

Руководство пользователя
RobboScratch

Оглавление

Введение	4
1. Функциональные возможности RobboScratch	5
2. Интерфейс RobboScratch	6
3. Формы блоков	8
4. Базовые блоки	10
4.1 События.....	10
4.2 Управление	11
4.3 Сенсоры.....	12
4.4 Операторы	14
4.5 Переменные	15
4.6 Другие блоки	15
5. Управление роболатформой	18
5.1. Описание возможностей роболатформы	18
5.2. Команды управления роболатформой	19
5.3. Подключение и проверка роболатформы в RobboScratch	21
5.4. Программирование роболатформы	25
6. Управление набором расширений роболатформы.....	27
6.1. Описание возможностей набора расширений	27
6.2. Команды управления набора расширений роболатформой	28
6.3. Подключение набора расширений роболатформы	29
6.4. Настройка захвата-манипулятора «Клешня»	31
6.5. Программирование «Клешни»	33
6.6. Автокоррекция RGB датчиков	35
6.7. Программирование RGB датчиков цвета	41
6.8. Программирование ультразвукового дальномера	42
7. Лаборатория	44
7.1. Описание возможностей лаборатории	44
7.2. Команды управления лабораторией.....	46
7.3. Подключение и проверка лабораторией в RobboScratch	47
7.4. Программирование лаборатории в RobboScratch.....	50
8. Набор расширений лаборатории	51

8.1. Описание возможностей набора расширений лаборатории	51
8.2. Команды управления набором расширений лаборатории	52
8.3. Программирование набора расширений лаборатории.....	52
9. Управление квадрокоптером	57
9.1. Описание возможностей квадрокоптера	57
9.2. Техника безопасности при полетах квадрокоптера	57
9.3. Состояние квадрокоптера по светодиодам	58
9.4. Команды управления квадрокоптером	59
9.5. Подключение квадрокоптера к RobboScratch	60
9.6. Примеры программ взлета квадрокоптера	62
10. Управление танцующим роботом	64
10.1. Описание возможностей танцующего робота	64
10.2. Команды управления танцующим роботом	64
10.3. Подключение и программирование танцующего робота	65
11. Управление набором «Схемотехника»	69
11.1. Описание возможностей	69
11.2. Команды управления	69
11.3. Подключение платы Arduino	70
11.4. Программирование Arduino	71
Дополнительная информация.....	73

Введение

Добро пожаловать в руководство пользователя программы RobboScratch!

RobboScratch – это усовершенствованная версия популярной среды визуального программирования Scratch, дополненная функциями, позволяющими подключать и программировать оборудование компании РОББО.

В RobboScratch вы будете создавать интерактивные проекты и игры, управлять роботами, квадрокоптерами и другими электронными устройствами. Это развивает логическое мышление, творческие способности и технические навыки. С помощью RobboScratch вы освоите базовые концепции программирования, электрических цепей и поймете, как работают электронные устройства.

RobboScratch предназначен для школьников, студентов, преподавателей и всех, кто интересуется программированием и робототехникой. Программа подходит для использования как в образовательных учреждениях или кружках, так и для самостоятельного изучения.

Благодаря интуитивно понятному интерфейсу и блочному принципу программирования начать работу с программой могут даже новички, не имеющие опыта в программировании. В то же время возможности RobboScratch позволяют создавать сложные проекты: программировать шины электронных устройств, программировать собственные изделия на платформе Arduino, создавать и использовать в работе новые переменные.

Разделы 1 – 5 описывают базовые возможности Scratch. Это необходимо для понимания логики блочного программирования.

Разделы 6 – 12 посвящены оборудованию РОББО: описаны возможности оборудования, блоки управления устройствами и предложены упражнения, протестированные на практике с преподавателями и учениками.

Скачать RobboScratch: <http://files.robbo.ru/Software/RobboScratch3.0/>.

1. Функциональные возможности RobboScratch

- **Блочное программирование:** создание программ путем перетаскивания и соединения блоков кода без синтаксиса.
- **Программирование робоплатформы:** управление сервомоторами, светодиодами, счетчиками шагов, кнопками, датчиком освещенности, датчиком линии, датчиком приближения, датчиком касания, аналоговыми и дискретными выходами.
- **Программирование лаборатории:** управление потенциометром, динамиком, светодиодами, датчиками света и звука, кнопками, аналоговыми и дискретными выходами.
- **Программирование квадрокоптера:** управление влетом, полетом и посадкой.
- **Программирование танцующего робота:** управление сервомоторами, LED-матрицей, ультразвуковым дальномером, светодиодами, излучателем звука.
- **Программирование платы Arduino и набора «Схемотехника»:** управление датчиками, LED-матрицей, аналоговыми и дискретными выходами;
- **Поддержка мультимедийных элементов:** использование изображений, звуков и музыки.

С оборудованием РОББО обучение происходит следующим образом:

1. Наглядное представление общего

Программист изучает оборудование и обдумывает, какого результата он хочет добиться.

2. Деление задачи на мелкие части

Программист шаг за шагом определяет события, с помощью которых сможет достичь желаемого результата.

3. Разработка программы

Программа складывается из разноцветных блоков. Пользователям ничего не нужно писать, как в других языках программирования. Блоки имеют защелки, которые не позволяют соединить несовместимые блоки.

4. Запуск программы

Программист подключает оборудование и видит, как оборудование выполняет написанную программу.

2. Интерфейс RobboScratch

На рисунке 1 показаны основные элементы интерфейса RobboScratch.

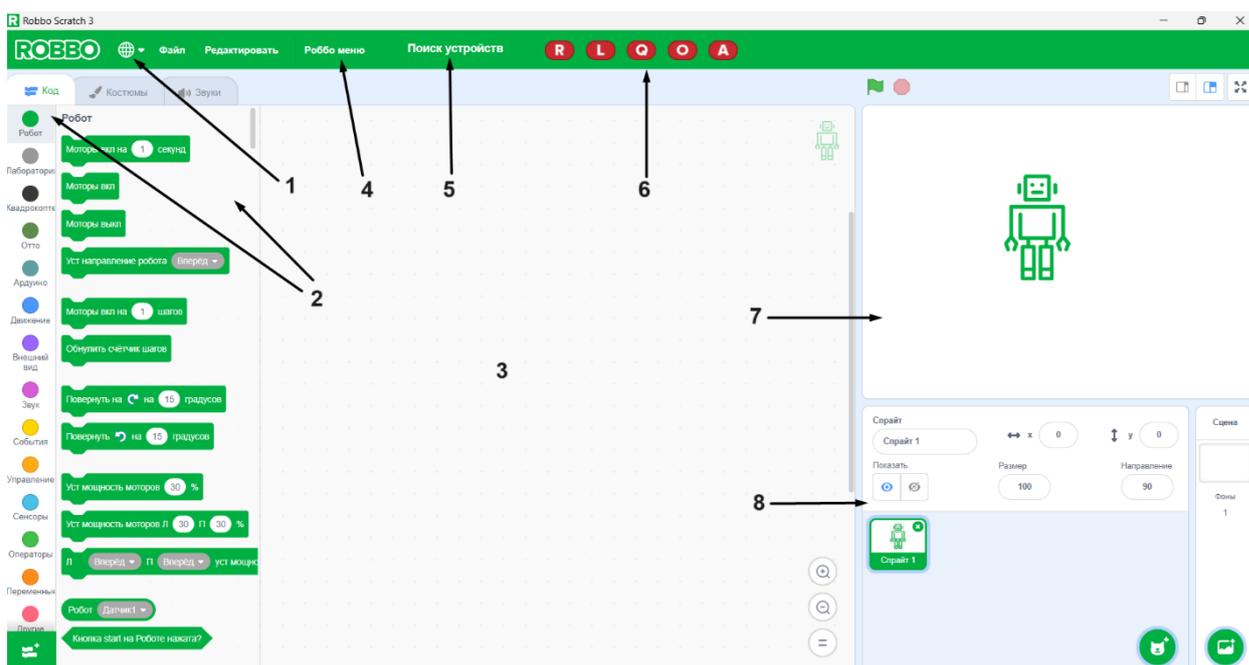


Рисунок 1 - Интерфейс RobboScratch

1 – Значок глобуса: смена языка, интерфейс полностью переведен на русский и английский языки.

2 – Меню команд: здесь расположены наборы цветных блоков, из которых состоит программа. У каждого оборудования свои собственные блоки, описывающие поведение только этого оборудования. На скриншоте показаны блоки робоплатформы. Есть общие блоки для создания логики программы.

3 – Зона программирования: здесь из блоков собираются программы.

4 – Роббо меню: здесь подключаются наборы расширений для робоплатформы и лаборатории и расположены таблицы коррекции для датчиков цвета.

5 – Поиск устройства: кнопка формирует связь между компьютером и подключенным оборудованием.

6 – Показания датчиков оборудования: здесь можно проследить, как программа влияет на оборудование. Кнопка R соответствует робоплатформе, кнопка L -лаборатории, кнопка O - танцующему роботу, кнопка A - плате Arduino.

7 – Сцена: в этой зоне перемещаются спрайты – это персонажи, которых можно программировать без подключенного оборудования.

8 – Зона редактирования спрайта: здесь можно нарисовать фон, изменить внешний вид спрайта или добавить нескольких спрайтов.

На рисунке 2 выделены две кнопки: позиция 9 – это редактор спрайта, позиция 10 – редактор фона.

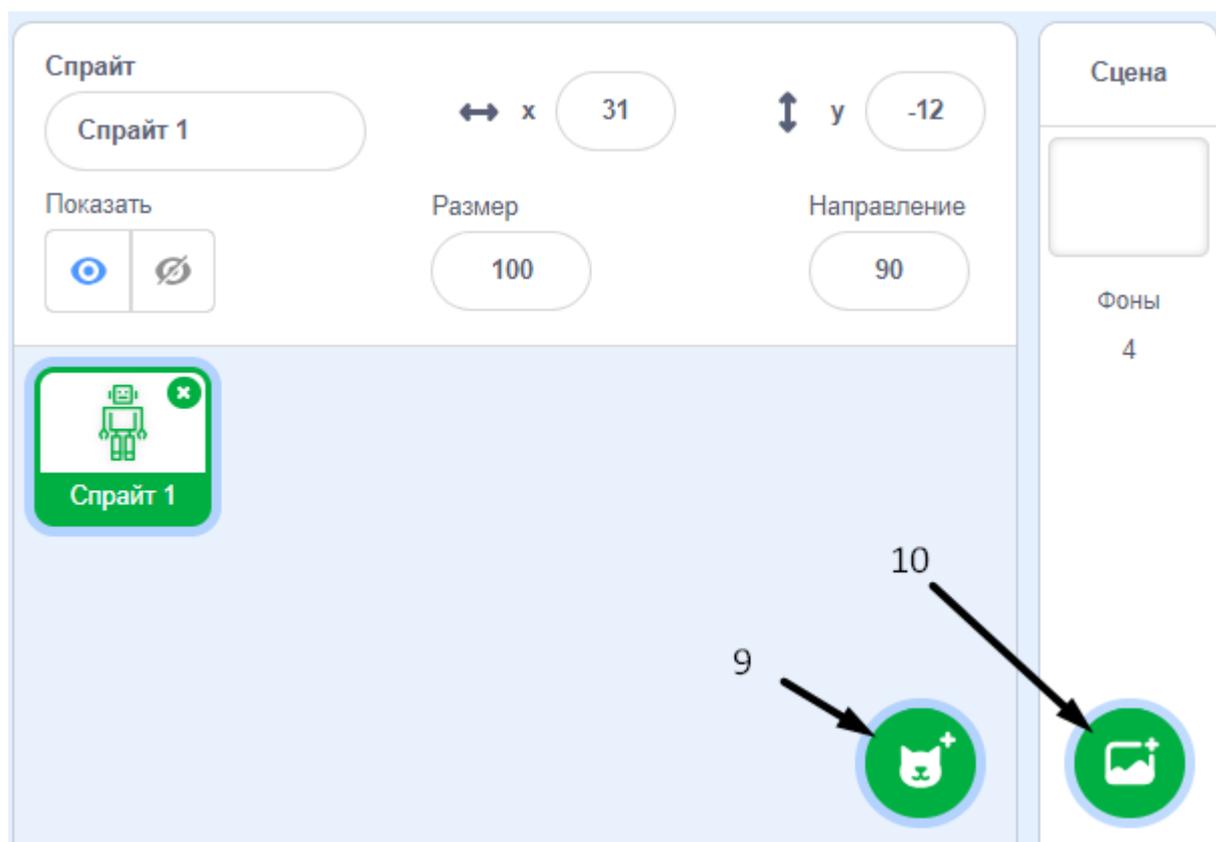


Рисунок 2 – Редактор фона и спрайта

3. Формы блоков

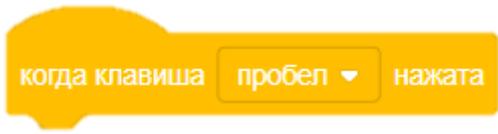
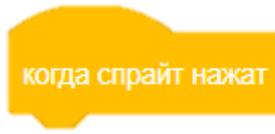
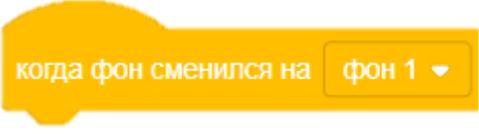
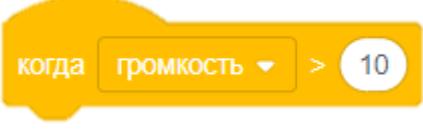
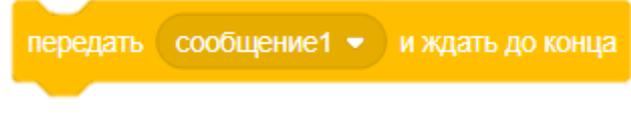
Существуют 6 форм блоков: шапки, обычные, логические, репортёры, обвивающие и закрывающие. Каждый блок имеет выступ для прикрепления блоков сверху и/или снизу.

Форма блока	Описание блока
	<p>Название: шапка.</p> <p>Событие, которое начинает любую программу. Например, щелчок по зелёному флагу, нажатие клавиши.</p> <p>Имеет выступ для крепления блоков ниже себя.</p>
	<p>Название: обычный блок.</p> <p>Команда с действием. Это самая большая категория блоков.</p> <p>Имеет выступ для крепления блоков выше и ниже себя.</p>
	<p>Название: логический блок.</p> <p>Проверяет заданное условие, и передает верно это условие или нет.</p> <p>Имеет форму шестиугольника и вставляется в блоки управления.</p>
	<p>Название: репортёр.</p> <p>Передает какие-либо значения. Например, значения с датчиков, числа, строки или переменные.</p> <p>Имеет форму закругленного прямоугольника и вставляется в блоки управления.</p>
	<p>Название: обвивающий блок.</p> <p>Может быть блоком повторения блоков внутри себя или блоком выполнения после проверки условий.</p> <p>Имеют выступ для крепления блоков сверху, внутри и снизу.</p>

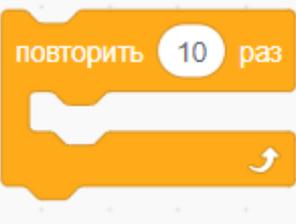
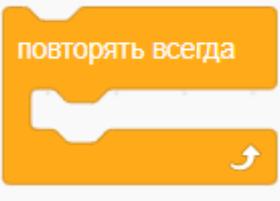
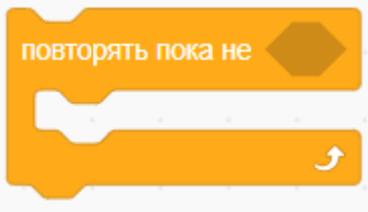
	<p>После выполнения блоков и его повторения (если он будет повторять) блок делает паузу около 10 миллисекунд.</p>
	<p>Название: закрывающий блок.</p> <p>Событие, которое заканчивает любую программу.</p> <p>Имеет выступ для крепления блоков выше.</p>

4. Базовые блоки

4.1 События

Блок события	Что блок делает?
	Запускает программу, если нажать зеленый флаг над сценой.
	Запускает программу, если нажать определенную клавишу. В примере программа запустится, если нажать пробел.
	Запускает программу, если нажать на спрайт (персонаж на сцене).
	Запускает программу, если сменился фон.
	Запускает программу, если переменные громкость/таймер больше указанного значения.
	Запускает программу, если получено сообщение от любого спрайта (персонажа на сцене).
	Передает сообщение всем спрайтам.
	Передает сообщение всем спрайтам и переходит к следующему блоку кода.

