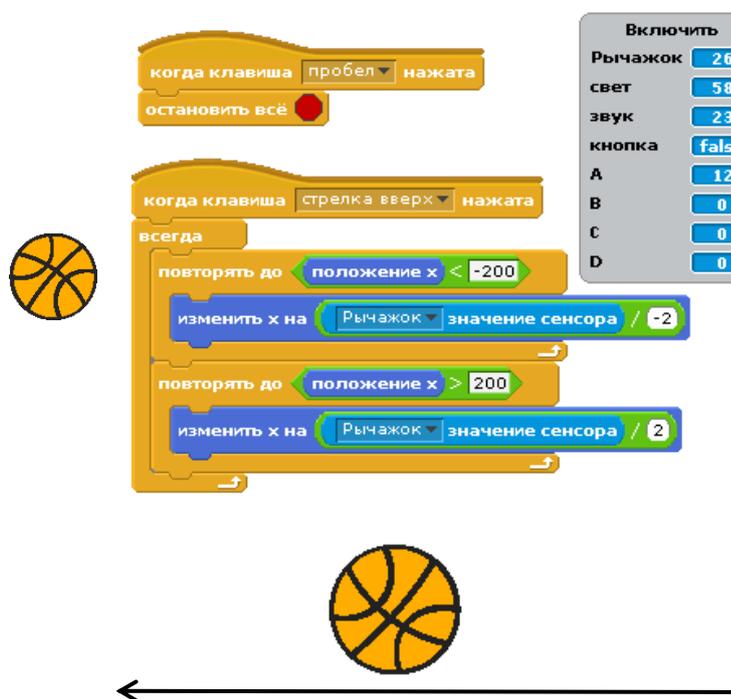


Рис. 31. Скрипт и результат его выполнения.

## Звуковой сенсор

Тема проекта: «Прыжок по команде (хлопку или слову)».

Условия выполнения проекта:

- исполнитель — Гимнастка;
- после запуска скрипта видно Гимнастку, готовую к прыжку;
- если громко крикнуть (или хлопнуть в ладоши), то Гимнастка подпрыгивает и через 3 секунды возвращается в исходное положение;
- если уровень шума в помещении постоянно выше значения 24 (например, включена музыка), то Гимнастка через 3 секунды всё равно вернется в исходное положение (отдохнёт) и снова подпрыгнет;
- остановка скрипта происходит при нажатии клавиши пробела.

Выполнение проекта

Зная, при каких значениях датчик звука реагирует на громкий звук (хлопок), можно подобрать значение, которое мы впишем в поле проверки условия. Решение построено на смене костюма для одного спрайта в результате проверки условия. На рис. 32 показано, что при смене значений датчика (больше 24) происходит смена костюма.

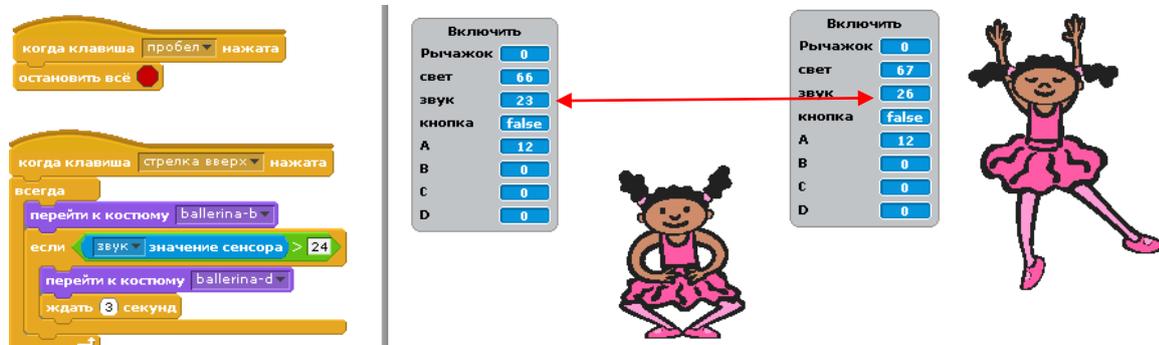


Рис. 32. Скрипт и результат его выполнения.

Эффект прыжка достигается сменой центра второго костюма (рис. 33).

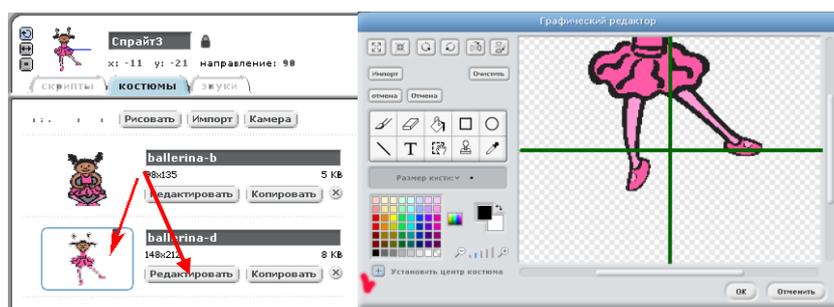


Рис. 33. Местоположение нового центра костюма ballerina-d.

Тема проекта: «Амплитуду на экран!»

Условия выполнения проекта:

- два исполнителя — Самописец и Летучая мышь;
- Самописец рисует красную линию — амплитуду звука;
- Летучая мышь находится в произвольном месте, «слушает» звуки и «запоминает» максимальную величину переменной Звук, показывая ее в облачке (рис. 34);
- величина шага меняется с помощью Рычажка;
- при нажатой клавише пробела останавливается исполнение всех скриптов, удаляется нарисованное изображение и Самописец устанавливается в начальное положение (нижний левый угол сцены).

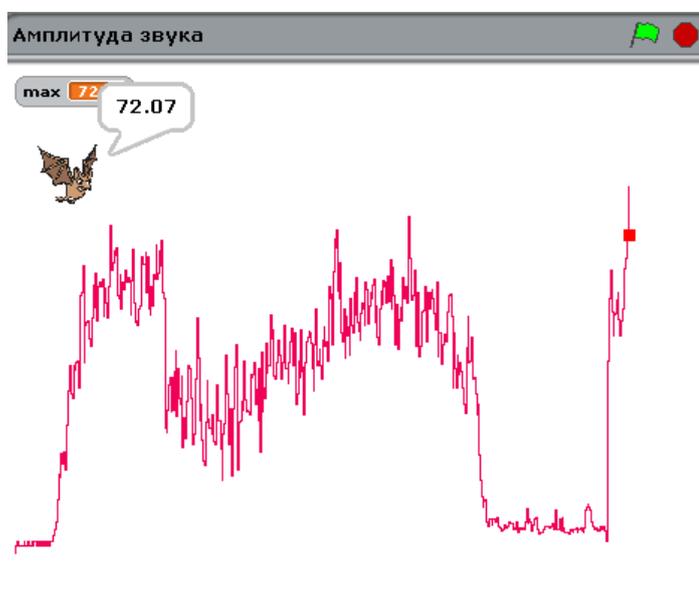


Рис. 34. Вид амплитуды во время звучания композиции *Ohne dich* (Rammstein). Максимальная амплитуда — 72.

Выполнение проекта

Создать два спрайта (рис. 35):

- красный квадрат — Самописец нарисовать самостоятельно;
- Летучую мышь взять из коллекции.

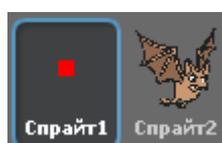


Рис. 35. Спрайты проекта «Амплитуду на экран!»

Согласно исследованиям звукового датчика (см. первую главу), его минимальное значение в тишине — 23. Поэтому необходимо отнять это число, чтобы «сдвинуть» амплитуду на поле в ноль. Диапазон чувствительности датчика — от 23 до 93, т.е. примерно 70 единиц, а сцена, где отображается амплитуда, имеет высоту 360 пикселей. Разделим 360 на 70, и получим величину, на которую нужно умножать, чтобы рисунок амплитуды вписывался в сцену. «-150» — это значение, на которое надо переместить рисунок по оси  $y$  вниз, т.к. центр координат спрайта (точка с координатами (0;0)) находится в центре сцены. Рычажок имеет диапазон от 0 до 100. Деление на 200 позволяет регулировать величину шага от 0 до 0,5.

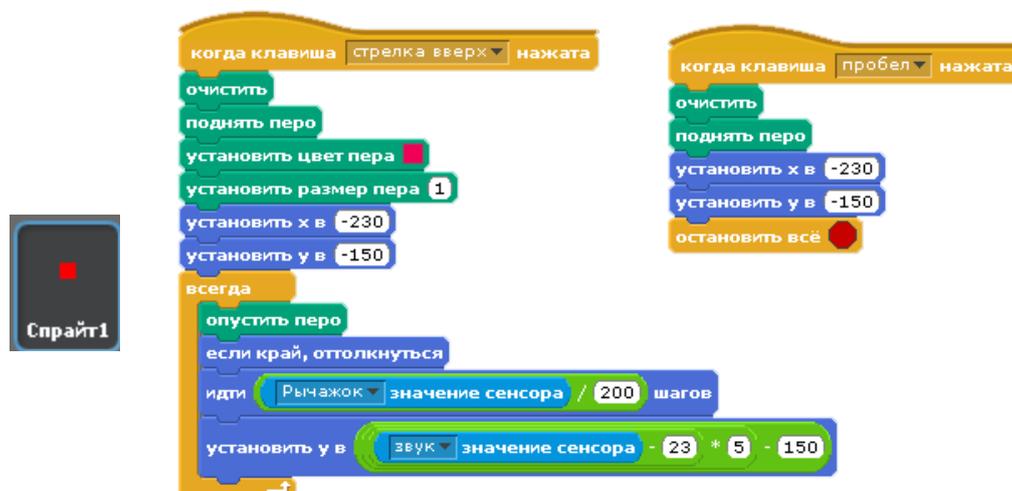


Рис. 36. Два скрипта для Спрайта 1: первый «рисует», второй — при нажатой клавише пробела «стирает» нарисованное и устанавливает «самописец» в начальное положение.

Задание для Летучей мыши требует использования переменной  $max$ <sup>4</sup>. До цикла ей присваивается нулевое значение. Если в цикле значение звукового датчика превышает нулевое значение, то значение переменной  $max$  обновляется и выводится на экран на 1 секунду.

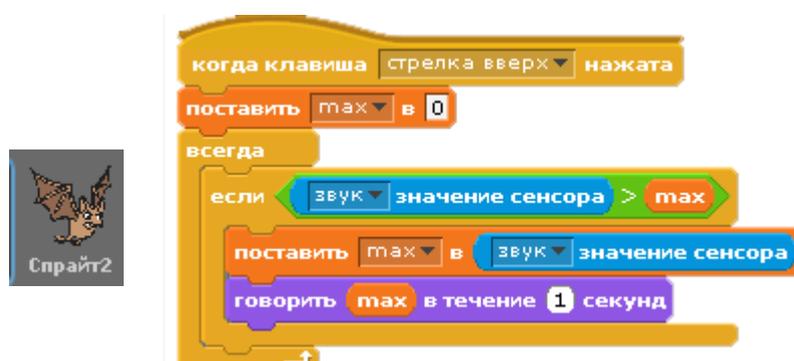


Рис. 37. Скрипт для Спрайта 2.

<sup>4</sup> Создание переменных происходит в группе Переменные.

## Выносной температурный датчик

Тема проекта: «Домашний термометр».

Условия выполнения проекта:

- два исполнителя — Шкала и Красная стрелка;
- Шкала комнатного термометра расположена слева;
- Красная стрелка двигается вдоль Шкалы, и при этом слева отображается в окне вывода числовое значение температуры (рис. 39);
- следует выяснить величину одного градуса в пикселях на нарисованной Шкале.
- по нажатию клавиши пробела исполнение всех скриптов останавливается, Шкала устанавливается в начальное положение (нижний левый угол сцены).

Выполнение проекта

Оба спрайта — температурную Шкалу от 0 до +50 и Красную стрелку — нарисовать самостоятельно (рис. 38).



Рис. 38. Спрайты проекта.

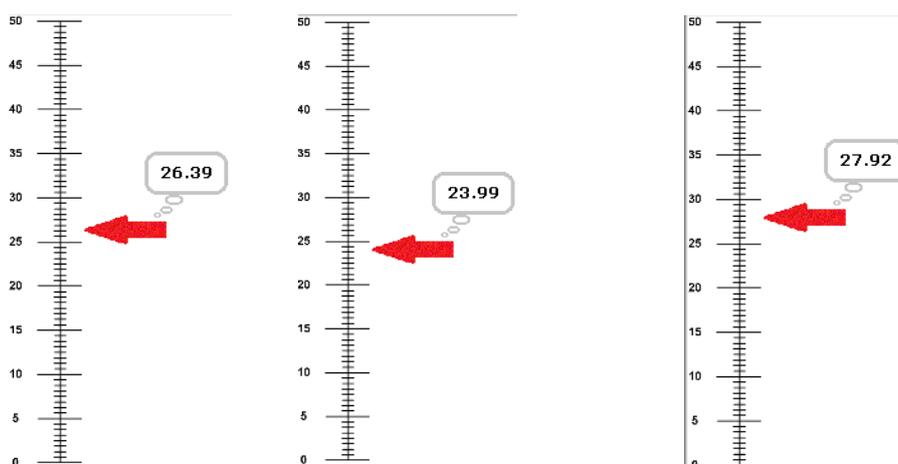


Рис. 39. Три вида экрана, отражающие результат выполнения проекта. Видно динамическое изменение значения температуры измеряемой среды.