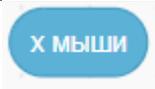
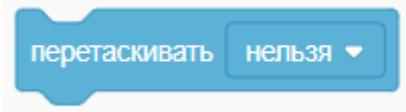
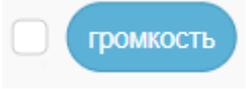
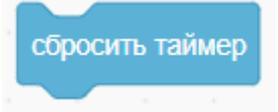
4.2 Управление

Блок управления	Что блок делает?
	<p>Останавливает выполнение программы на заданное количество секунд.</p>
	<p>Повторяет блоки внутри себя заданное количество раз.</p>
	<p>Постоянно повторяет действия внутри себя.</p>
	<p>Выполняет внутренние блоки, если условие верно. Если неверно – внутренние блоки пропускаются.</p>
	<p>Если условие верно, выполняются первые внутренние блоки, если неверно – вторые внутренние блоки.</p>
	<p>Ждет, когда условие будет выполнено.</p>
	<p>Выполняет внутренние блоки, пока логическое условие неверно.</p>
	<p>Останавливает проект, скрипт и всех спрайтов.</p>

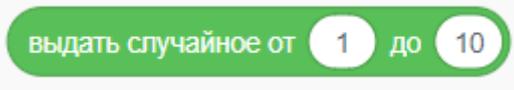
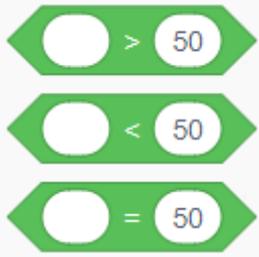
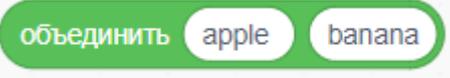
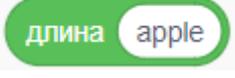
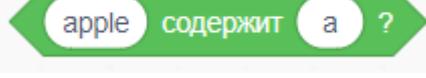
	Добавьте этот блок в зону программирования, чтобы экстренно остановить программу.
	Повторяет действия клона.
	Создает клон спрайта.
	Удаляет клон спрайта.

4.3 Сенсоры

Блок сенсора	Что блок делает?
	Проверяет, касается ли спрайт указателя мыши, края или другого спрайта.
	Проверяет, касается ли оборудование или спрайт заданного цвета.
	Проверяет, касается ли первый цвет спрайта второго цвета.
	Передает расстояние в пикселях между спрайтом и указателем мыши или указанным спрайтом, даже если он скрыт.
	Спрайт произносит заданный текст.
	Возвращает последний текст, введенный с помощью блока
	Проверяет, нажата ли указанная клавиша.
	Проверяет, нажата ли любая из кнопок мыши.

	Возвращает координату X указателя мыши.
	Возвращает координату Y указателя мыши.
	Устанавливает оборудование или спрайт как перетаскиваемое или нет.
<input type="checkbox"/> 	Возвращает громкость шума, принимаемого микрофоном, по шкале от 0 до 100.
<input type="checkbox"/> 	Возвращает количество времени, прошедшее с момента загрузки проекта или последнего сброса таймера.
	Устанавливает значение таймера обратно в 0.0.
	Возвращает указанное значение указанного оборудования/спрайта/сцены.
<input type="checkbox"/> 	Возвращает текущий локальный год, месяц, дату, день недели, час, минуту или секунду, в зависимости от аргумента.
	Возвращает количество дней (и частей дня) с 00:00:00 1 января 2000 года.
<input type="checkbox"/> 	Возвращает имя пользователя, просматривающего проект.

4.4 Операторы

Блок оператора	Что блок делает?
	<p>Блоки:</p> <ol style="list-style-type: none"> 1) сложения, 2) вычитания, 3) умножения, 4) деления <p>двух репортеров (чисел, переменных или других значений).</p>
	<p>Выдает случайное число в заданных пределах.</p>
	<p>Блоки сравнения значения двух репортеров (чисел, переменных или других значений).</p>
	<p>Передает ответ: «да», если оба условия выполнены.</p>
	<p>Передает ответ: «да», если одно из условий выполнилось.</p>
	<p>Передает обратное значение условия.</p>
	<p>Объединяет две строки в одну.</p>
	<p>Передает заданный номер буквы в строке.</p>
	<p>Передает длину строки.</p>
	<p>Передает значение «да» или «нет», если в строке содержится символ.</p>
	<p>Определяет остаток от деления первого значения на второе.</p>

	Округляет число до целого.
	Определяет модуль целого меньшего, целого большего, квадратного корня, sin, cos, tan и т. д. у числа.

4.5 Переменные

Блок переменной	Что блок делает?
	Кнопки создания: <ul style="list-style-type: none"> • переменной, • списка, • блока.
	Если поставить галочку у переменной, она будет отображаться на сцене рядом со спрайтом.
	Задаёт значение выбранной переменной.
	Изменяет значение выбранной переменной.
	Показывает на сцене заданную переменную.
	Убирает заданную переменную со сцены.

4.6 Другие блоки

Другие блоки – это категория пользовательских блоков.

Чтобы создать собственный блок, нажмите кнопку

В открывшемся окне выберите конфигурацию нового блока:

- значение – это блок репортер, может передавать какие-либо значения. Например, значения с датчиков, числа, строки или переменные;
- логический блок проверяет заданное условие и передает, верно это условие или нет;
- подпись можно добавить после значения или логического блока.

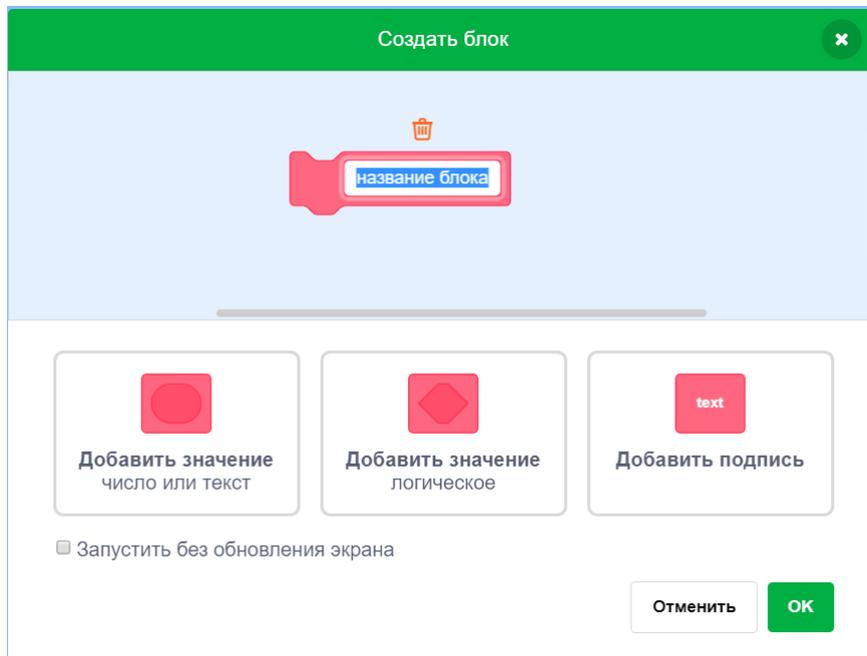
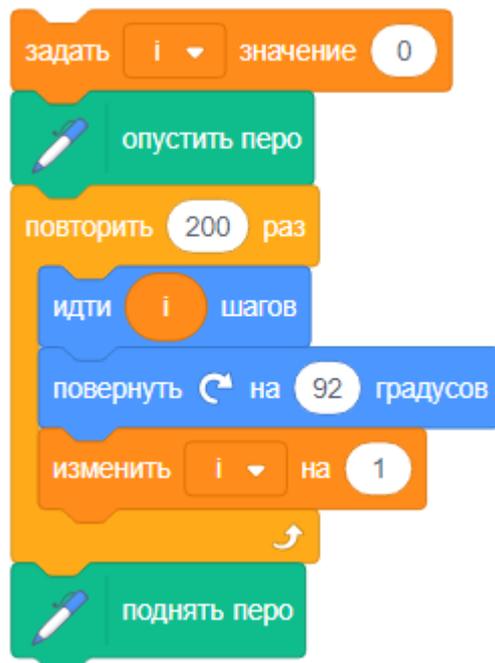


Рисунок 3 – Создание пользовательского блока

При включении опции «запустить без обновления экрана» запуск программы будет проходить быстрее, без промежуточных шагов.

В программе ниже спрайт рисует картинку. Если запустить скрипт с опцией без обновления экрана, то пользователь увидит результат, а не то, как спрайт рисует картинку.



5. Управление робоплатформой

5.1. Описание возможностей робоплатформы

В состав набора робоплатформы входят датчик касания, датчик освещенности, светодиодный датчик, датчик приближения, два датчика линии.



Рисунок 4 – Робоплатформа с набором датчиков

На схеме ниже указаны элементы робоплатформы, которые можно запрограммировать в RobboScratch.

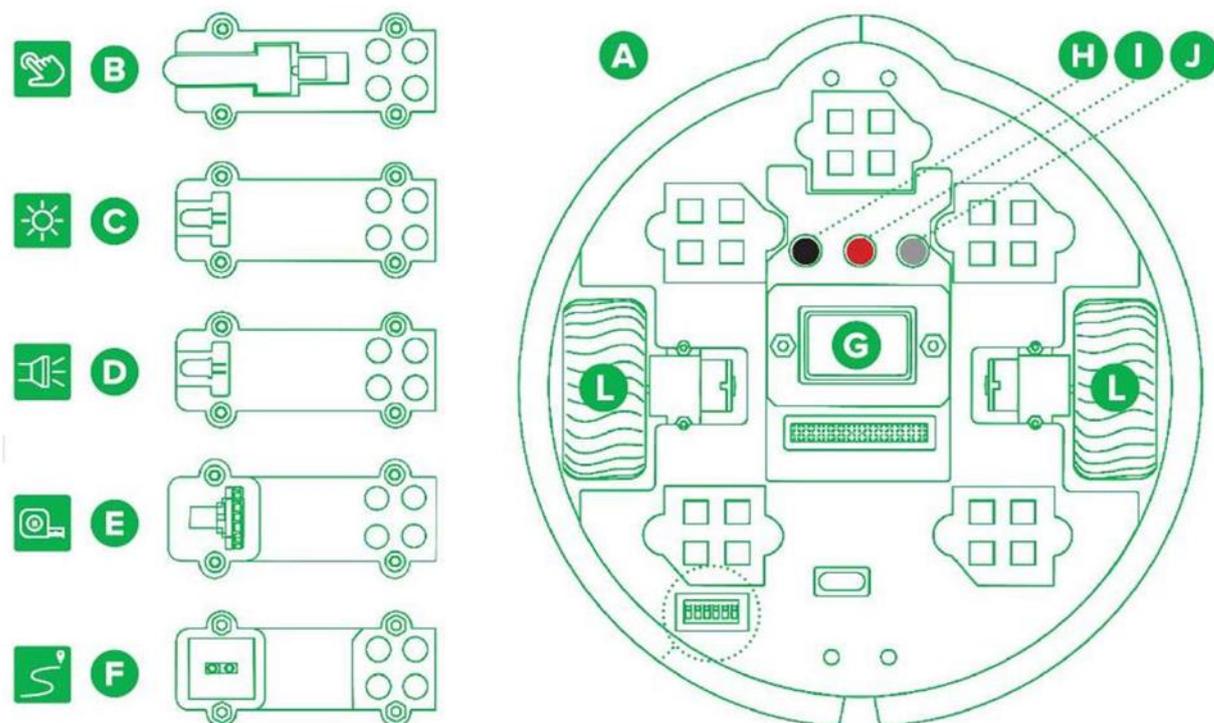


Рисунок 5 – Элементы управления робоплатформой

Описание частей и элементов управления:

- A - робоплатформа;
- B - датчик касания - распознает прикосновение;
- C - датчик освещенности - определяет, насколько светло или темно в комнате;
- D - светодиодный модуль - работает как фонарик;
- E - ИК датчик приближения - определяет приближенность к объекту;
- F - датчик линии - распознает светлую и темную поверхности;
- G - слот для аккумулятора типа "Крона" – аккумулятор нужен для подключения по Bluetooth;
- H - кнопка Reset - перезагружает робоплатформу;
- I - кнопка ON/OFF - включает или выключает питание от батареи;
- J - кнопка Start - программируемая кнопка;
- L - колеса.

Значения с датчиков, кнопок и моторов показаны в меню робоплатформы (Рисунок 6)

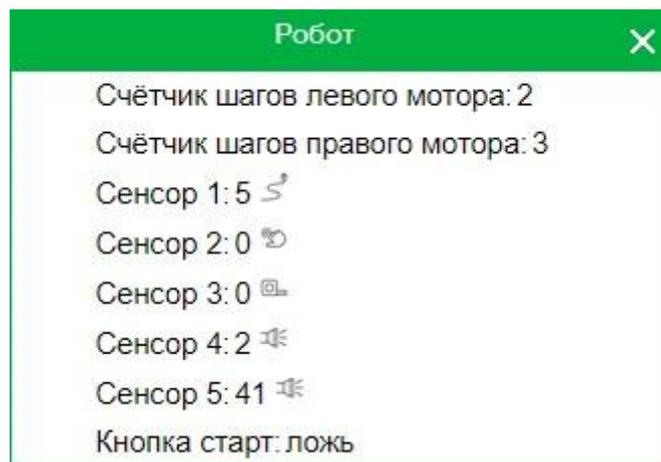
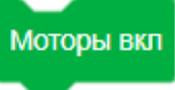
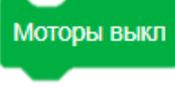
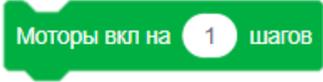
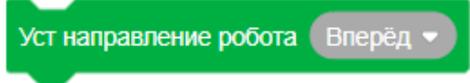
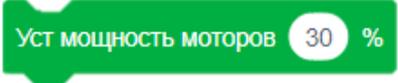
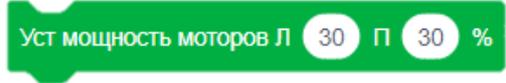
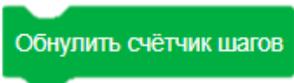
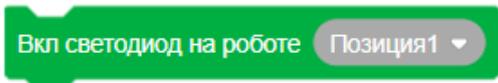
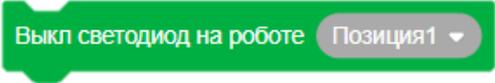


Рисунок 6 – Меню робоплатформы

5.2. Команды управления робоплатформой

Блоки движения	Что блок делает?
	Запускает движение робоплатформы на заданное количество секунд
	Запускает движение робоплатформы
	Останавливает движение робоплатформы

	<p>Запускает движение робоплатформы на заданное количество шагов</p>
	<p>Устанавливает направление робоплатформы. Можно задать направление: вперед, назад, влево, вправо.</p>
	<p>Устанавливает мощность обоих моторов робоплатформы. Чем выше значение, тем быстрее робоплатформа будет ехать.</p>
	<p>Устанавливает мощность отдельно левого (Л) и правого (П) моторов. Мощность задается в процентах. При разных значениях мощности моторов робоплатформа будет поворачиваться влево или вправо.</p>
	<p>Устанавливает направление и мощность левого (Л) и правого (П) мотора. Можно задать направление вперед и назад. Мощность задается в процентах.</p>
	<p>Обнуляет счетчики шагов в показаниях датчиков робоплатформы (кнопка ).</p>
	<p>Поворачивает робоплатформу вправо на 15 градусов.</p>
	<p>Поворачивает робоплатформу влево на 15 градусов.</p>
	<p>Включает светодиодный датчик. Нужно выбрать позицию 1-5 в зависимости от того, в какой слот датчик подключен.</p>

	<p>Выключает светодиодный датчик. Нужно выбрать позицию 1-5 в зависимости от того, в какой слот датчик подключен.</p>
	<p>Блок репортер: фиксирует номер слота датчика. Доступные варианты: слоты датчиков 1-5, левый и правый счетчики шагов.</p>
	<p>Логический блок: передает значения, нажата ли серая кнопка Start.</p>

5.3. Подключение и проверка робоплатформы в RobboScratch

Шаг 1. Подключите робоплатформу к компьютеру через micro-USB или по Bluetooth.

Подключение через micro-USB.

Не вставляйте аккумулятор в слот робоплатформы.

Подключите робоплатформу к компьютеру с помощью кабеля micro-USB.

После успешного сопряжения светодиод внизу робоплатформы будет гореть постоянно (рис.6).

Подключение по Bluetooth.

1. Вставьте литиевый аккумулятор Крона на 9В в слот робоплатформы.

Светодиод внизу робоплатформы начнет мигать: Bluetooth робоплатформы готов к сопряжению. После успешного сопряжения этот индикатор будет гореть постоянно.

2. Откройте параметр системы Bluetooth на компьютере и выполните поиск ближайших устройств.

3. Выберите устройство типа: "ROB-R-X-XX-XX".