Для Спрайта 2 (Шкала) напишем два скрипта: стандартная остановка по нажатию клавиши пробела, а при запуске проекта (клавиша вверх) установка шкалы в нижний левый угол сцены (рис. 40).



Рис. 40. Скрипт для спрайта Шкала.

Пояснения к формуле для Красной стрелки.

1. Выясним значение в пикселях одного градуса нарисованной шкалы. Для этого измерим в пикселях нарисованную шкалу и разделим эту величину на 50. Установим указатель манипулятора мыши на нулевое значение нарисованной шкалы (нижний левый угол). На информационной панели заметим числовое значение. В нашем случае это «-175». При наведении на максимальное значение нарисованной шкалы замечаем координаты **175**. Таким образом, величина движения на один градус вычисляется по формуле  $(175+175)/50=7$ . Это цена одного деления нарисованной шкалы в пикселях в установке координат Красной стрелки на рис. 39.

2. Используем формулу 2 зависимости показаний датчика от показаний термометра, выведенную в первой главе:  $x = 2,2237y + 0,9261$ . Используем эти коэффициенты в установке координат Красной стрелки (рис. 41).



Рис. 41. Скрипт для спрайта Красная стрелка.

**Примечание.** Калибровка датчика производилась бытовым термометром, точность которого вызывает сомнение, поэтому при выполнении проекта стало заметно, что датчик показывает температуру выше, чем подсказывают собственные ощущения. Необходимо провести калибровку своего температурного датчика с использованием более точного термометра.

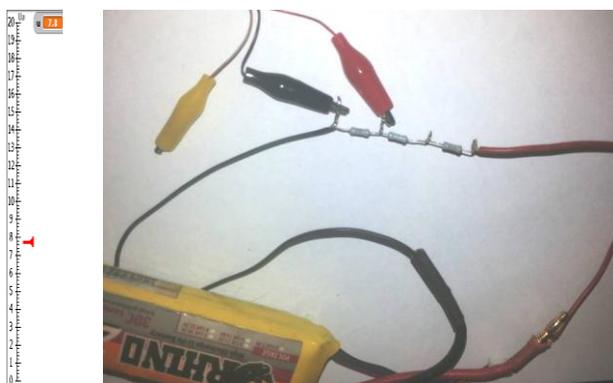
## Внешние датчики

*Тема проекта:* «Вольтметр для измерения напряжения постоянного тока в диапазоне от 4 до 20 В».

*Условия выполнения проекта:*

- собрать из трех резисторов по 91 кОм устройство «Делитель напряжения» (рис. 42 и 43);
- провести калибровку устройства «Делитель напряжения» при помощи цифрового вольтметра;
- использовать в скрипте два исполнителя: Шкала вольтметра, Красная стрелка;
- Шкала вольтметра расположена слева;
- Красная стрелка двигается вдоль Шкалы вольтметра;
- измеряемое числовое значение напряжения отображается в окне вывода переменной (рис. 42);
- при нажатии клавиши пробела исполнение всех скриптов останавливается.

*Выполнение проекта*



*Рис. 42. Измерение напряжения на аккумуляторе с помощью Устройства и вид экрана.*

В комплекте ScratchDuino.Лаборатории имеется соединительный шнур с тремя зажимами — «крокодилами». Чёрный зажим — «минус», красный — «аналоговый вход», жёлтый — питание для датчика (4,36 В). В этом можно убедиться, проверив с помощью вольтметра.

Использовать чёрный и красный зажимы. Для диапазона от 4 до 20 В изготовить устройство «Делитель напряжения». Сопротивление вольтметра должно быть достаточно высоким. Чтобы не использовать сложные

расчёты и не определять внутреннее сопротивление аналогового входа, откалибровать устройство опытным путем. Взять три резистора по 91 кОм. На рис. 43 видно, что фактически измеряется падение напряжения на одном резисторе. Провести калибровку устройства «Делитель напряжения».



Рис. 43. Вид устройства «Делитель напряжения», собранного из трёх резисторов по 91 кОм и присоединённых «крокодилов».

Таблица 4. Связь показаний значения сенсора ScratchDuino.Лаборатории и цифрового вольтметра

Напряжение, подаваемое на Устройство и измеренное цифровым вольтметром	Показание сенсора	Коэффициент
3,94	18,57	4,71
4,95	23,4	4,73
5,9	28	4,75
6,97	33	4,73
8,03	38,03	4,74
9	42,72	4,75
10,02	47,51	4,74
11,08	52,49	4,74
12	56,9	4,74
13,02	61,68	4,74
14,03	66,57	4,74
15,77	74,7	4,74
	<b>СРЕДНЕЕ</b>	<b>4,74</b>

Как видно из таблицы 4, зависимость является линейной. Среднее значение коэффициента — **4,74**.

Оба спрайта — Шкалу вольтметра от 0 до 20 В и Красную стрелку — нарисовать самостоятельно (рис. 44).



Рис. 44. Спрайты проекта.

Для спрайта Шкала вольтметра написать скрипт, согласно которому при запуске проекта (нажатой на клавиатуре компьютера клавиши «Вверх») происходит установка Шкалы в нижний левый угол сцены (рис. 45).

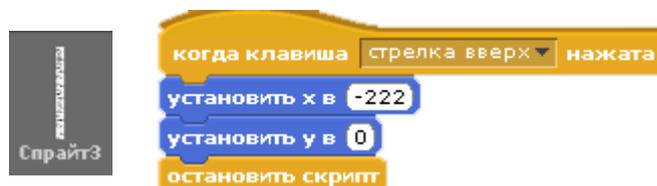


Рис. 45. Скрипт для спрайта Шкала.

Пояснение к формуле в скрипте для спрайта Красная стрелка:

- установить стрелку ближе к рисунку шкалы;
- использовать переменную  $u$  для измеряемого напряжения;
- установить стрелку по оси  $y$  в соответствии со значением переменной;
- отобразить переменную.



Рис. 46. Скрипт для спрайта Красная стрелка.

Провести тестирование устройства «Делитель напряжения». Измерить напряжение аккумулятора, как показано на рис. 42. На экране видно динамичное изменение показаний.

**Примечание.** С помощью этого Устройства можно измерять напряжение от 0 до 4 вольт, однако точность измерений будет невысока. В то же время измерение напряжения от 0 до 4 вольт не вызывает сложностей, т.к. не требует делителей: достаточно подключить источник питания до 4 В к чёрному и красному зажимам. Важно провести калибровку по описанной выше методике и написать отдельный скрипт.

## Развиваем сообщество единомышленников

Каждый счастливый обладатель конструкторов проекта ScratchDuino, будь то педагог или родители, спустя какое-то время испытывают лёгкое разочарование: а что дальше? Как правило, свои идеи создания новых проектов быстро заканчиваются. Однако если есть место, где можно показать свои проекты, получить на них отзыв, посмотреть готовые проекты других коллег, то творческие идеи получают новый импульс. Для этого и создаются сетевые сообщества, больше напоминающие клубы единомышленников, где нет различий по возрасту и уровню подготовки. В проекте ScratchDuino местом такого общения является wiki-портал [wiki.scratchduino.ru](http://wiki.scratchduino.ru). В этой главе будут рассмотрены рекомендации для новых участников этого сообщества.