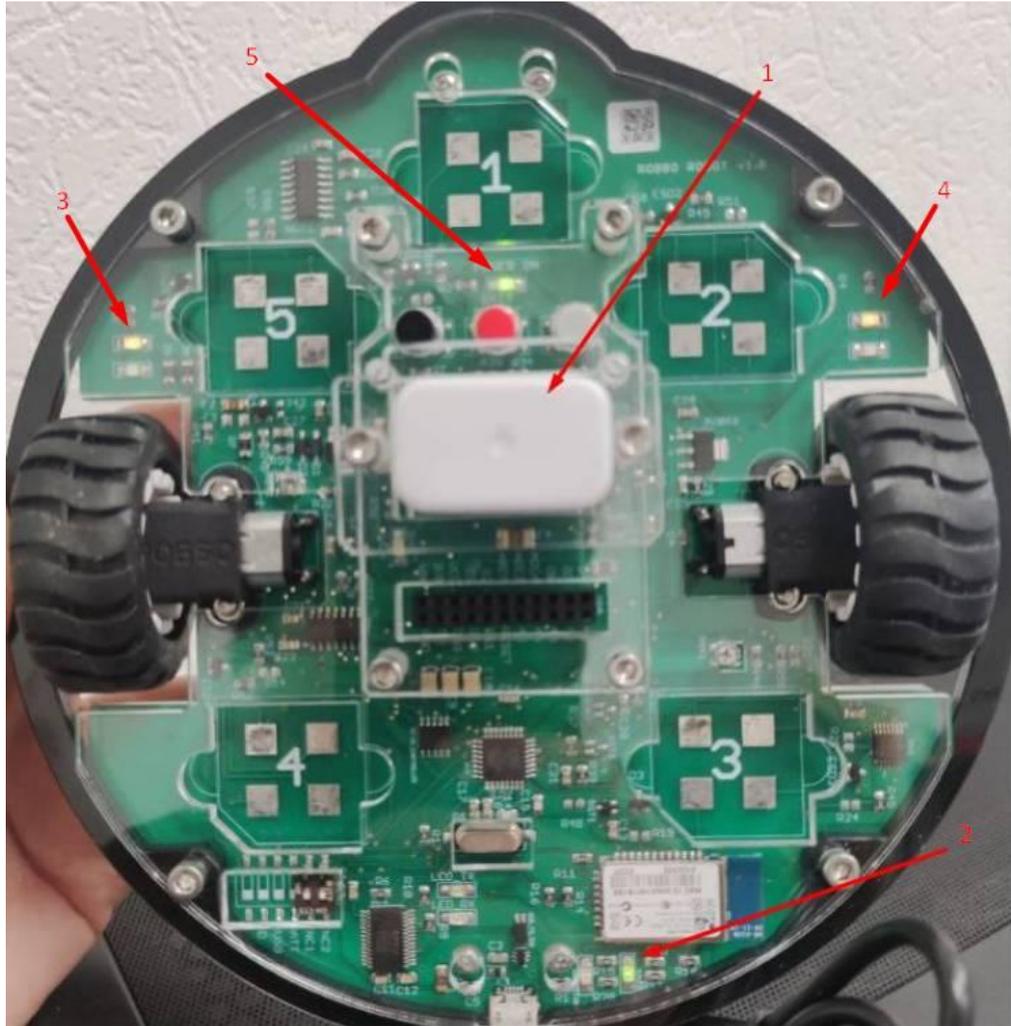
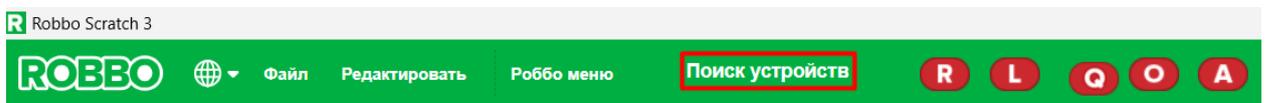


Рисунок 7 – Подключение робоплатформы: 1 - аккумулятор в слоту робоплатформы, 2 - индикатор Bluetooth, 3 и 4 - индикаторы готовности моторов робоплатформы, 5 - индикатор питания робоплатформы

Шаг 2. Откройте RobboScratch и нажмите **Поиск устройств**.



Если появится окно с предложением обновления прошивки робоплатформы, нажмите **ОК**.

Когда робоплатформа подключится, кнопка R станет зеленого цвета:  > 

Шаг 3. Подключите датчики к робоплатформе.

В любой слот можно подключить любой датчик. В шаге 5 нужно будет выбрать, в какой слот подключен конкретный тип датчика.

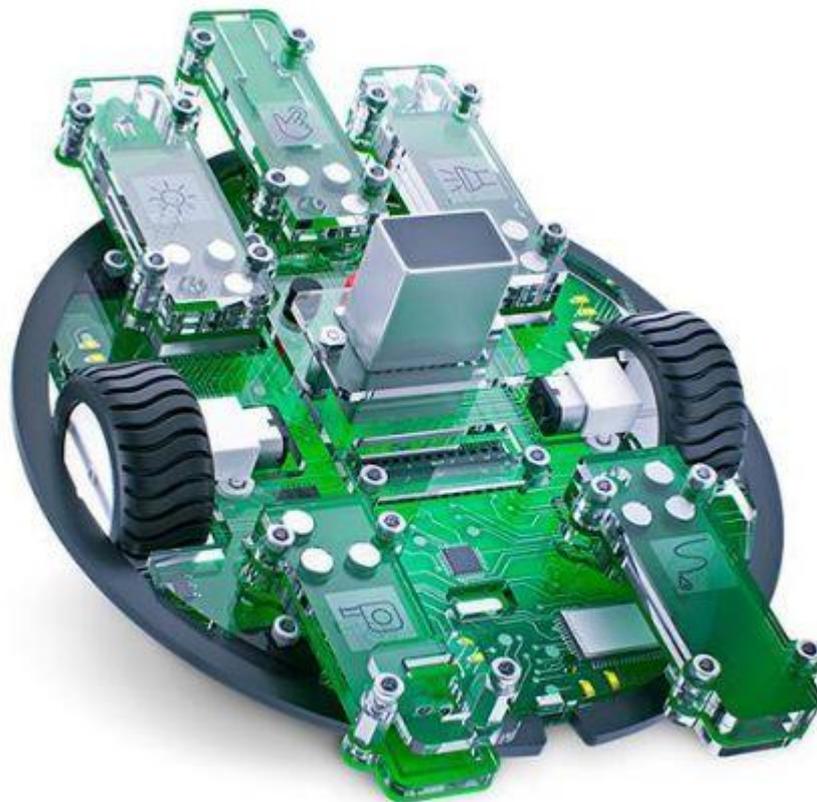


Рисунок 8 – Датчики в робоплатформе

Шаг 4. В RobboScratch нажмите **Роббо меню > Включить расширенный набор датчиков робота.**

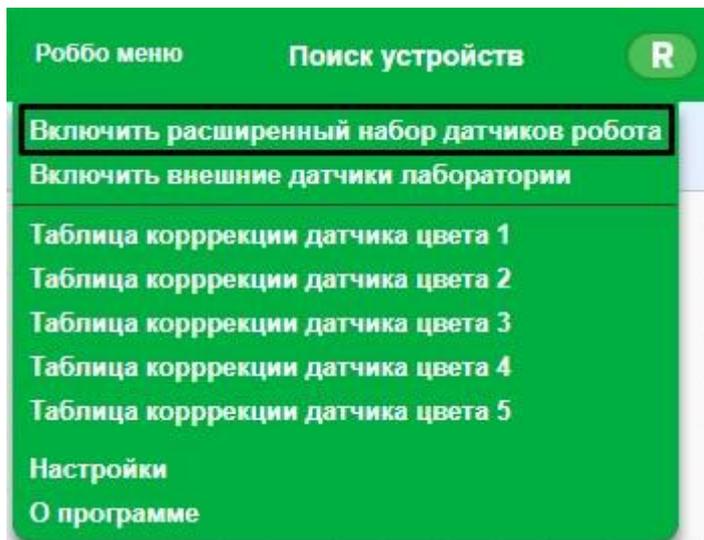
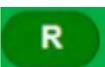


Рисунок 9 – Подключение датчиков в RobboScratch

Шаг 5. Нажмите , чтобы отобразить меню датчиков робоплатформы.

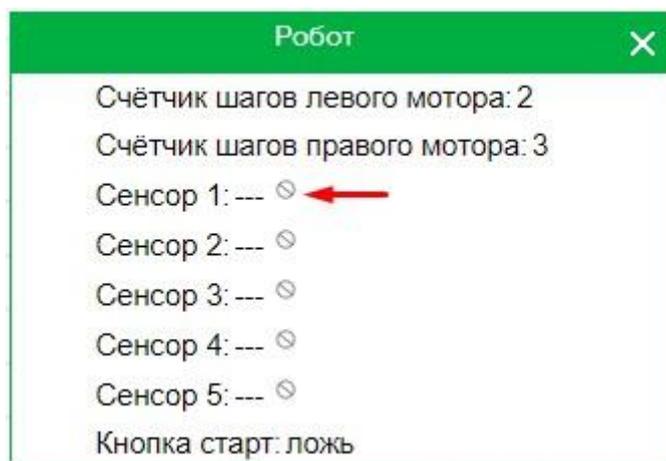


Рисунок 10 – Меню датчиков робоплатформы

Шаг 6. Выбор типа датчика.

Нажмите на  возле одного из сенсоров, откроется меню с выбором типа:

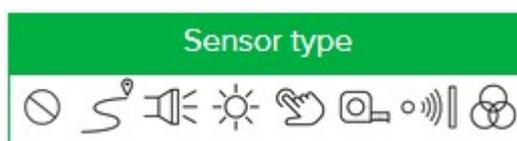


Рисунок 11 – Выбор типа датчика

Рисунок датчика в открывшемся меню такой же, как на самом датчике. Слоты датчиков подписаны на робоплатформе и имеют номера 1-5.

Выберите нужный тип датчика в RobboScratch в соответствии с номером слота на робоплатформе и изображением датчика. Например:

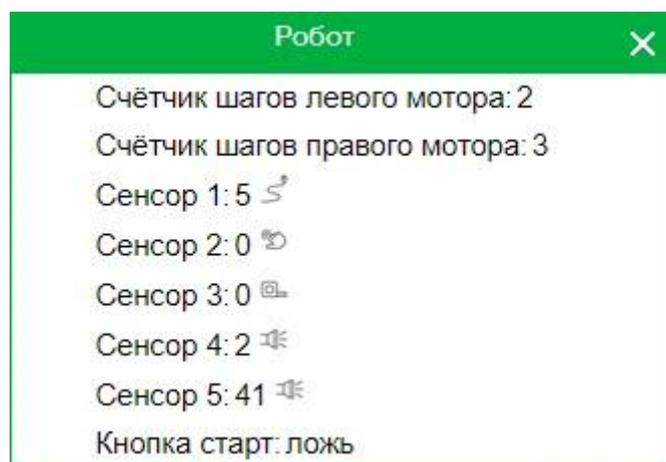


Рисунок 12 – Датчики определены в RobboScratch

Шаг 7. Проверьте, как изменяются значения в RobboScratch:

- датчик касания – реагирует на прикосновение;
- два датчика линии – реагируют на черную линию на пути у робота;
- датчик освещенности – реагирует на близость источника освещения;

- датчик расстояния – реагирует, когда робот приближается к препятствию;
- светодиод – работает как фонарик.

5.4. Программирование робоплатформы

Ниже описано несколько сценариев. Повторите их, чтобы научиться управлять робоплатформой.



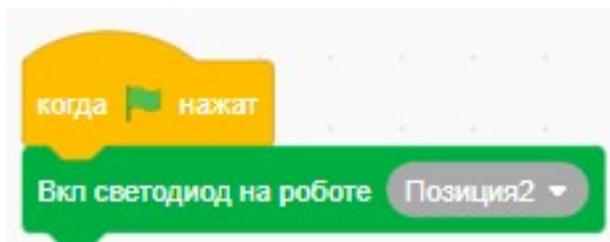
В программах используются команды из меню

Перед созданием программ убедитесь, что робоплатформа подключена к RobboScratch.

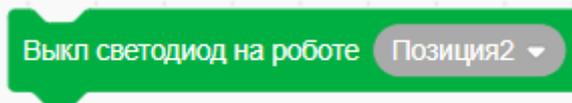
1. Как зажечь светодиод?

Шаг 1. Подключите светодиодный датчик во второй слот робоплатформы.

Шаг 2. Составьте программу, которая изображена ниже.



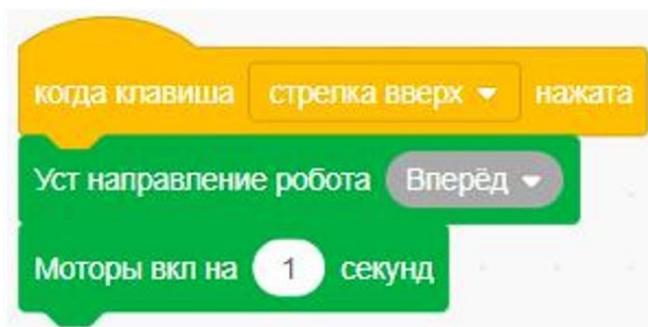
Светодиодный датчик засветится. Чтобы его отключить, добавьте в зону



программирования блок: и нажмите на него.

2. Как привести робоплатформу в движение?

Шаг 1. Составьте программу, которая изображена ниже.



Шаг 2. Нажмите на клавиатуре клавишу со стрелкой вверх.

Робоплатформа будет ехать одну секунду вперед.

3. Как привести робоплатформу в движение с датчиком касания?

Шаг 1. Подключите датчик касания в слот 4.

Шаг 2. Составьте программу, которая изображена ниже.



Шаг 3. Нажмите стрелку вверх на клавиатуре и нажмите на датчик касания.

При нажатии на датчик касания значение изменяется с 0 до 100.

Когда вы нажмете стрелку вверх на клавиатуре и нажмете на датчик касания, роботоплатформа будет ехать вперед одну секунду.

6. Управление набором расширений робоплатформы

6.1. Описание возможностей набора расширений

Набор расширений расширяет возможности робоплатформы.

В состав набора входят:

- клешня (рисунок 13);
- два датчика цвета RGB (рисунок 14);
- ультразвуковой дальномер (рисунок 15).

Захват-манипулятор “Клешня” предназначен для перемещения и захвата цилиндрических предметов небольшой массы. Хорошо подходят алюминиевые банки из-под газированных напитков.



Рисунок 13 – Клешня

Два датчика цвета RGB: у одного сенсор направлен вперед, у второго - вниз. Датчик излучает свет, рассчитывает хроматичность и насыщенность отраженного луча и сравнивает полученные результаты с ранее заданными значениями таблицы коррекции.

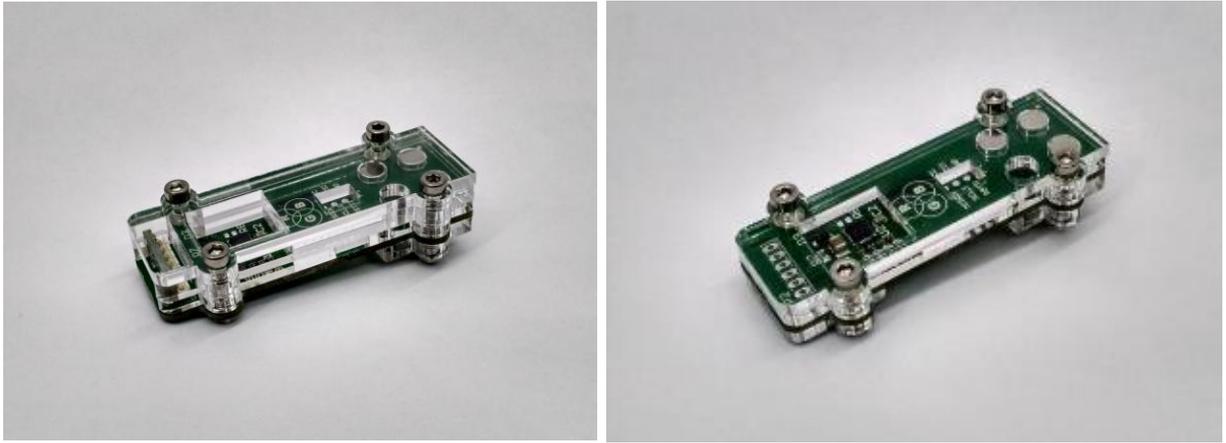


Рисунок 14 – Датчики цвета RGB

Ультразвуковой дальномер предназначен для определения расстояния до препятствий. Рабочий диапазон датчика от 3 до 400 сантиметров.

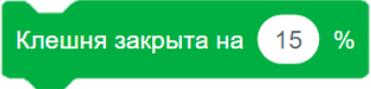


Рисунок 15 – Ультразвуковой дальномер

Значения с датчиков набора расширений показаны в меню робоплатформы (Рисунок 6).

6.2. Команды управления набора расширений робоплатформой

Вид блока	Что блок делает?
	<p>Блок репортер: фиксирует значение.</p> <p>В блоке выбираются значения:</p> <ul style="list-style-type: none"> • номер слота; • цвет. <p>В первом раскрывающемся списке нужно выбрать слот, куда подключен датчик. Доступные слоты: 1 – 5.</p> <p>Доступные цвета: красный, зеленый, синий.</p>

	<p>Логический блок: да/нет.</p> <p>В блоке выбираются значения:</p> <ul style="list-style-type: none"> • номер слота; • цвет. <p>В первом раскрывающем списке нужно выбрать слот, куда подключен датчик. Доступные слоты: 1 – 5.</p> <p>Если датчик обнаруживает заданный цвет, логический блок передает значение "да". В противном случае передается значение "нет".</p> <p>Доступные цвета: красный, мажента (розово-фиолетовый), желтый, зеленый, синий, циан (сине-зелёный), черный, серый, белый, неизвестный.</p>
	<p>Закрывает клешню на заданное количество процентов.</p>
	<p>Устанавливает положение клешни.</p> <p>Доступные варианты: открыта, полуоткрыта, закрыта.</p>

6.3. Подключение набора расширений робоплатформы

Шаг 1. Подключите робоплатформу к RobboScratch по инструкции раздела 7.3.

Когда робоплатформа подключится, кнопка R станет зеленого цвета: .

Шаг 2. В RobboScratch нажмите **Роббо меню > Включить расширенный набор датчиков работа.**

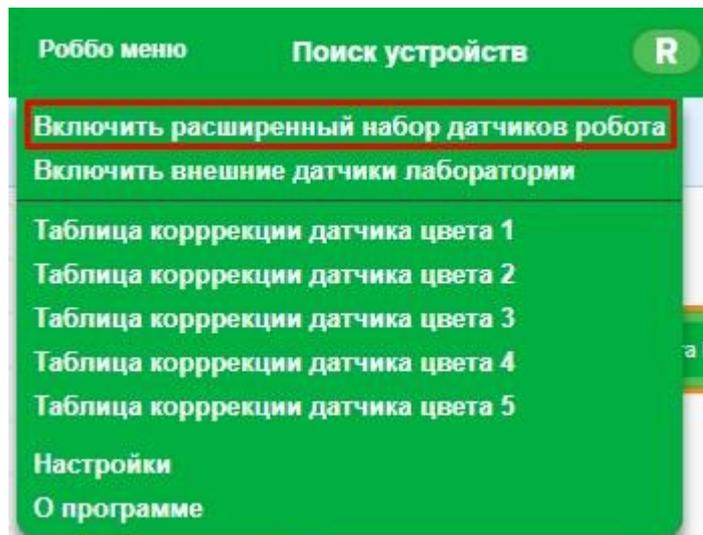


Рисунок 16 – Подключение набора расширений робоплатформы

Шаг 4. Подключите датчики в слоты робоплатформы.

Шаг 5. Нажмите кнопку , чтобы открыть показания датчиков.

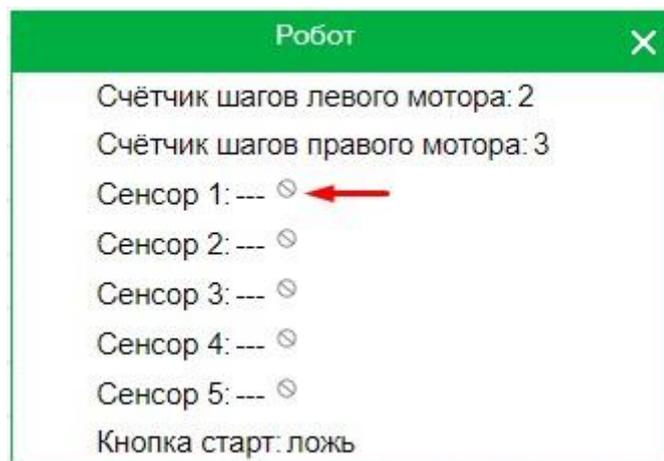


Рисунок 17 – Меню робоплатформы

Шаг 6. Нажмите на  возле сенсора, номер которого соответствует номеру слота робоплатформы.

Ультразвуковой датчик в RobboScratch определяется как: .

Оба датчика цвета RGB определяются как: .

Шаг 7. Установите клешню на головки болтов верхней панели аккумуляторного отсека робота.

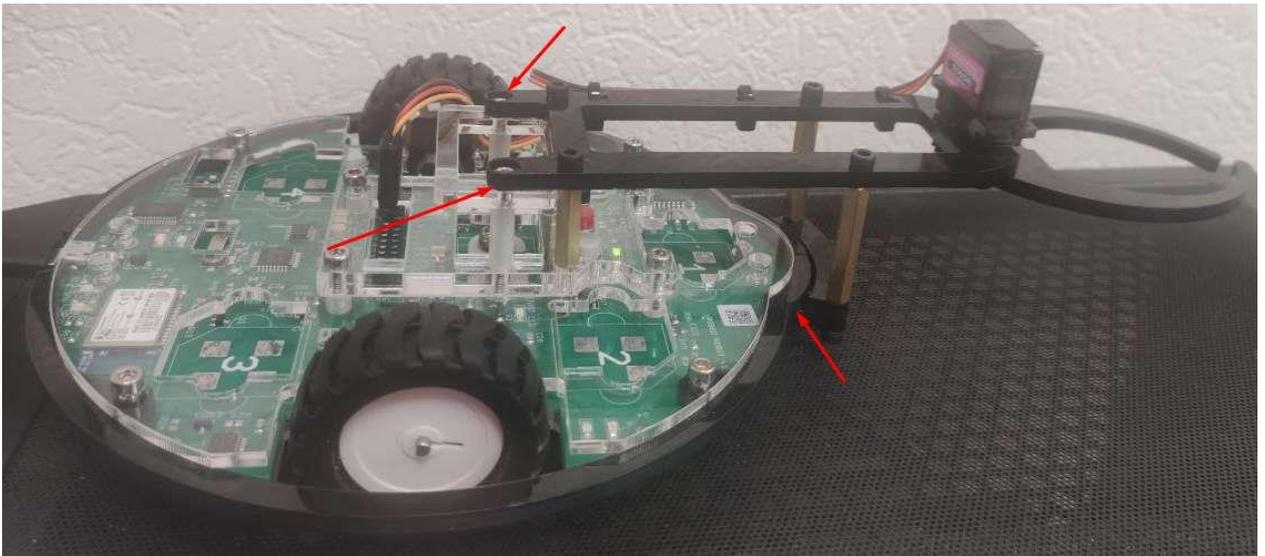


Рисунок 18 – Подключение клешни

Шаг 8. Подключите клешню к дополнительному разъему робота контактам «GND», «5V» и «D13». Контакт GND (заземление) коричневого цвета или промаркирован белой точкой.

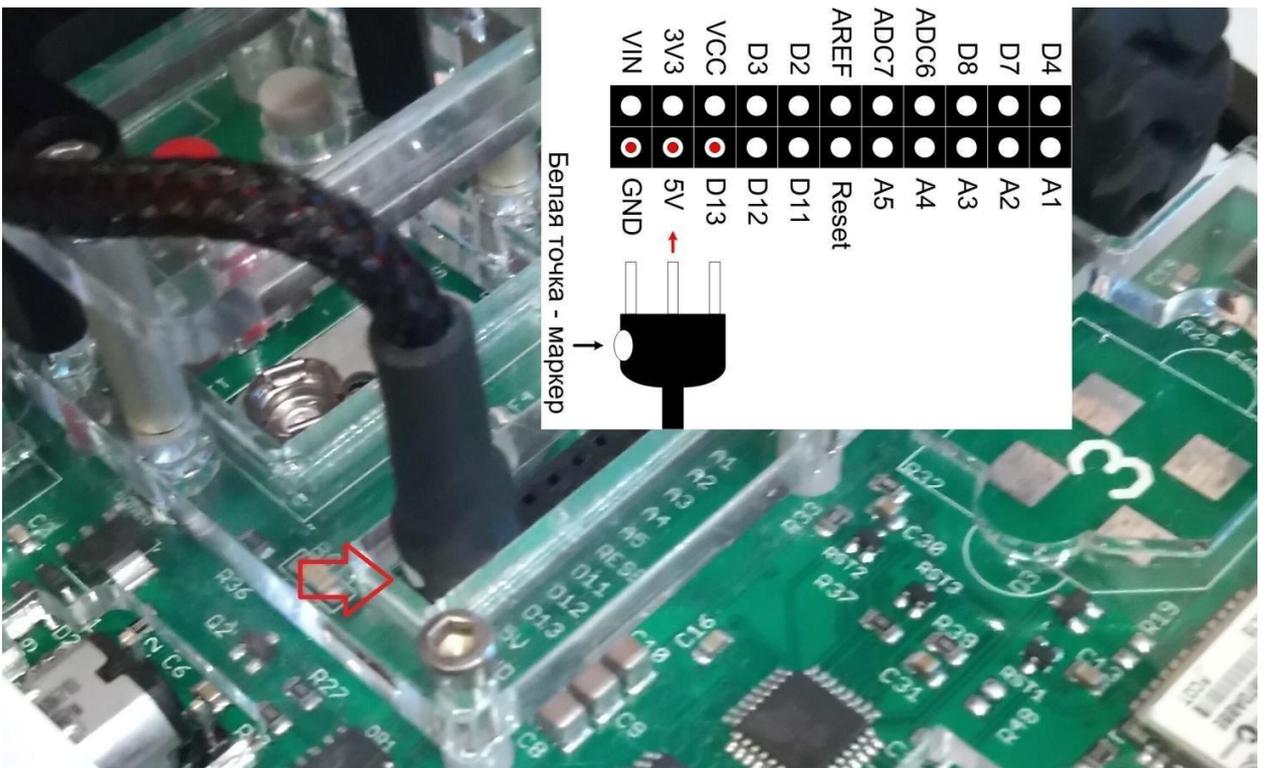


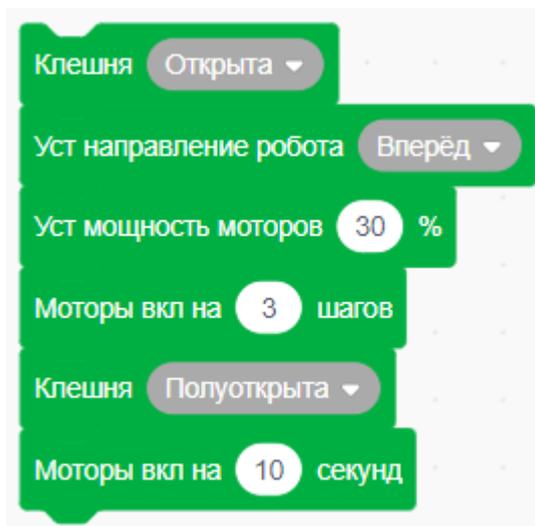
Рисунок 19 – Подключение клешни

Датчики подключены. Для корректной работы нужно настроить датчики согласно инструкции ниже.

6.4. Настройка захвата-манипулятора «Клешня»

Настройка заключается в определении необходимого угла закрытия лепестков клешни для захвата предмета.

Шаг 1. Составьте программу:



Шаг 2. Поставьте на путь робоплатформы какой-то предмет. Например, пустую бутылку 0.5 л или 0.33 л.

Шаг 3. Запустите программу и оцените качество захвата предмета.

Если в положении клешни «полуоткрыта» предмет удерживается захватом, не сжимается слишком плотно, но в то же время позволяет робоплатформе перемещать предмет, не теряя его, то калибровка не требуется. В своих проектах вы можете пользоваться этим оператором.

Если же угол закрытия клешни недостаточный для стабильного удержания предмета, то в проектах нужно использовать оператор с углом закрытия клешни. В этом случае перейдите к шагу 4.

Шаг 4. Для определения необходимого угла закрытия создайте переменную claw.

Шаг 5. Составьте программу ниже.



Шаг 6. Нажмите пробел, чтобы запустить основной алгоритм программы.

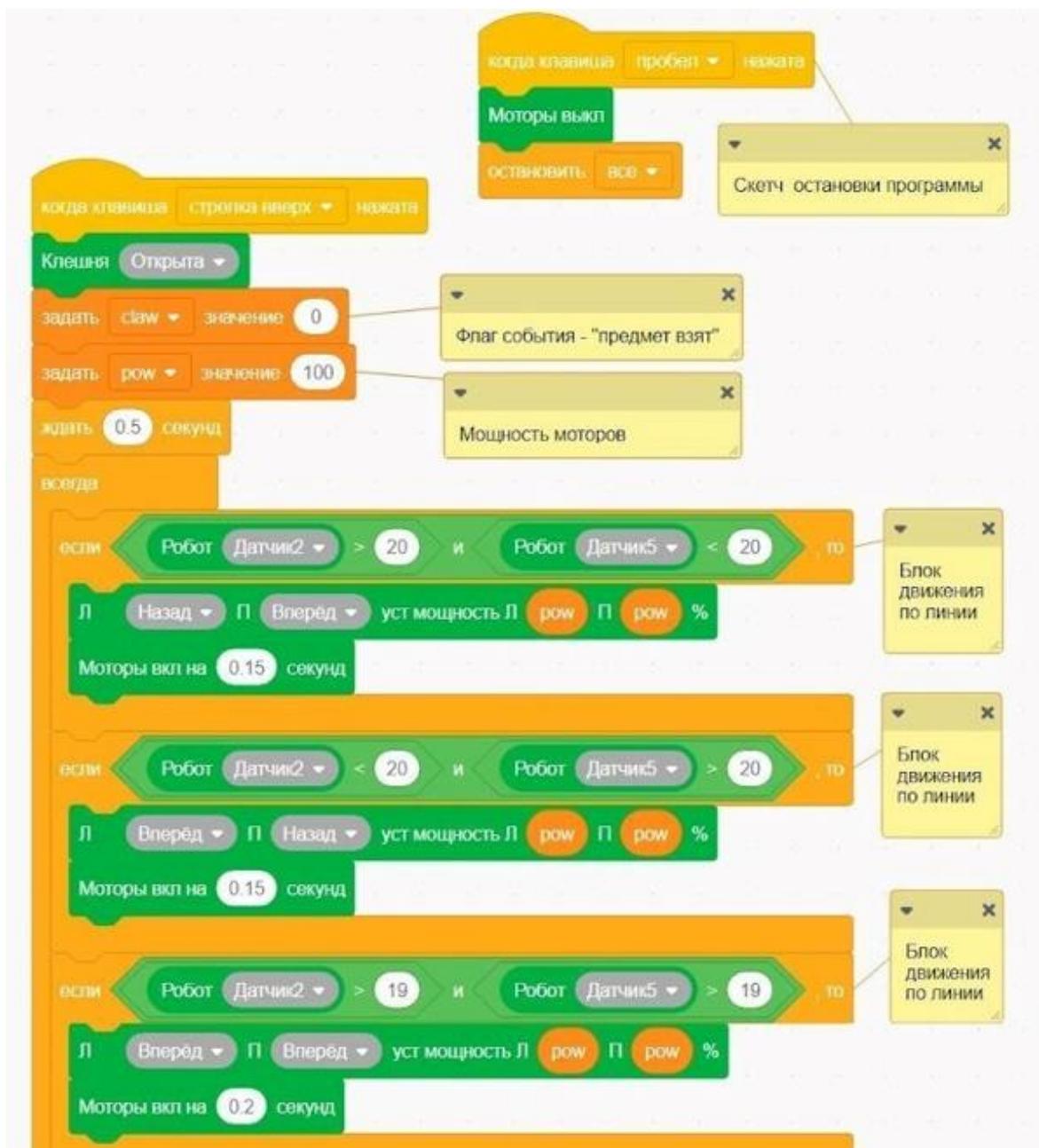
Шаг 7. Нажимайте клавиши вправо и влево на клавиатуре, чтобы закрывать и открывать клешню. Ориентируясь на значение переменной `claw`, подберите необходимый угол закрытия клешни.