### Регистрация на wiki-портале проекта ScratchDuino

1. Пройдём по адресу <http://wiki.scratchduino.ru> и найдём в правом верхнем углу ссылку Представиться/зарегистрироваться (рис. 47).

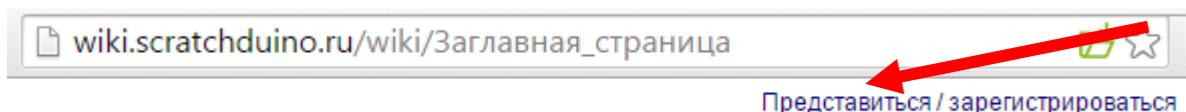


Рис. 47. Начало регистрации на портале <http://wiki.scratchduino.ru>.

2. Откроется окно авторизации, в котором вы в дальнейшем будете вводить свои логин и пароль. В первый раз выбираем ссылку Создать учётную запись (рис. 48).

### Представиться или зарегистрироваться

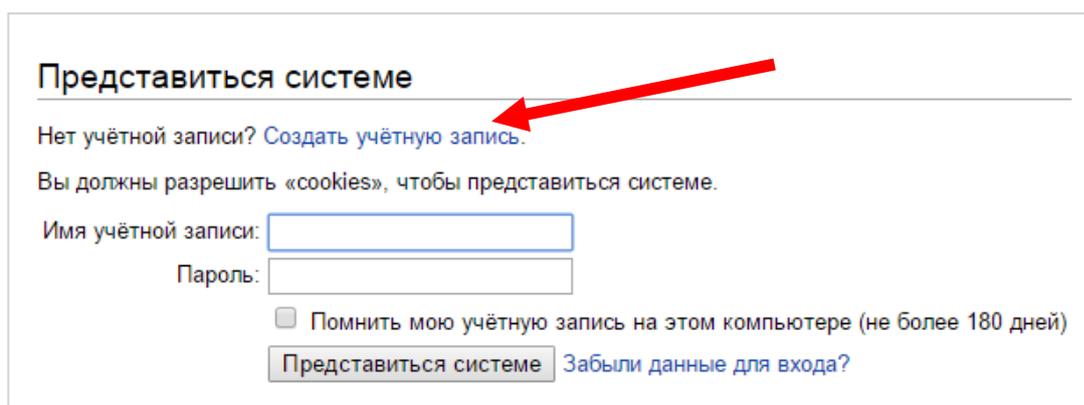
A screenshot of a login/registration page titled 'Представиться системе'. It contains the text 'Нет учётной записи? Создать учётную запись.' with a red arrow pointing to the link 'Создать учётную запись.'. Below this is a message: 'Вы должны разрешить «cookies», чтобы представиться системе.' There are two input fields: 'Имя учётной записи:' and 'Пароль:'. Below the password field is a checkbox labeled 'Помнить мою учётную запись на этом компьютере (не более 180 дней)'. At the bottom, there are two buttons: 'Представиться системе' and 'Забыли данные для входа?'.

Рис. 48. Страница авторизации (она же позволяет перейти на страницу создания учётной записи).

3. Заполняем данные о себе. Обратите внимание, что имя учётной записи будет вашей визитной карточкой на портале. Поэтому выбирайте вариант, по которому вас легко можно узнать. Например, если ваше имя Иванов Иван Иванович, то можно использовать такие варианты Имени учётной записи: Иванов Иван Иванович, Иванов Иван, Ivanov, Ivanov Ivan и т.д. Избегайте анонимных имён. Нажимаем кнопку Создать учётную запись.

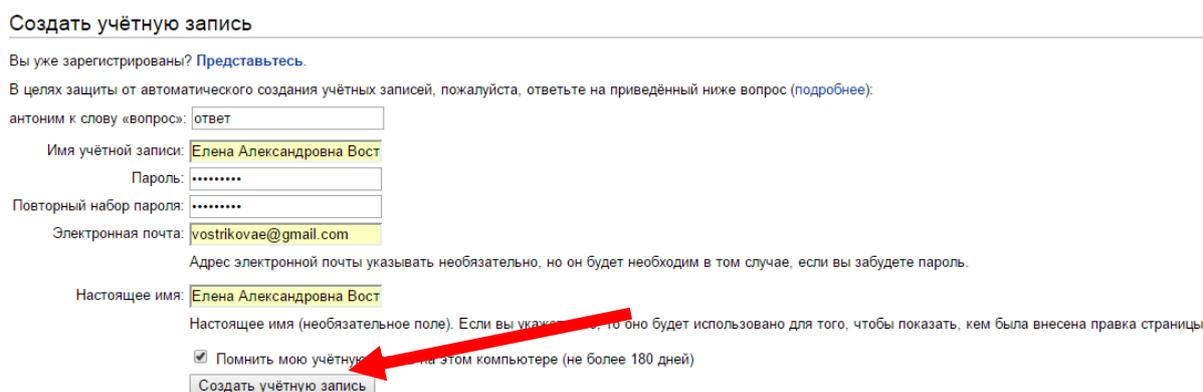


Рис. 49. Анкета создания учетной записи.

4. Если все поля заполнены корректно, то появится вот такая информация (рис. 50):

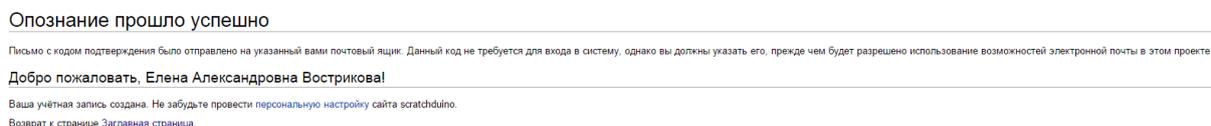


Рис. 50. Уведомление об успешном создании учётной записи.

5. Проверьте входящие письма и убедитесь, что письмо не попало в спам. Ниже приведен примерный текст письма, которое приходит на вашу почту. Щёлкните по ссылке в письме (рис. 51 и 52).

MediaWiki Mail                      scratchduino:Запрос на подтверждение адреса электронной почты - Кто-т

Рис. 51. Тема письма в папке Входящие в вашем почтовом ящике.

Кто-то (возможно, вы) с IP-адресом 94.199.78.161 зарегистрировал на сервере проекта scratchduino учётную запись «Елена Александровна Вострикова», указав этот адрес электронной почты.

Чтобы подтвердить, что эта учётная запись действительно принадлежит вам и включить возможность отправки электронной почты с сайта scratchduino, откройте приведённую ниже ссылку в браузере:

<http://wiki.scratchduino.ru/wiki/Special:ConfirmEmail/bee4a6bfc7fd22952adf333dd5217556>

Рис. 52. Текст письма в вашем почтовом ящике.