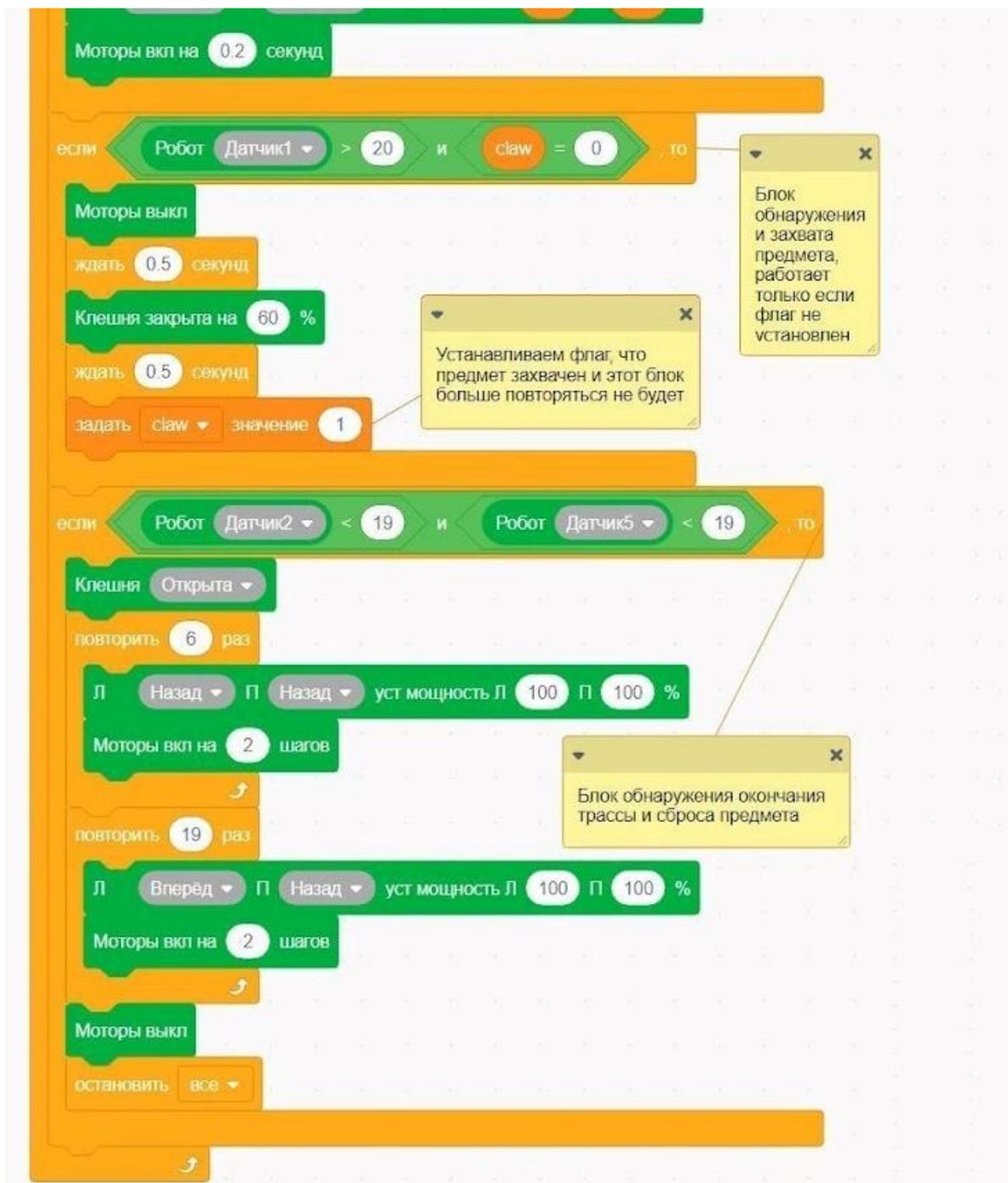
Подобранное значение используйте для своих проектов.

6.5. Программирование «Клешни»

Ниже описано, как создать программу из видео: <https://www.youtube.com/watch?v=5Dmk7osB3Eo&feature=youtu.be>

Подготовьте банку и трассу с черной линией. Подключите к робоплатформе клешню и датчик приближения, выберите эти типы датчиков в RobboScratch. Создайте скетч ниже.





Стрелка вверх запускает движение робота, пробел - останавливает. Программа работает следующим образом: робот-платформа движется по черной линии, обнаруживает на линии банку, хватает её и доставляет её до пункта назначения (до конца черной линии).

6.6. Автокоррекция RGB датчиков

Возможность RGB датчика определять цвета зависит от освещенности в комнате, оттенка цвета, структуры поверхности предмета. Для правильной работы датчика в ваших рабочих условиях нужно настроить таблицу коррекции цветов RobboScratch.

В RobboScratch есть пять таблиц коррекции. Порядковый номер таблицы соответствует номеру слота робоплатформы. То есть, если вы переставите датчик цвета в другой слот, - процедуру коррекции нужно будет делать ещё раз.

Вам потребуется: набор цветных карточек из комплекта и датчик цвета, который смотрит вниз.

Шаг 1. Установите робоплатформу на ровную поверхность. Подключите к слоту 1 датчик цвета, который смотрит вниз. К слоту 3 подключите светодиодный датчик. Выберите нужные типы датчиков в RobboScratch: для RGB , для светодиодного .

Шаг 2. Подложите под датчик белую карточку из комплекта карточек, входящих в состав набора расширений.

Шаг 3. Зайдите в **Роббо меню > Таблица коррекции датчика 1 >** и нажмите **Автокоррекция**.

Пример результата автокоррекции белой карточки.



Рисунок 20 – Автокоррекция

Шаг 3. В открытом окне Color corrector 1 поставьте галочки, чтобы включить дополнительные настройки:

- Show advanced options
- Show maximally advanced options

Шаг 4. Скачайте таблицу коррекции:

<https://drive.google.com/file/d/1c9wRTe8ivlI3OMbF47c16Y-yJZWdpBgo/view>

Шаг 5. В окне **Color corrector 1** нажмите **Загрузить** и **Применить изменения**.

Шаг 6. Поочередно устанавливайте под датчик цветные карточки и смотрите, как определяется цвет в RobboScratch.

На скриншоте ниже определился красный.

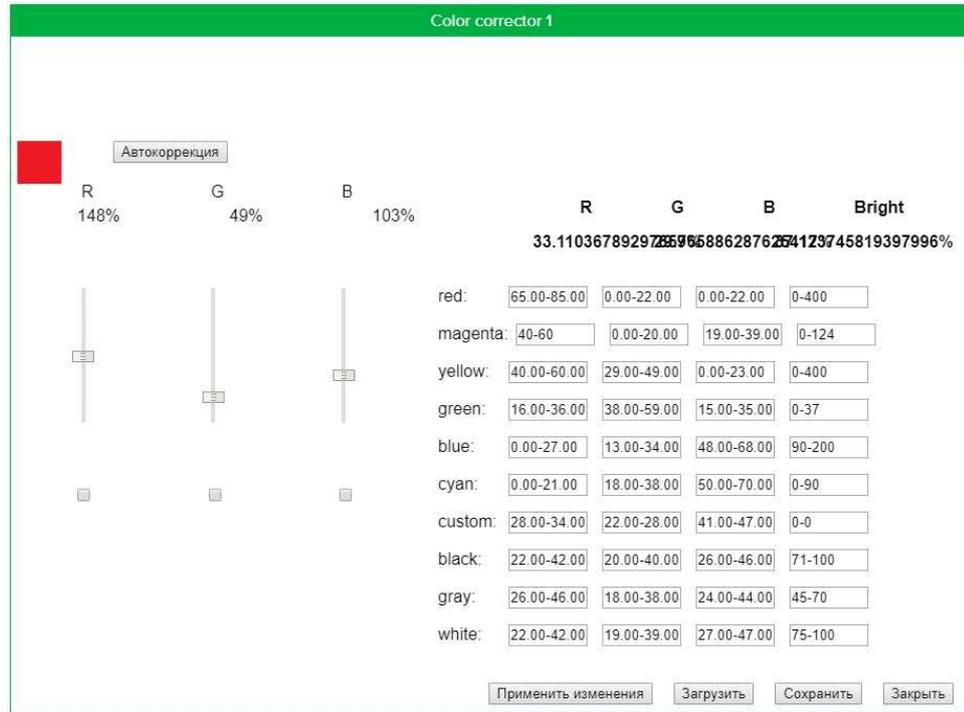


Рисунок 21 – Таблица коррекции

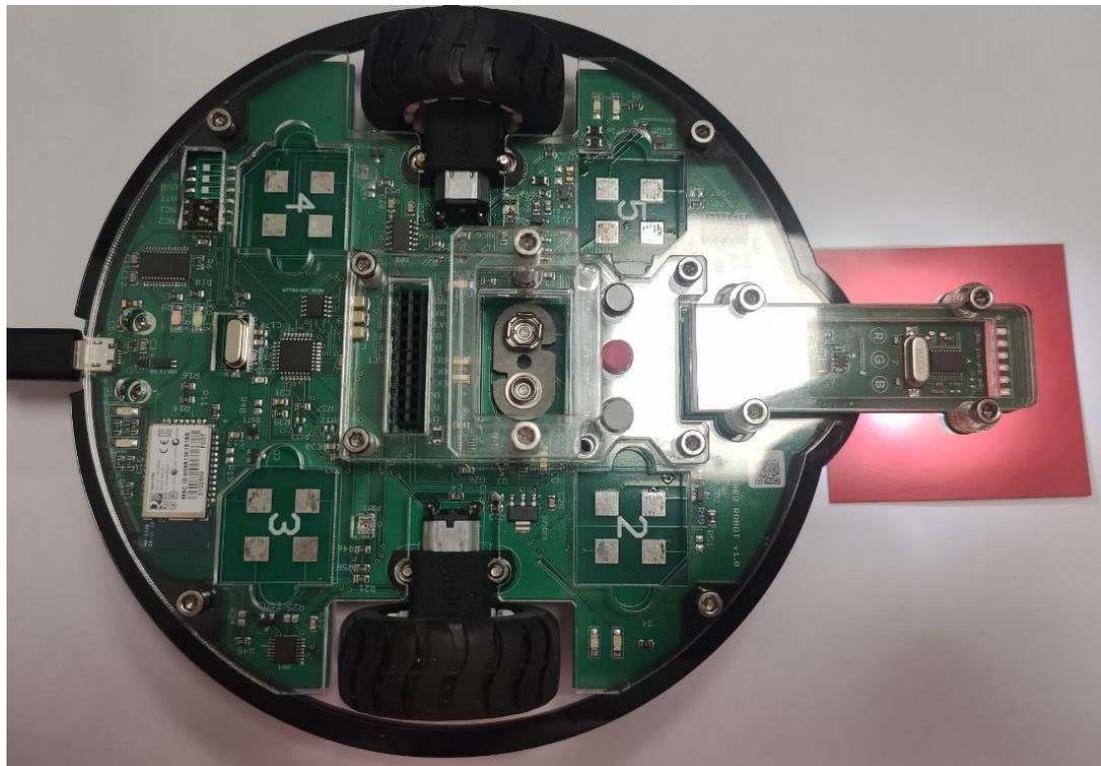


Рисунок 22 – Определение красного цвета

Настройка завершена, если все цвета определяются корректно, то есть, когда вы подносите цветную карточку, RobboScratch высвечивает этот цвет.

Если RobboScratch не отображает цвет карточки, то переходите к следующим шагам.

Шаг 7. Скачайте программу для коррекции таблицы цветов и откройте её в RobboScratch:

https://drive.google.com/file/d/1pYBIOa3p8ubPxO0V6qhR-dj_b_X81-IX/view

Шаг 8. Подложите под робоплатформу карточку, которая некорректно отображается. Переменные %красный, %синий, %зеленый показывают, как датчик цвета определяет текущее количество основных цветов в карточке.



Рисунок 23 – Процентное соотношение цветов

Шаг 9. Сравните значения переменных с диапазонами Таблицы коррекции:

- значение из переменной %красный должно входить в диапазон столбца R таблицы коррекции;
- значение из переменной %зеленый должно входить в диапазон столбца G таблицы коррекции;
- значение из переменной %синий должно входить в диапазон столбца B таблицы коррекции.

Color corrector 1

Show advanced options
 Show maximally advanced options

Автокоррекция

R 148% G 49% B 103%

R 72% G 13.894736842105263157894736%

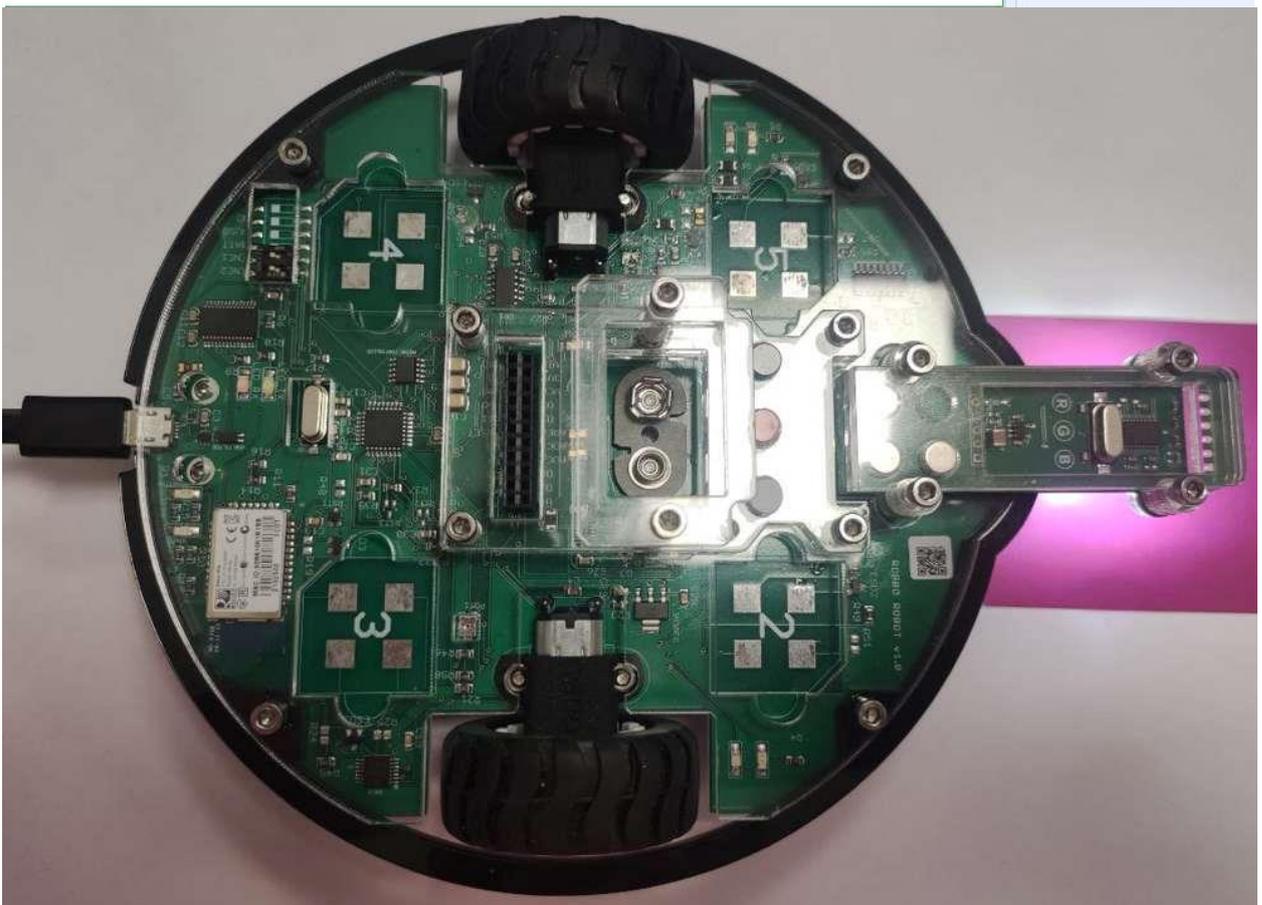
Bright

red:	65.00-85.00	0.00-22.00	0.00-22.00	0-400
magenta:	20.00-40.00	0.00-22.00	19.00-39.00	0-400
yellow:	40.00-60.00	29.00-49.00	0.00-23.00	0-400
green:	16.00-36.00	38.00-59.00	15.00-35.00	0-400
blue:	0.00-27.00	13.00-34.00	48.00-68.00	90-200
cyan:	0.00-21.00	18.00-38.00	50.00-70.00	0-90
custom:	28.00-34.00	22.00-28.00	41.00-47.00	0-0
black:	22.00-42.00	20.00-40.00	26.00-46.00	0-75
gray:	26.00-46.00	18.00-38.00	24.00-44.00	100-400
white:	22.00-42.00	19.00-39.00	27.00-47.00	75-100

color_sum 497
%красный 55.331992
%зеленый 14.688129
%синий 29.979879
corrected_color_sum 0
brightness Infinity

Спрайт: 1
Показать:
Размер: 10

Применить изменения Загрузить Сохранить Закрыть



На скриншоте выше показан пример, когда не определяется пурпурный цвет (magenta). Нужно значения 20.00-40.00 в столбце R изменить так, чтобы 55

(текущее значение переменной %красный) с запасом входило в таблицу коррекции. Нужно поставить в столбце R: 48.00-68.00.

Шаг 10. Если, несмотря на предыдущий шаг, цвет не определяется в RobboScratch, нужно изменить значение в столбце bright в соответствии со значением переменной brightness.

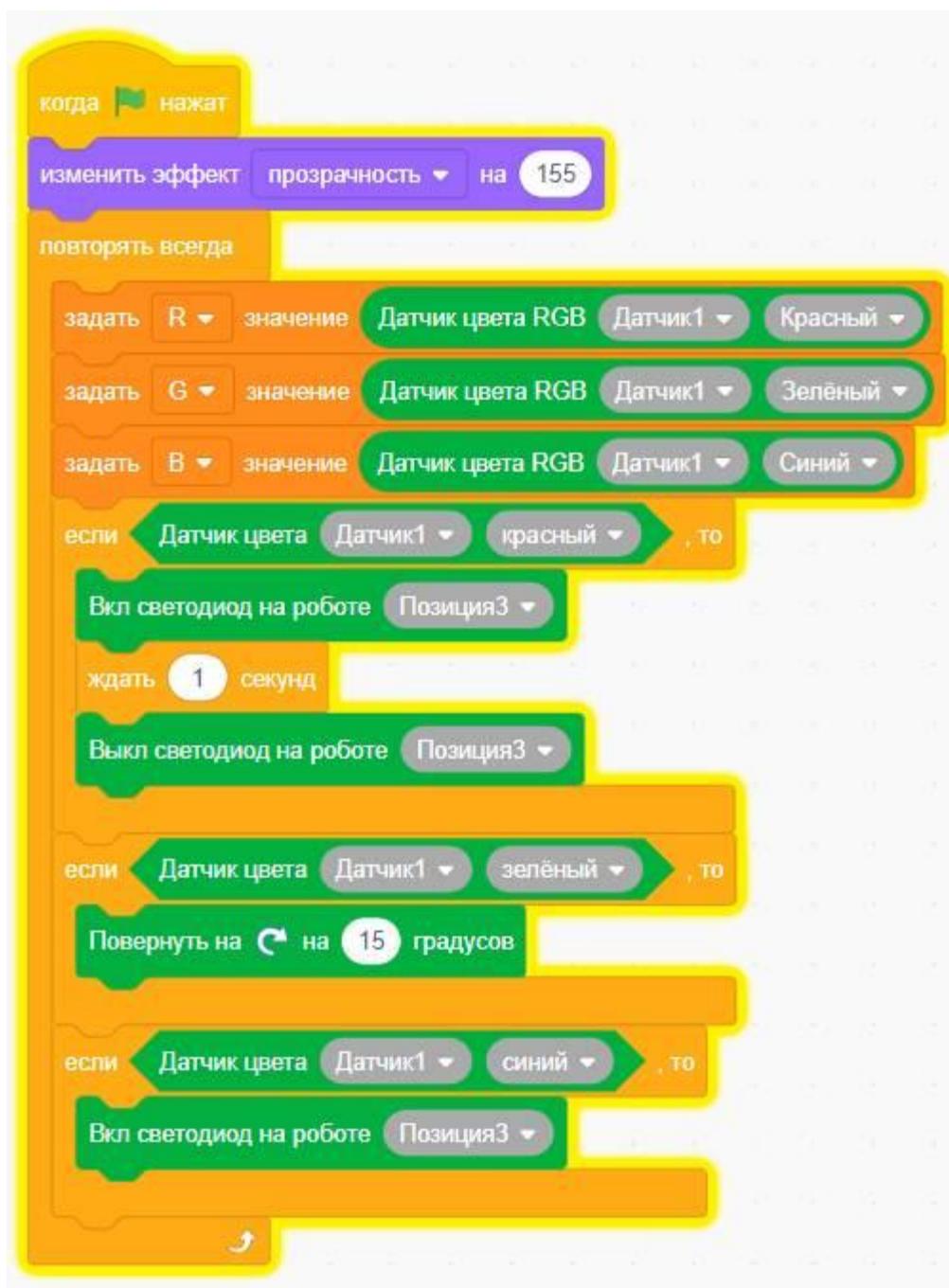
Вы можете видеть, что переменная brightness задается условно, через сумму всех значений, и корректируется коэффициентом. Поэтому нужно начинать с коррекции столбцов R, G и B. Коррекция столбца bright актуальна в основном для белого, черного и серого цветов.

Шаг 11. После того, как вы настроите все цвета, сохраните таблицу с расширением *.json. В следующие разы вы сможете воспользоваться этой таблицей.

Готово! Теперь в порт 1 можно устанавливать оба RGB датчика и создавать свои проекты по определению цветов и сортировки объектов. Для использования других слотов робоплатформы настройку нужно повторить. Для этого нужно настроить таблицы коррекции 2...5.

6.7. Программирование RGB датчиков цвета

Повторите программу ниже, чтобы научиться управлять датчиком цвета.



В слот 1 вставьте датчик цвета RGB, в слот 3 вставьте светодиодный датчик.

Чтобы протестировать программу, нажмите на флажок и подкладывайте под датчик цветные карточки.

Когда RobboScratch обнаружит определенный цвет, роботоплатформа сделает одно из действий:

- белый цвет: персонаж-робот в окне программы не отображается;

- красный цвет: светодиодный датчик включается и выключается через одну секунду;
- зеленый цвет: роботплатформа поворачивается на 15 градусов;
- синий цвет: светодиодный датчик включается.

6.8. Программирование ультразвукового датчика

Ультразвуковой датчик не нужно предварительно настраивать. Чтобы научиться управлять датчиком, составьте программу по Рисунок 25.

Используйте такую трассу:



Рисунок 24

The screenshot shows a programming environment with a script for a robot's movement. The script starts with a 'нажата' (clicked) event block. It then sets a 'flag' variable to 0 and a 'row' variable to 30. A 'повторять всегда' (repeat forever) loop contains several conditional blocks:

- Initialization:** 'right' sensor is set to 'Робот Датчик2' and 'left' sensor is set to 'Робот Датчик5'.
- Forward Movement:** An 'если' (if) block checks if both 'right' > 20 and 'left' > 20. If true, it sets motor power to 'row' and turns motors on for 3 steps.
- Left Turn:** An 'если' block checks if 'right' > 20 or 'right' = 20 and 'left' < 20 or 'left' = 20. If true, it sets motor power to 'row' and turns motors on for 1 step in reverse.
- Right Turn:** An 'если' block checks if 'right' < 20 or 'right' = 20 and 'left' > 20 or 'left' = 20. If true, it sets motor power to 'row' and turns motors on for 1 step in reverse.
- Intersection:** An 'если' block checks if 'right' < 20 and 'left' < 20. If true, it indicates reaching the intersection.

Annotations in yellow boxes explain the logic:

- 'Нажанием кнопки "стрелка вверх" запускаем программу' (Clicking the 'up arrow' button starts the program).
- 'Эта переменная понадобится, чтобы знать, поворачивать ли на следующем перекрестке' (This variable will be needed to know whether to turn at the next intersection).
- 'Переменная для установки мощности моторов' (Variable for setting motor power).
- 'Получаем данные с датчиков линии' (Get data from line sensors).
- 'Блок движения по черной линии. Проезжаем прямо' (Block of movement along the black line. Drive straight).
- 'Блок движения по черной линии. Подворачиваем влево.' (Block of movement along the black line. Turn left).
- 'Блок движения по черной линии. Подворачиваем влево.' (Block of movement along the black line. Turn left).
- 'Если доехали до перекрестка' (If reached the intersection).

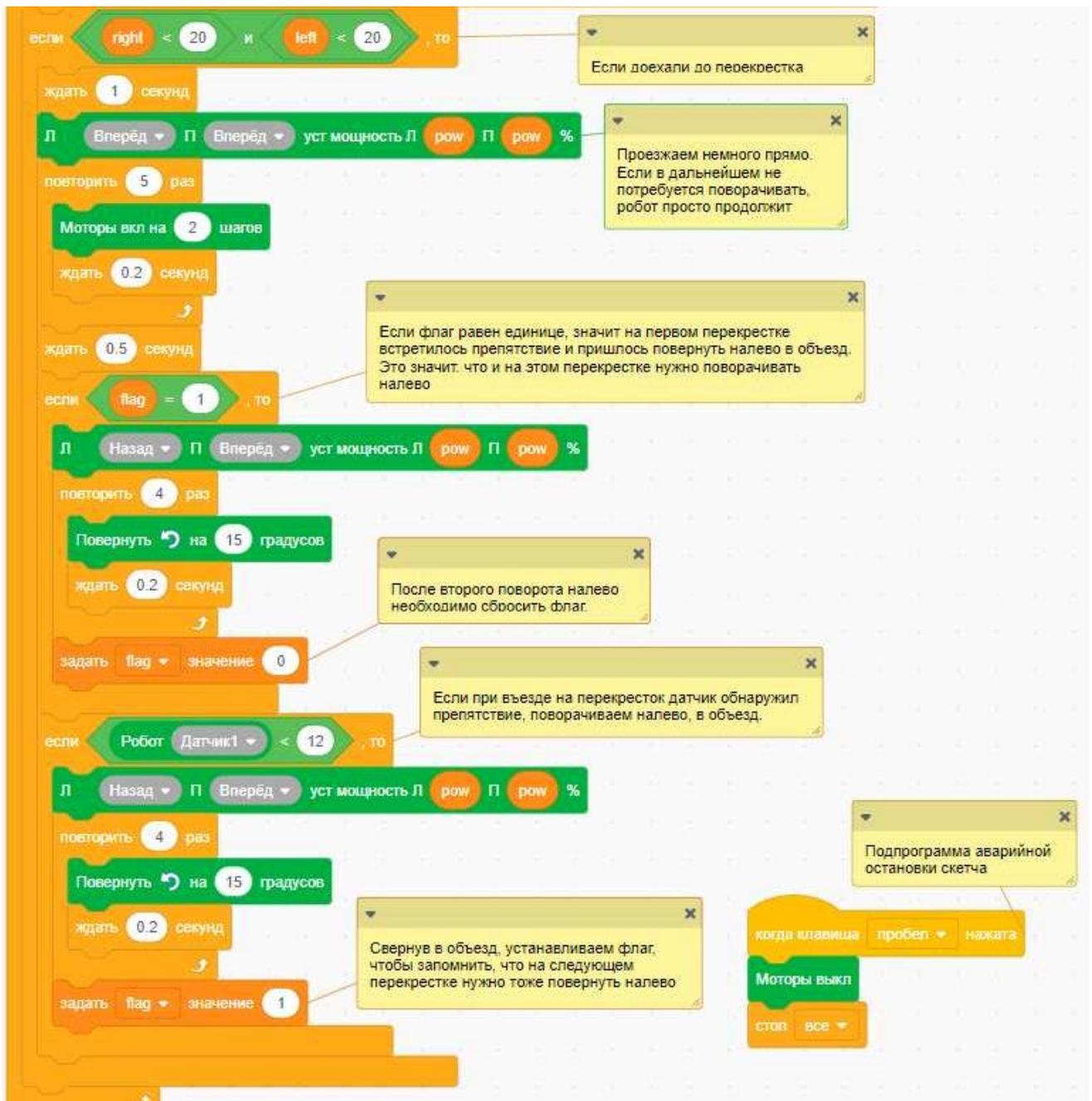


Рисунок 25 – Программирование ультразвукового датчика

Готовую программу можно скачать по ссылке:

<https://drive.google.com/file/d/1u-PywiSOK5xt1o8QD8Roifm1QX2FgGLb/view>.

Результат работы программы можно увидеть в видео:

<https://www.youtube.com/watch?v=kT36J83mnm&feature=youtu.be>.

Программа работает по такому принципу: робот движется по черной линии кольцевой трассы с вариантом объезда. В случае, если нет препятствия, робот будет проезжать по кольцевой трассе. Если будет установлено препятствие, робот должен проехать в объезд.

7. Лаборатория

7.1. Описание возможностей лаборатории

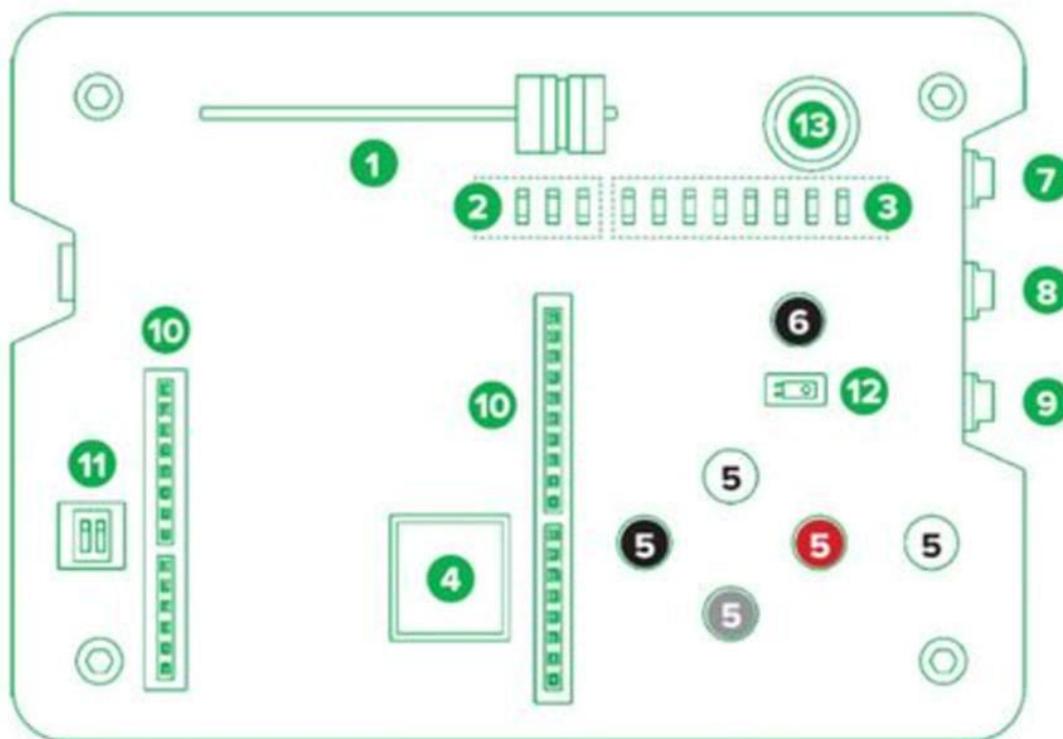


Рисунок 26 – Лаборатория

Описание частей и элементов управления лаборатории (номера в списке соответствуют номерам на схеме).

1. Рычажок – это резистор с подвижным отводным контактом, передает значения от 0 до 100.
2. Три светодиода: красный, желтый, зеленый.
3. Управляемая матрица из 8 светодиодов.
4. Динамик - воспроизводит звук разной высоты.
5. Пять кнопок управления.
6. Кнопка сброса.
7. Разъем цифрового ввода/вывода D13 - порт для чтения и записи цифровых данных 0/1.
8. Разъем аналогового входа A1 – это порт для чтения аналоговых значений от нуля до ста, можно подключить дискретный датчик;
9. Разъем аналогового входа AO – это порт для чтения аналоговых значений от нуля до ста, можно подключить дискретный датчик;
10. Разъемы для подключения Arduino Uno.

11. DIP-переключатели:

- положение 1-ON отключает элементы лаборатории №1, №3, №4, №12, №13, при этом лабораторию можно использовать в качестве платформы Arduino Uno.
- положение 1-OFF включает все элементы лаборатории, часть разъемов №10 подключены к собственным компонентам лаборатории;
- положение 2-ON отключает USB-контроллер, при этом лабораторию можно использовать в качестве платформы Arduino Uno.
- положение 2-OFF обеспечивает последовательное соединение через USB-кабель.

Во время программирования держите переключатель в положении 2-OFF.

12. Датчик света – это фототранзистор, реагирует на свет.

13. Датчик звука – это микрофон, реагирует на звук и шум.

Значения с датчиков и кнопок показаны в меню лаборатории (Рисунок 27 Рисунок 6).