6. Щёлкнув по ссылке в письме, вы перейдёте на сайт проекта и попадёте на страницу с надписью, показанной на рис. 53:

### Подтверждение адреса электронной почты

Ваш адрес электронной почты подтверждён.

*Рис. 53. Страница портала после подтверждения почтового ящика.*

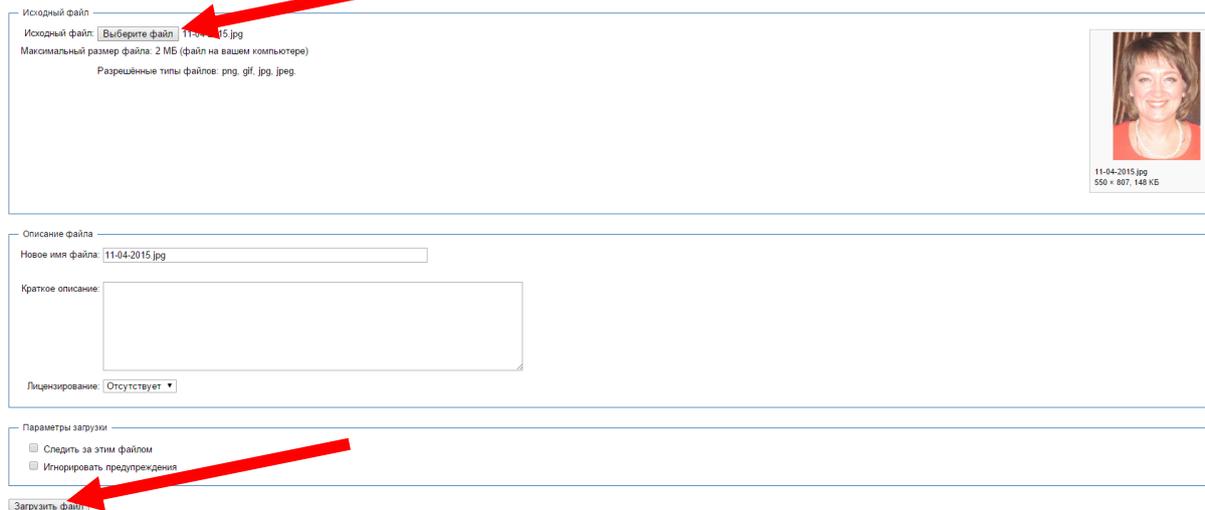
7. Приступим к созданию личной странички, где вы сможете выкладывать свои творческие проекты для обмена и обсуждения. Многие участники Wiki-портала ScratchDuino никогда не встречались лично, поэтому важно разместить на странице свою фотографию. Рекомендуется выбрать ту, на которой вы в хорошем настроении. Ведь создание проектов для роботов — это приятное занятие! Величина файла фотографии не должна превышать 2 МБ. Разрешённые типы файлов: png, gif, jpg, jpeg.

8. Выберите в левом меню портала ссылку Загрузить файл. В открывшейся странице найдите и нажмите кнопку Выберите файл. На своем компьютере выберите фото. Выбранный вами файл отобразится на странице. Но он ещё не загружен! В нижней части страницы найдите кнопку Загрузить файл. Нажмите её.

#### Загрузить файл

Воспользуйтесь этой формой для загрузки файлов на сервер. Чтобы просмотреть ранее загруженные файлы, обратитесь к [списку загруженных файлов](#). Загрузка файлов также записывается в [журнал загрузок](#), данные об удалённых файлах можно найти в [журнале удалений](#). Для включения файла в статью вы можете использовать строки вида:

- `[[файл:Ф11е. jpg]]` для вставки полной версии файла.
- `[[файл:Ф11е. png|200px|thumb|left|описание]]` для вставки слева от текста уменьшенной копии (слева по ширине верши файла с выводом под ним указанного описания).
- `[[media:Ф11е. ogg]]` для вставки ссылки на файл, без отображения его содержимого на странице.



*Рис. 54. Кнопки Выбрать файл и Загрузить файл.*



*Рис. 55.*

*Имя страницы, где расположена ваша фотография*

9. В результате страница с загруженным файлом будет выглядеть, как показано на рис. 55. Скопируйте имя страницы. В данном примере это Файл:11-04-2015.jpg.

10. Перейдём к непосредственному созданию страницы участника сообщества ScratchDuino на портале <http://wiki.scratchduino.ru>. Обратите внимание, что имя вашей учётной записи выделено в верхней части страницы красным цветом (рис. 56). Это значит, что страница названа, но не создана! Щёлкнем по имени учётной записи.



*Рис. 56. Ваша личная страница названа, но не создана.*

11. Вы перейдете на страницу с сообщением, которое показано на рис. 57.

## Участник: Елена Александровна Вострикова

В настоящий момент текст на данной странице отсутствует. Вы можете [найти упоминание данного названия](#) в других статьях, [найти соответствующие записи журналов](#) или [создать страницу с таким названием](#).

*Рис. 57. Страница, где вам предлагают создать личную страницу.*

Выберите ссылку на фразу «создать страницу с таким названием».

12. Перед вами откроется окно текстового редактора.

## Редактирование Участник:Елена Александровна Вострикова



{{subst:Шаблон:Личная страница участника сообщества ScratchDuino}}

*Рис. 58. Ваша страница в режиме редактирования.*

Для участников сообщества создан шаблон личной страницы. Для создания страницы достаточно внимательно (без ошибок) напечатать фразу {{subst:Шаблон:Личная страница участника сообщества ScratchDuino}}

и нажать кнопку Записать страницу. В результате будет создана ваша страничка с шаблонным содержанием (рис. 59).

## Участник: Елена Александровна Вострикова



**Город:** *Здесь пишем свой регион и город (поселок)*

**Образовательное учреждение:** *Здесь пишем название своей школы, учрежде*

**Должность:** *для взрослых участников сообщества здесь пишем свою должно*

**Мой вклад в развитие проекта ScratchDuino:**

*Здесь размещаем ссылки на страницы с портфолио ваших творческих проек.*  
Например, так

- **Проект ScratchDuino** *Лаборатория Морской бой* (замените этот пример на

**Мои обсуждения творческих проектов других участников сообщества ScratchDuino**

*Здесь размещаем ссылки на страницы с портфолио творческих проектов других участников соот*  
*которые вам понравились или те проекты, которые вы бы хотели улучшить*

Категория: Участники сообщества ScratchDuino

Рис. 59. Ваша страничка с шаблонным содержанием.

13. Перейдите на вкладку Править. Заполните обязательные строчки Личной страницы участника сообщества ScratchDuino. Замените имя файла логотипа Logo-1024be1-300x54.png на имя файла вашей личной фотографии. Вместо фразы Участник сообщества ScratchDuino напишите свои имя, фамилию и т.д.