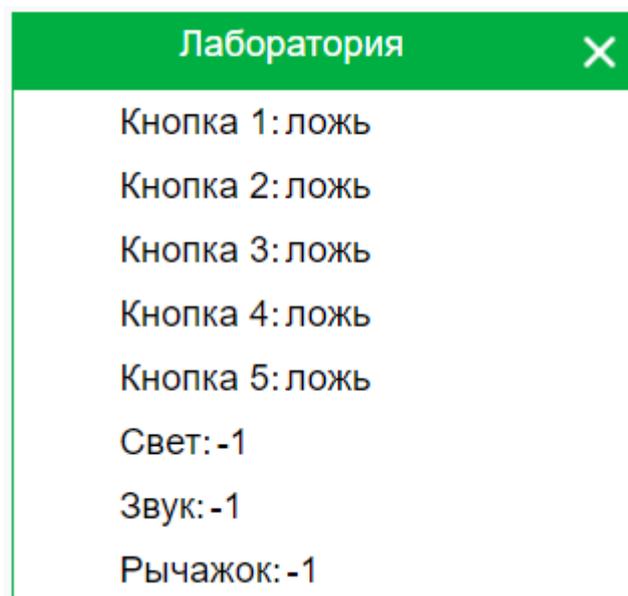
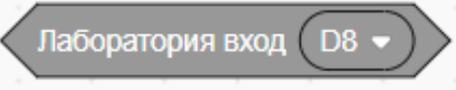


Рисунок 27 - Меню лаборатории

7.2. Команды управления лабораторией

Вид блока	Что блок делает?
	Включает выбранный светодиод на лаборатории. Доступные номера светодиодов: 0 – 7.
	Выключает выбранный светодиод на лаборатории. Доступные номера светодиодов: 0 – 7.
	Включает выбранный цветной светодиод на лаборатории. Доступные светодиоды: красный, желтый и зеленый.
	Выключает выбранный цветной светодиод на лаборатории. Доступные светодиоды: красный, желтый и зеленый.
	Воспроизводит звук из динамика выбранной тональности.
	Передает значения с выбранного датчика. Доступные варианты: свет, звук, рычажок.
	Проверяет, нажата ли выбранная кнопка на лаборатории. Передает значения: да / нет. Доступные варианты: 1 – 5.
	Передает значения с выбранного аналогового входа. Доступные варианты: A0 – A6.

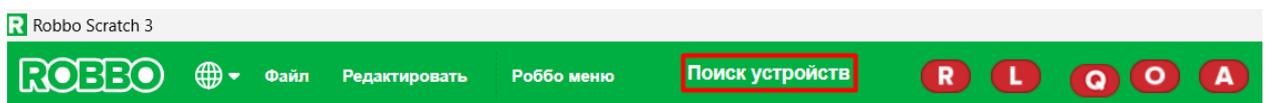
	<p>Передает значения да / нет с выбранного дискретного входа.</p> <p>Доступные варианты: D8 – D13.</p>
	<p>Включает или выключает выбранный дискретный выход лаборатории.</p> <p>Доступные варианты: D8 – D13.</p>
	<p>Устанавливает выбранный дискретный выход лаборатории в назначенное ШИМ значение.</p>

7.3. Подключение и проверка лабораторией в RobboScratch

Шаг 1. Подключите Лабораторию к своему компьютеру через USB.

При подключении прозвучит звуковой сигнал, промигают светодиоды матрицы и засветит зеленый светодиод.

Шаг 2. Откройте RobboScratch и нажмите **Поиск устройств**.



Может открыться окно с перечнем всех устройств, подключенных к компьютеру по USB. В этом случае нажмите Подключение.

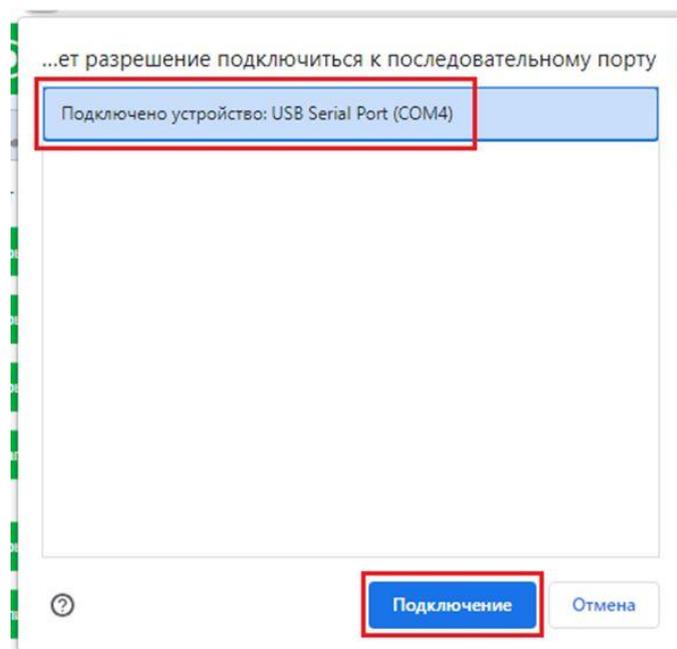


Рисунок 28 – Подключение лаборатории

Примечание. Программное обеспечение на устройствах автоматически обновляется. Соглашайтесь на обновление прошивки: это позволит устройству стабильно работать.

Когда Лаборатория подключится, прозвучит звуковой сигнал и кнопка L станет зеленого цвета:  > .

Шаг 3. Нажмите кнопку .

Отобразится меню показаний датчиков, кнопок и рычажка.

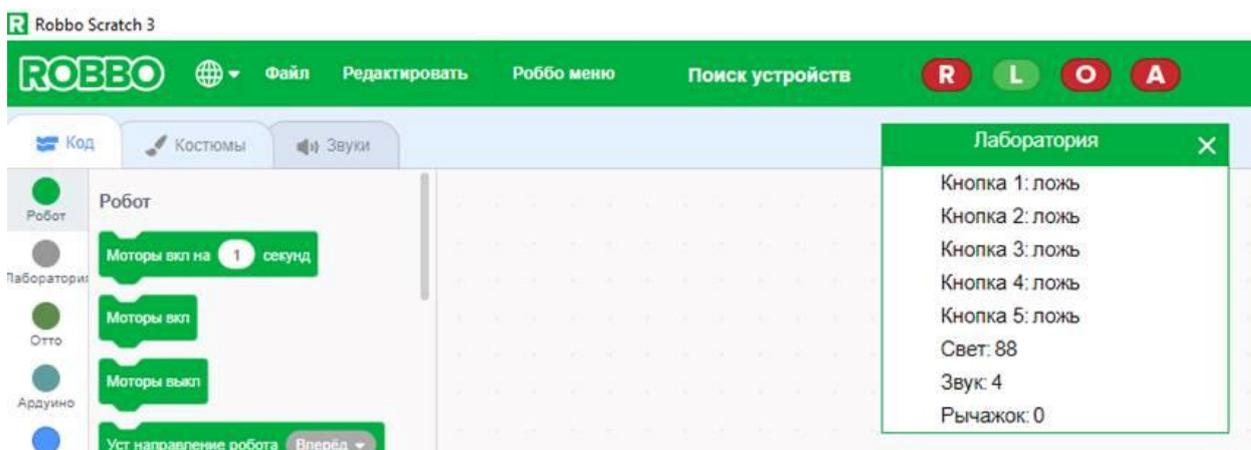


Рисунок 29 – Проверка датчиков лаборатории

Проверьте как изменяются значения элементов взаимодействия:

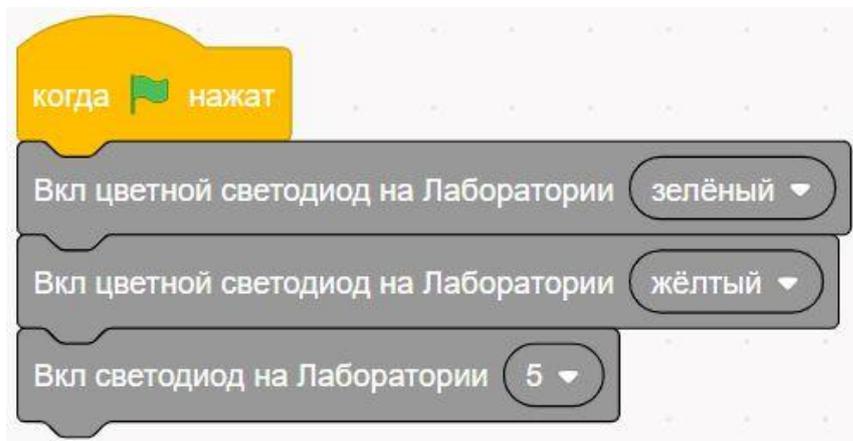
- передвиньте рычажок – значения в поле показаний изменится в диапазоне от 0 до 100;
- воспроизведите шум рядом с датчиком звука – значение увеличится;
- проверьте датчик света: если поднести его к свету, например, к включенной лампе – значения в строке показания увеличатся, если закрыть датчик – значения уменьшатся;
- нажав черную обособленную кнопку у датчика света, вы отключите лабораторию от компьютера, при этом светодиоды на матрице поочередно включатся и выключатся. Для повторного подключения повторите шаг 2;
- при нажатии на одну из пяти кнопок значение в программе изменится с Ложь на Правда. Нажимать можно одну или сразу несколько кнопок.

Шаг 4. Чтобы проверить светодиоды, создайте программу в RobboScratch.

- Используйте одну из команд запуска из бокового меню  События.

- Используйте одноименные команды запуска светодиодов из бокового меню **Лаборатория**. У команд есть выпадающее меню настройки, выберите номер светодиода (от 0 до 7) или цвет (зеленый, жёлтый, красный).

Пример программы проверки светодиодов:



Результат работы программы:

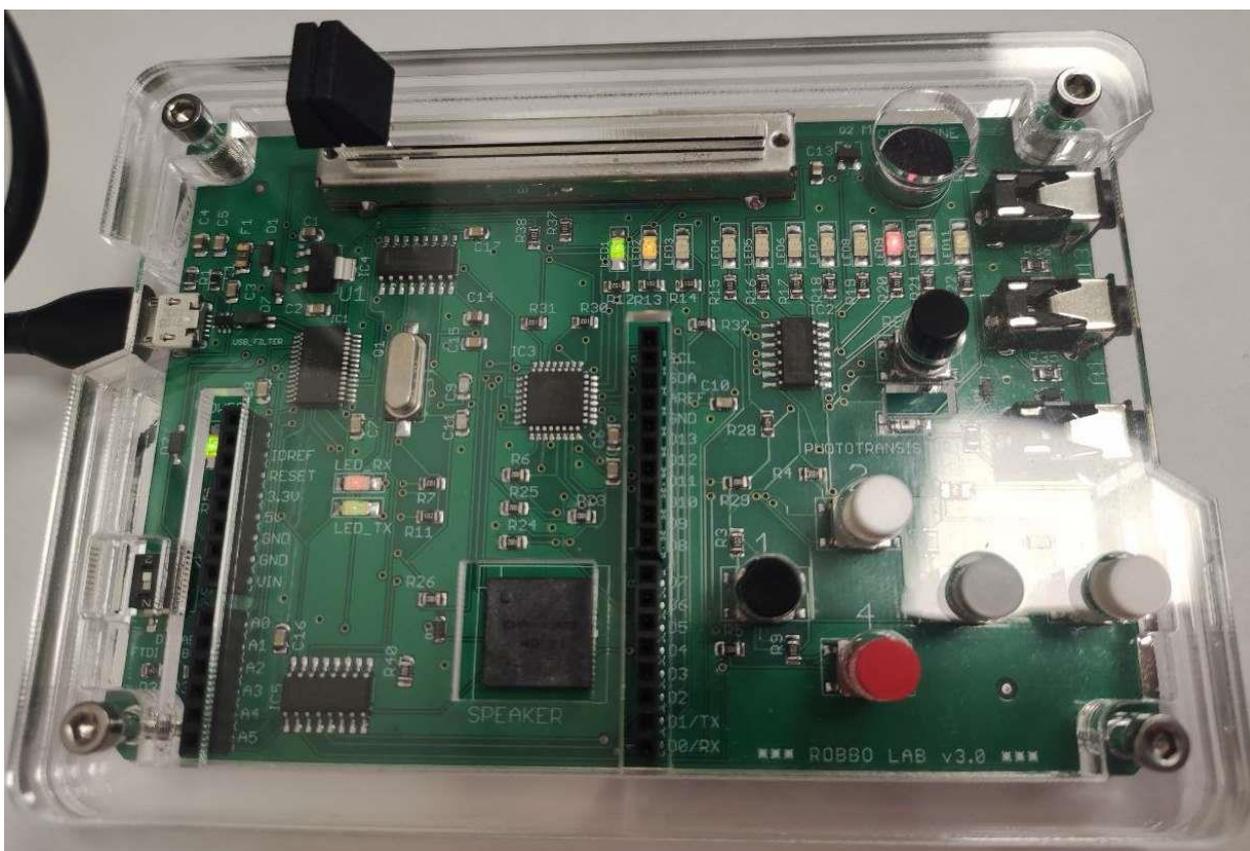
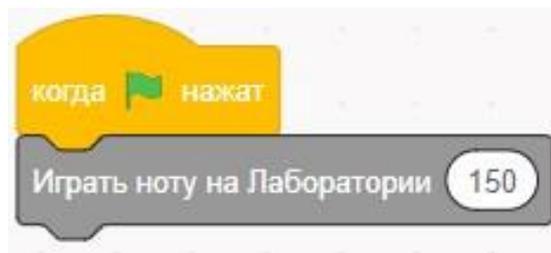


Рисунок 30 – Запуск программы лаборатории

Шаг 5. Чтобы проверить динамик лаборатории, создайте программу в RobboScratch.



Результат: Лаборатория воспроизведет короткий звуковой сигнал.

7.4. Программирование лаборатории в RobboScratch

Лабораторию можно использовать в качестве контроллера для управления другими электронными устройствами. Для этого нужно, чтобы все устройства были подключены (по USB или Bluetooth) и определены в RobboScratch (или другой среде программирования).

Ниже пример программы, где роботоплатформа управляется через Лабораторию.



Описание работы программы: если рычажок лаборатории сдвинут на значение больше 5, то роботоплатформа проедет вперед, после этого из динамика Лаборатории прозвучит звук и загорятся светодиоды.

8. Набор расширений лаборатории

8.1. Описание возможностей набора расширений лаборатории

Набор расширений лаборатории позволяют решать дополнительные задачи:

- измерение температуры окружающей среды или объекта;
- измерение параметров электрической цепи.

В состав набора расширений лаборатории входят (Рисунок 31):

- три крокодила – это кабели с зажимами с двух сторон, они нужны для измерения напряжения. Зажимы подключаются к измеряемому предмету и к пустому датчику;
- датчик для измерения напряжения – это пустой датчик с выходами 5V, IN и GND для подключения крокодилов. Подключается к лаборатории через разъем 3.5 Jack;
- датчик для измерения температуры – это изолированная плата. Подключается к лаборатории через разъем 3.5 Jack.

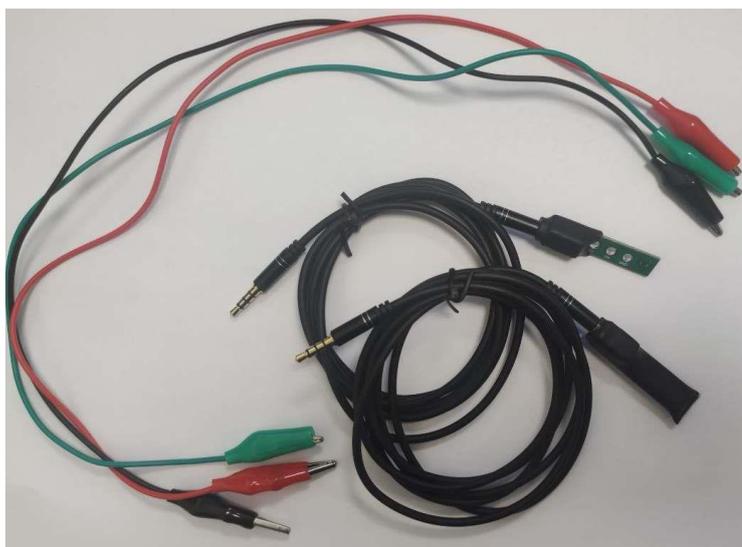


Рисунок 31 - Набор расширений лаборатории

Плата датчика температуры изолирована, чтобы датчик можно было опускать в жидкости. Но будьте осторожны: не допускайте попадания жидкости в разъем AUX, мочить можно только изолированную часть датчика.

Погрешность датчика напряжения составляет 15%. В RobboScratch 3 напряжение определяется в условных единицах: 5 Вольт = 100.

Значения с датчиков показаны в меню набора расширений лаборатории (Рисунок 32). К цифровому порту можно подключить датчик напряжения, к аналоговому – датчик напряжения или датчик температуры.

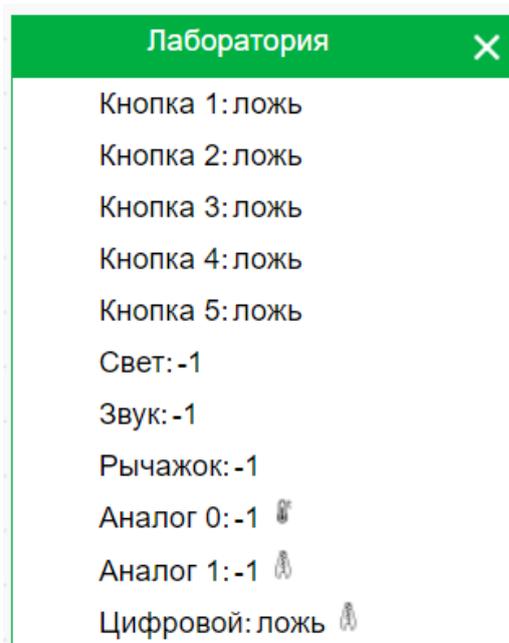


Рисунок 32 - Меню показаний набора расширений лаборатории

8.2. Команды управления набором расширений лаборатории

Вид блока	Что блок делает?
	<p>Передает значения с выбранного аналогового входа.</p> <p>Доступные варианты: A0 – A5.</p>
	<p>Проверяет, передаются ли показания с выбранного дискретного входа. Передает значения: да / нет.</p> <p>Доступные варианты: D8 – D13.</p>

8.3. Программирование набора расширений лаборатории

Шаг 1. Проверьте, что лаборатория подключена к RobboScratch:

- на лаборатории горит зеленый светодиод;
- кнопка зеленого цвета.