Например, так может быть заполнена личная страница в режиме Править:

```
[[Изображение:11-04-2015.jpg|thumb|100px|left|Вострикова
Елена Александровна]]
""Город:"" Новокузнецк
""Образовательное учреждение:"" [http://ipknk.ru MAOU ДПО Ин-
ститут повышения квалификации Новокузнецк]
""Должность:""
[https://sites.google.com/site/otdelrazvitiiaobrazovania/ начальник
отдела развития образования], [http://vostrikova.blogspot.com
Мой блог]
""Мой вклад в развитие проекта ScratchDuino:""
*[https://goo.gl/Lbzskk Методические рекомендации для прове-
дения мастер-класса]
*[[Проект ScratchDuino.Лаборатория Морской бой]]
```

""Мои обсуждения творческих проектов других участников сообщества ScratchDuino""

Пока не принимала участие в обсуждениях. Жду появления новых проектов

[[Категория: Участники сообщества ScratchDuino]]

14. Нажмите кнопку Записать страницу.

В целях защиты система просит ввести кодовое слово (рис. 60). Введите слово и снова нажмите кнопку Записать страницу.

Ваша правка содержит новые внешние ссылки. В целях защиты от автоматического спама, пожалуйста, ответьте на приведённый ниже вопрос ([подробнее](#)):  
антоним к слову «вопрос»:

*Рис. 60. Проверка, что с порталом работает человек.*

После внесения изменений шаблонная страничка станет вашей!

## Участник: Елена Александровна Вострикова



Вострикова  
Елена  
Александровна

**Город:** Новокузнецк

**Образовательное учреждение:** [МАОУ ДПО Институт повышения квалификации Новокузнецк](#)

**Должность:** начальник отдела развития образования [🔒](#), [Мой блог](#)

**Мой вклад в развитие проекта ScratchDuino:**

- Методические рекомендации для проведения мастер-класса [🔒](#)
- Проект ScratchDuino [Лаборатория Морской бой](#)

**Мои обсуждения творческих проектов других участников сообщества ScratchDuino**

Пока не принимала участие в обсуждениях. Жду появления новых проектов

Категория: [Участники сообщества ScratchDuino](#)

*Рис. 61. Ваша страничка с заполненным содержанием.*

Щёлкните по названию категории Участники сообщества ScratchDuino, и вы перейдёте на страницу с именами всех, кто создал свои страницы и готовится (или уже готов) к размещению своих творческих проектов. Можете изучать страницы участников сообщества и принимать участие в обсуждениях готовых творческих проектов.

## Размещение фестивальной работы на [wiki.scratchduino.ru](http://wiki.scratchduino.ru)

Если вы создали проект, который восхищает вас самих, и вы готовы поделиться им с единомышленниками, то отправляйте свою работу на ежегодный фестиваль по свободной робототехнике ScratchDuino.

На своей личной странице в режиме Править добавьте имя страницы, где будет размещён ваш проект. Для этого нужно написать строчку по следующим правилам:

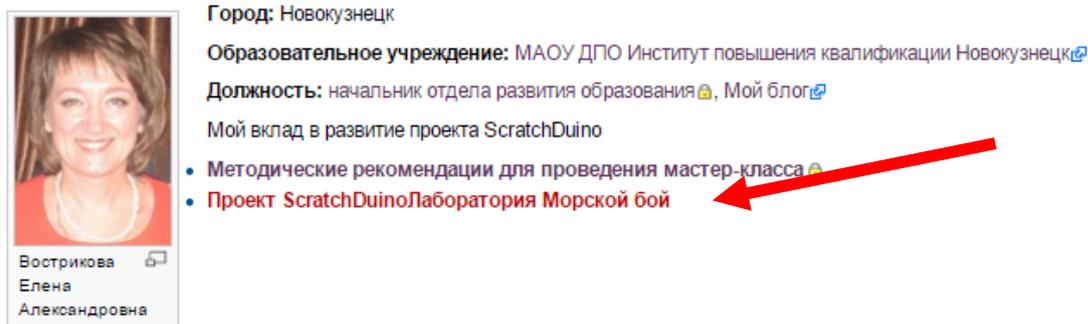
1. Начинаться имя страницы должно со слова Проект.
2. После пробела указываете, в какой номинации создан ваш проект.
3. Название вашего проекта должно быть оригинальным и не повторять названия других проектов. Для этого зайдите в [[Категория: Проекты]] и убедитесь, что ваше название не использовали другие.

Пример такой строчки приведён ниже.

\* ""[[Проект ScratchDuino.Лаборатория Морской бой]]""

Нажмите кнопку Записать страницу!

### Участник: Елена Александровна Вострикова



Город: Новокузнецк  
Образовательное учреждение: MAOU ДПО Институт повышения квалификации Новокузнецк  
Должность: начальник отдела развития образования, Мой блог  
Мой вклад в развитие проекта ScratchDuino

- Методические рекомендации для проведения мастер-класса
- Проект ScratchDuino Лаборатория Морской бой

Вострикова  
Елена  
Александровна

Рис. 62. Страница портфолио проекта названа, но ещё не создана.

Щёлкнем по новой ссылке и перейдём в режим редактирования новой страницы.

### Редактирование Проект ScratchDuino Лаборатория Морской бой

Вы перешли по ссылке на страницу, которой пока не существует. Чтобы её создать, наберите текст в окне, расположенном ниже (подроб-



Рис. 63. Режим редактирования новой страницы.

Для участников фестиваля создан шаблон портфолио фестивального проекта. Для создания страницы достаточно будет внимательно (без ошибок) напечатать фразу `{{subst:Шаблон:Описание фестивальной работы ScratchDuino 2016}}` и нажать кнопку Записать страницу.

В результате будет развёрнуто портфолио фестивального проекта.

## Проект ScratchDuinoЛаборатория Морской бой

**Содержание** [убрать]

- 1 `Имя автора (авторов) проекта и ссылка(и) на личную (е) странички на http://wiki.scratchduino.ru/`
- 2 `Категория участника`
- 3 `Номинация, в которую подается фестивальный проект`
- 4 `Описание фестивального проекта`
- 5 `Ссылка на скачивание фестивального проекта`

**Имя автора (авторов) проекта и ссылка(и) на личную (е) странички на <http://wiki.scratchduino.ru/>** [править]

На личной страничке должны быть указаны:

- Фамилия Имя полностью
- фото желательно
- ссылки на все разменные фестивальные проекты
- ссылки на все фестивальные проекты, в обсуждении которых принимал(а) участие

**Категория участника** [править]

- дошкольное образование
- начальные классы школы (7-10 лет)
- дополнительное образование (7-10 лет)
- семейное образование (7-10 лет)
- основная школа (11-15 лет)
- дополнительное образование (11-15 лет)
- семейное образование (11-15 лет)
- старшая школа (16-17 лет)
- студенты высшая школа (18-23)
- педагоги школы
- педагоги дополнительного образования
- участник вне категории

**Номинация, в которую подается фестивальный проект** [править]

- ScratchDuinoЛаборатория
- ScratchDuinoРоботоплатформа
- ScratchDuinoЛаборатория (модификация)
- ScratchDuinoРоботоплатформа (модификация)

**Описание фестивального проекта** [править]

Что должны знать участники, которые хотят провести испытание вашего проекта

**Ссылка на скачивание фестивального проекта** [править]

Активная ссылка на файл в формате sb

Категории: [Проекты](#) | [ScratchDuino 2016](#)

*Рис. 64. Шаблонная страница портфолио фестивального проекта.*

Далее важно заполнить все разделы портфолио. Ниже приведён текст заполненного портфолио в режиме редактирования страницы.

```
==Имя автора (авторов) проекта и ссылка(и) на личную (е) странички на http://wiki.scratchduino.ru/==
[[Участник:Елена Александровна Вострикова]]
==Категория участника ==
```

Участник вне категории

==Номинация, в которую подается фестивальный проект==

ScratchDuino.Лаборатория

==Описание фестивального проекта==

Когда-то в эпоху до появления компьютеров и интернета дети играли на игровых автоматах Морской бой. Сейчас можно поиграть в ее онлайн-версию по этому адресу <http://morskoy-boy.15kop.ru/game/>.

Мы создали подобие такой игры. В верхней части экрана плывут кораблики, в нижней части экрана — пушка. Движение пушкой выполняем с помощью левой и правой кнопок на ScratchDuino.Лаборатории. При нажатии на верхнюю кнопку пушка стреляет. В игре возникает пауза. При нажатии на нижнюю кнопку игра продолжается.

В игре есть несколько уровней. С каждым уровнем скорость корабликов увеличивается. При промахах уменьшаются жизни. На фоне игры появляются морские обитатели.

==Ссылка на скачивание фестивального проекта==

[<https://goo.gl/U5RCYx> Активная ссылка на скачивание файла в формате sb]

[[Категория: Проекты]]

[[Категория: ScratchDuino 2016]]

Нажмите кнопку Записать страницу!

Теперь ваш проект доступен всем участникам сообщества. Отзывы можно написать и прочитать на странице проекта во вкладке Обсуждения.

## Правила обсуждения фестивальных проектов на портале [wiki.scratchduino.ru](http://wiki.scratchduino.ru)

Фестиваль — это не конкурс, а активный обмен идеями, находками и готовыми проектами, и постоянный поиск наилучших вариантов применения свободной робототехники. Поэтому наряду с количеством и качеством размещённых проектов Экспертная группа при подведении итогов фестиваля будет принимать во внимание количество и качество проведенных участниками обсуждений.

Обсуждение — это всегда диалог: один спрашивает, другой отвечает. Если на вкладке Обсуждение (рис. 65) вам заданы вопросы, сделаны замечания и предложены идеи для улучшения вашего проекта, то считается хорошим тоном поблагодарить за вопрос и дать обстоятельный ответ, согласиться или не согласиться с замечаниями, рассмотреть и принять (не принять) предложения.

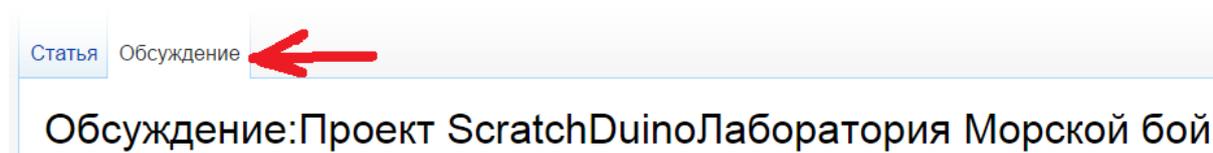


Рис. 65. Расположение вкладки Обсуждение  
на странице портфолио проекта.

Чтобы обсуждение не свелось к банальным смайликам или коротким фразам: «Прикольно!», «Класс!», предлагаем использовать в обсуждениях проектов стратегию «3-2-1». Это значит, что вы задаёте автору ТРИ вопроса, делаете ДВА замечания и вносите ОДНО предложение. Пример применения такой стратегии показан на рис. 67. В конце своего сообщения нажмите кнопку «Ваша подпись и момент времени». Появится вот такой набор символов: --~~~~. Нажмите кнопку Записать страницу.

Обсуждение проекта совершенно незнакомых людей требует вежливости. Начинайте любое обсуждение с приветствия:

*Здравствуйте! Добрый день! Приветствую Вас!* и т.д.

В тексте желательно использовать вежливые обороты. Вместо резкой фразы «Проект — отстой!» напишите так: «Идея проекта требует доработки».

Сразу научиться разрабатывать оригинальные проекты трудно, и важно поддержать новичков. Доброе слово окрыляет!

### Редактирование Обсуждение:Проект ScratchDuinoЛаборатория Морской бой

Вы перешли по ссылке на страницу, которой пока не существует. Чтобы её создать, наберите текст в окне, расположенном ниже (подробнее см. [справочн](#) нажмите кнопку **назад** своего браузера.



Здравствуйте, Елена Александровна! Мне понравилась Ваша игра. Правда я не могу пройти дальше второго уровня :(

Мои вопросы:

# Как Вам пришла в голову эта игра?

# Вы ее долго делали?

# В каком редакторе Вы рисовали фон игры и персонажей?

Замечания:

# Жизни нарисованы не ровно

# Мне не понравился черный крест в конце игры

Предложения:

# Хорошо бы выбирать уровень игры. Хотя я знаю, что это сложно наверно

Жду новых Ваших игр! ---~~~~~|

*Рис. 66. Пример заполненной страницы обсуждения в режиме Править.*

Для вопросов используйте фразы с таким началом:

- Когда? Как? Где? Почему?
- Верно ли я понял(а), что ...
- А что будет, если ...
- Можно ли ...
- В каком месте скрипта ...
- и т.д.

Для замечаний можно использовать фразы, которые начинаются примерно так:

- Было бы лучше, если ...
- Когда нажимаешь, то ... (*указываете ошибку, которую не заметил Автор*)
- Жаль, что нельзя ...
- и т.д.

Предложения пишите так, словно ставите себя на место Автора. Всегда можно что-то посоветовать, предложить. Мы вместе ищем лучшие пути, поэтому приветствуется любая идея, которая поможет автору сделать проект лучше.

## Обсуждение:Проект ScratchDuinoЛаборатория Морской бой

Здравствуйте, Елена Александровна! Мне понравилась Ваша игра. Правда я не могу пройти дальше второго уровня :(

Мои вопросы:

1. Как Вам пришла в голову эта игра?
2. Вы ее долго делали?
3. В каком редакторе Вы рисовали фон игры и персонажей?

Замечания:

1. Жизни нарисованы не ровно
2. Мне не понравился черный крест в конце игры

Предложения:

1. Хорошо бы выбирать уровень игры. Хотя я знаю, что это сложно наверно

Жду новых Ваших игр! --Зиновьева Дарья ([обсуждение](#)) 14:46, 23 июня 2015 (MSK)

*Рис. 67. Пример заполненной страницы обсуждения  
в режиме Читать.*



Анонимные обсуждения Экспертной группой фестиваля не принимаются во внимание! В конце текста всегда нажимайте кнопку Ваша подпись и момент времени.

## Использованные источники

1. Бочкала А. Е. Общедоступное программирование в Scratch / А. Е. Бочкала. Режим доступа: <http://scratch.uvk6.info> (дата обращения: 02.07.2015)
2. Летописи Режим доступа: [Letopisi.org/index.php/Скретч](http://Letopisi.org/index.php/Скретч) (дата обращения: 02.07.2015)
3. Скретч Вики Режим доступа: [Scratch-ru.info/wiki/Заглавная страница](http://Scratch-ru.info/wiki/Заглавная_страница) (дата обращения: 02.07.2015)
4. Шапошникова С. В. Введение в Scratch : цикл уроков по программированию для детей / С. В. Шапошникова. Режим доступа : [http://younglinux.info/sites/default/files/scratch\\_lessons.pdf](http://younglinux.info/sites/default/files/scratch_lessons.pdf) (дата обращения: 02.07.2015)

## Информационная среда проекта

Официальный сайт ScratchDuino	<a href="http://www.scratchduino.ru/">http://www.scratchduino.ru/</a>
Файл-сервер	<a href="http://files.scratchduino.ru/">http://files.scratchduino.ru/</a>
Wiki ScratchDuino	<a href="http://wiki.scratchduino.ru/">http://wiki.scratchduino.ru/</a>
Группа ScratchDuino на Google+	<a href="https://goo.gl/uVRm6D">https://goo.gl/uVRm6D</a>
Видеоканал ScratchDuino на YouTube	<a href="https://goo.gl/Y5jDz8">https://goo.gl/Y5jDz8</a>
Группа Вконтакте	<a href="http://vk.com/scratchduino">http://vk.com/scratchduino</a>

**Адрес русскоязычной техподдержки** [support@scratchduino.ru](mailto:support@scratchduino.ru)

## Приложение

### **Положение о фестивале «Свободная робототехника ScratchDuino»**

#### *1. Общие положения*

Всероссийский (Международный при участии команд из других стран) фестиваль «**Свободная робототехника ScratchDuino**» (далее — Фестиваль) проводится ежегодно в соответствии с задачами государственной политики в области развития образовательной робототехники и непрерывного IT-образования в Российской Федерации.

*Учредители Фестиваля:* ЗАО «Тырнет» и партнерские организации.

*Целями Фестиваля являются:*

- обобщение и распространение опыта использования свободной робототехники;
- внедрение свободной робототехники в учреждения основного и дополнительного образования, а также в семейной форме и в форме самообразования.

*Участники Фестиваля:* педагогические работники, ученики и творческие коллективы, состоящие из взрослых и детей, по уровням и видам образования.

#### *2. Организация Фестиваля*

2.1. Подготовка и проведение Фестиваля осуществляется Организационным комитетом.

2.2. Организационный комитет:

- формирует состав Экспертной группы;
- информирует о сроках, порядке проведения и результатах Фестиваля;
- проводит рассмотрение фестивальных материалов;
- проводит поощрение активных участников Фестиваля.

#### *3. Содержание и порядок проведения Фестиваля*

3.1. Фестиваль проводится в 3 этапа.

1-й этап — Практический (сентябрь–декабрь):

- участники фестиваля регистрируются на портале <http://wiki.scratchduino.ru> и размещают портфолио фестивальных проектов;
- участники фестиваля изучают работы друг друга, пишут отзывы и задают авторам вопросы;
- участники подают заявку в таблице самозаписи на странице Фестиваля;
- в Группе ScratchDuino на Google+ <https://goo.gl/uVRm6D> проводятся консультационные семинары;
- информирование о Фестивале в СМИ.

2-й этап — Экспертный (январь):

- работа Экспертной группы;
- определение активных участников Фестиваля по номинациям и уровням образования.

3-й этап — Финальный (7 февраля — Всемирный день робототехники)

- объявление и поощрение активных участников Фестиваля.

3.2. Экспертиза фестивальных материалов проходит по двум (четырёх, если будут проекты) номинациям по уровням образования:

- ScratchDuino.Лаборатория;
- ScratchDuino.Робоплатформа;
- ScratchDuino.Лаборатория (модификация);
- ScratchDuino.Робоплатформа (модификация).

3.3. Фестивальные проекты размещаются участниками самостоятельно на портале <http://wiki.scratchduino.ru/> в виде портфолио проекта с обязательным указанием категорий [[Категория: Проекты]]

[[Категория: ScratchDuino 2016]]

3.4. Фестивальные материалы экспертной группой не рецензируются.

3.5. Имена активных участников Фестиваля будут размещены на главной странице сайта 7 февраля.

#### 4. Критерии отбора активных участников Фестиваля

- количество и качество размещённых проектов;

— количество и качество проведённых обсуждений проектов других участников.

#### *5. Сроки проведения Фестиваля*

5.1. Портфолио Проектов размещается на сайте <http://wiki.scratchduino.ru/> до \_\_\_\_ « \_\_\_\_ » 20\_\_ г.

5.2. Список активных участников и виды поощрения будут опубликованы на сайте 7 февраля — во Всемирный день робототехники.

#### *6. Поощрение участников Фестиваля*

6.1. Участники Фестиваля получают электронные сертификаты.

6.2. На основании экспертных оценок фестивальных материалов определяются **УЧАСТНИКИ, ЛАУРЕАТЫ и ПОБЕДИТЕЛИ** Фестиваля по номинациям и по уровням образования.

*Руководство пользователя*

**Вострикова** Елена Александровна, кандидат педагогических наук,  
начальник отдела развития образования МАОУ ДПО ИПК, г. Новокузнецк

[elena.vostrikova@scratchduino.com](mailto:elena.vostrikova@scratchduino.com)

**Захаров** Леонид Сергеевич, инженер-программист ФГКУ УВО ГУ  
МВД России по Кемеровской области, г. Новокузнецк

**Львова** Екатерина Анатольевна, ведущий инженер-разработчик  
проекта ScratchDuino, г. Санкт-Петербург

[ekaterina.lvova@scratchduino.com](mailto:ekaterina.lvova@scratchduino.com)

**ScratchDuino.Лаборатория**

Корректор Е. В. Толстякова

---

Сдано в печать ХХ.ХХ.2015. Тираж 1000 экз.

---

Множительный центр ЗАО «Тырнет»

197022 г. Санкт-Петербург, пр. Медиков, 5, корп. 7