В разделе 7.3. Подключение и проверка лабораторией в RobboScratch описано, как подключить лабораторию.

Шаг 2. В RobboScratch нажмите **Роббо меню > Включить внешние датчики лаборатории.**

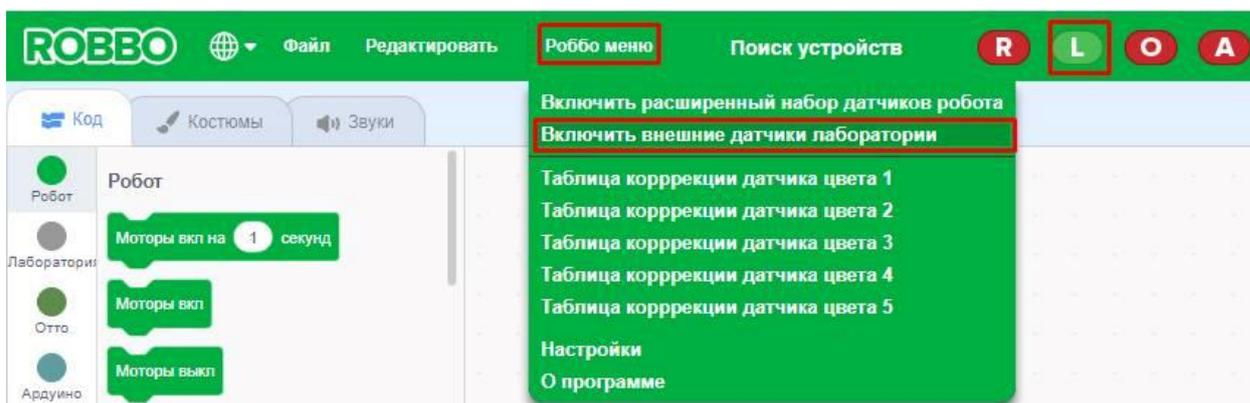


Рисунок 33 – Подключение набора расширений

В результате в меню показаний добавятся три новых показания.

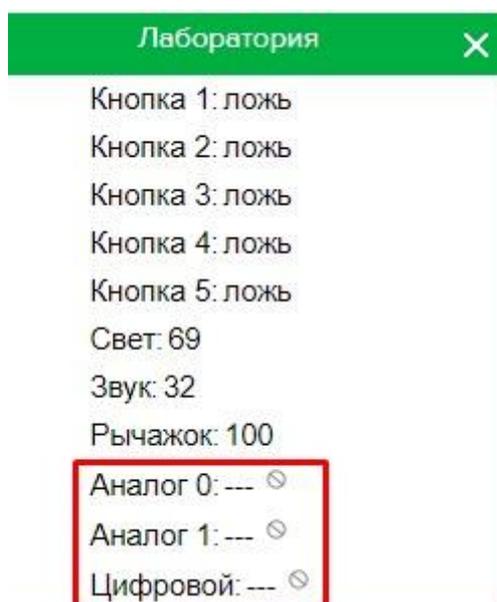


Рисунок 34 – Меню датчиков лаборатории

Шаг 3. Подключите датчик температуры к входу лаборатории А0 или А1.

На рисунке показано подключение датчика температуры к порту А0 лаборатории.

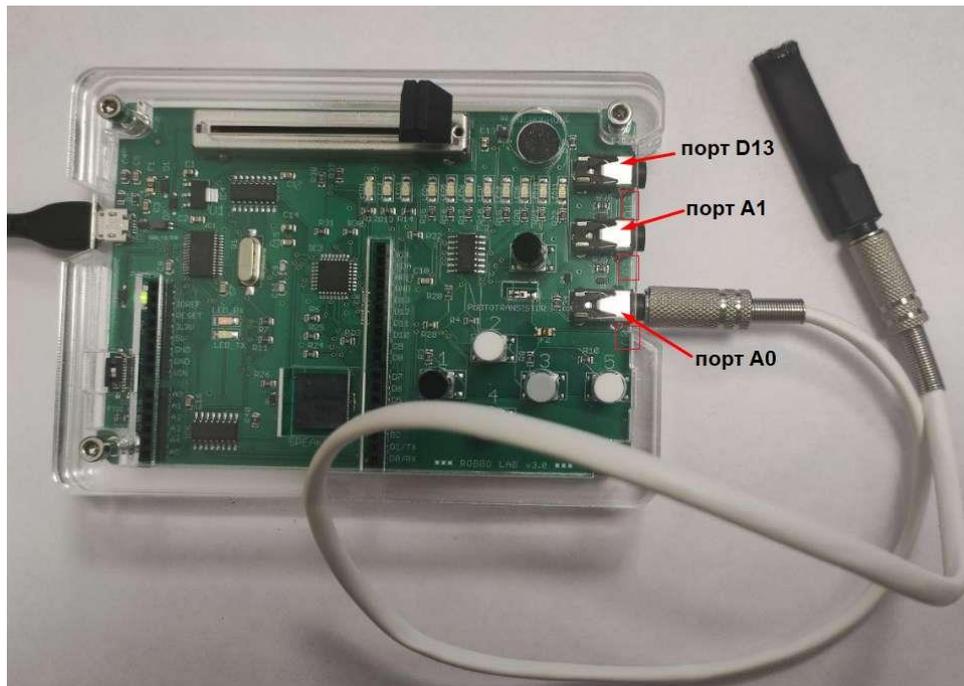
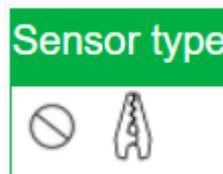


Рисунок 35 – Набор расширений подключен к лаборатории

Шаг 4. В RobboScratch выберите тот же порт и в нем нажмите , потом выберите изображение датчика в поле Sensor type:



На рисунке в поле Аналог 0 выбран датчик температуры, физически датчик также подключен к A0. Температура определяется как 20 градусов Цельсия.

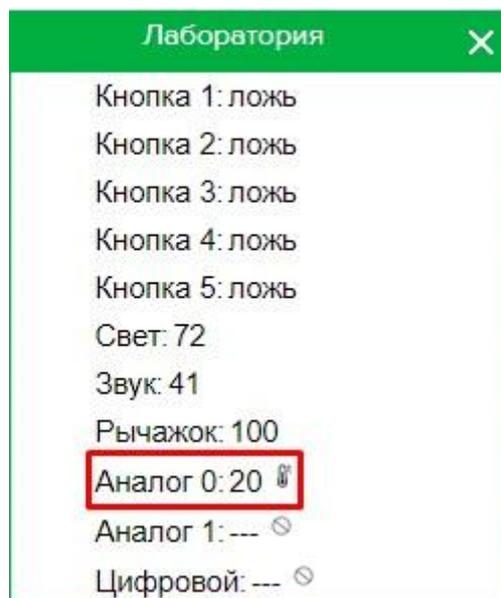


Рисунок 36 – Измерение температуры

Шаг 5. Подключите датчик напряжения к аналоговому порту A0 или A1 лаборатории, определите тот же порт в RobboScratch.

Датчик имеет выводы для крокодилов:

- 5V – выходной пин от стабилизатора напряжения с выходом 5В и максимальным током 1А. Используйте только для подключения дополнительных устройств;
- IN – выходной пин для вывода входного сигнала ("плюс");
- GND – вывод для подключения земли ("минус").

Шаг 6. Подключите крокодилы к выходам IN и GND на плате датчика.

На фото красный провод - IN, зеленый - GND.

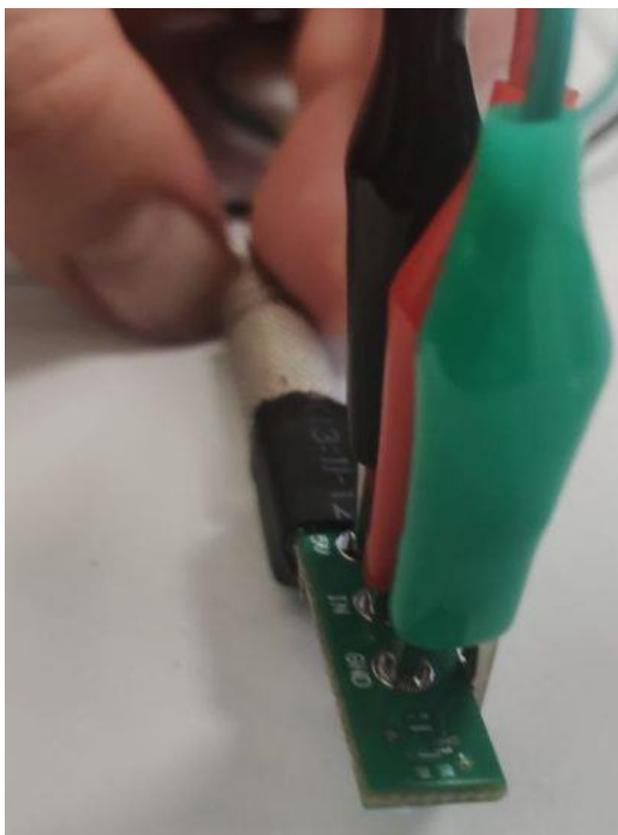


Рисунок 37 – Подключение крокодила

Шаг 7. Кабель от IN нужно подключить к + батарейки, кабель от GND к – батарейки.



Рисунок 38 – Измерение напряжения

В строке показаний RobboScratch датчик определяет 34. Если перевести из условных единиц в вольты, получается 1.7В. Измерения верны: производитель батареек заявляет напряжение 1.5В, погрешность датчика 15%.

9. Управление квадрокоптером

9.1. Описание возможностей квадрокоптера

РОББО Программируемый квадрокоптер – это беспилотный летательный аппарат с открытым исходным кодом. Программирование квадрокоптера развивает навыки управления, пилотирования и маневрирования в реальных условиях.

Квадрокоптер весит 27 граммов, его размер 92x92x29 мм. Конструкция оснащена 3-осевым гироскопом, 3-осевым акселерометром, 3-осевым магнитометром и высокоточный датчиком давления. Время полета квадрокоптера с аккумуляторной батареей 7 минут. Квадрокоптер ет переносить дополнительный вес 15 граммов.

В комплект входит: квадрокоптер, Bluetooth адаптер, аккумуляторная батарея, пропеллеры.

9.2. Техника безопасности при полетах квадрокоптера

1. Летайте только под присмотром учителя или с его разрешения.
2. Не начинайте полет при подключенном проводе USB.
3. Перед каждым полетом проверяйте целостность пропеллеров. Полет с гнутым или разрушенным пропеллером может разрушить квадрокоптер полностью.
4. Квадрокоптер можно запускать только в помещении.
5. Перед полетом убедитесь, что над квадрокоптером нет препятствий: ламп, проводов, тросов.
6. Завершайте программу, как только поняли, что квадрокоптер потерял управление.
7. Все присутствующие в помещении люди должны находиться на расстоянии не менее 2 метров от квадрокоптера. Учитывайте возникновение нештатных ситуаций и отклонение квадрокоптера от намеченного курса.
8. Между вами и квадрокоптером не должно находиться ни одного человека.
9. Если квадрокоптер упал на бок, но еще работает - выключите его и только потом поднимите.
10. Переподключайте питание квадрокоптера перед каждым полетом. Для этого нужно нажать на кнопку, расположенную возле аккумулятора.

Значения текущих координат квадрокоптера, угол поворота и уровень заряда батареи показаны в меню квадрокоптера (Рисунок 39).

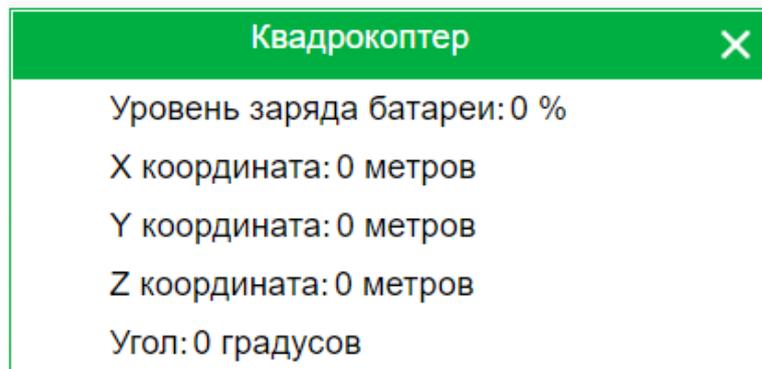


Рисунок 39 – Меню квадрокоптера

9.3. Состояние квадрокоптера по светодиодам

Пропеллеры квадрокоптера должны быть подключены как на Рисунок 40.

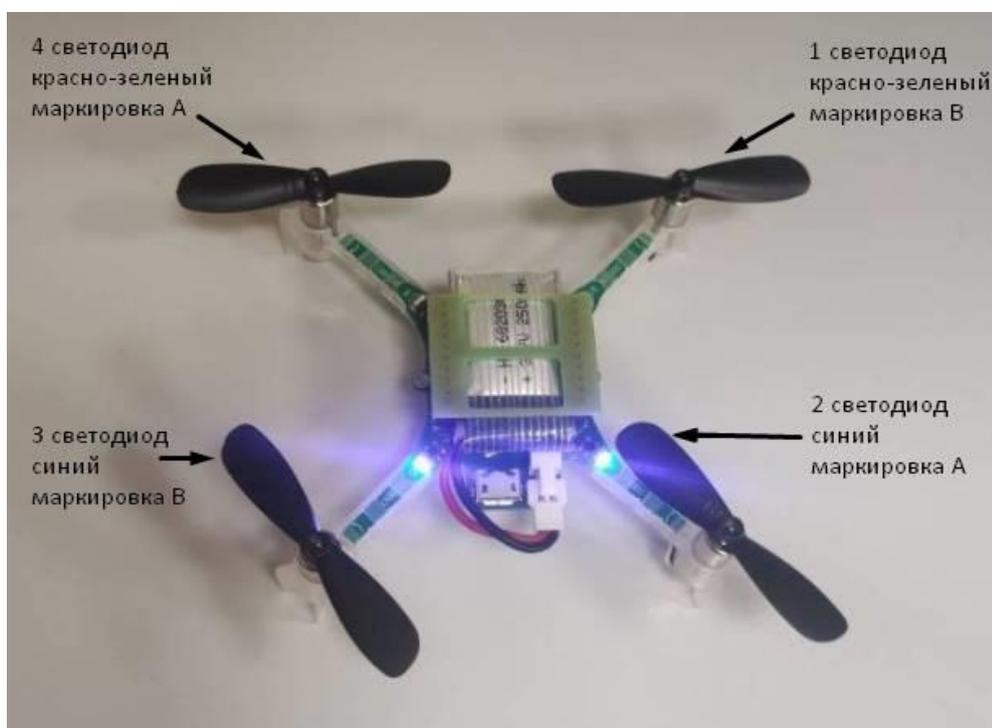


Рисунок 40 – Подключение пропеллеров квадрокоптера

Светодиоды квадрокоптера передают готовность к работе. Ниже описаны шесть состояний работы квадрокоптера.

1. Квадрокоптер подключен к RobboScratch, можно начинать полет, если:

- синие светодиоды 2 и 3 непрерывно горят;
- передний правый светодиод 1 мигает зеленым дважды в секунду;
- 4 светодиода очень часто мигает красно-зеленым.

2. Квадрокоптер нужно подключить к RobboScratch или к Python, если:

- синие светодиоды 2 и 3 непрерывно горят,
- передний правый светодиод 1 мигает зеленым дважды в секунду.

3. Радиоадаптер не подключен к ПК, если передний левый светодиод (4) мигает красным и зеленым.

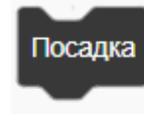
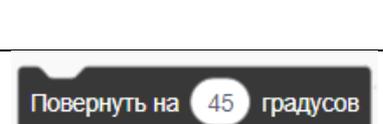
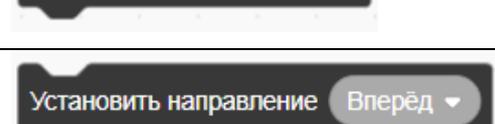
4. Батарея разряжена, если передний правый светодиод (1) непрерывно горит зеленым.

5. Датчики нужно откалибровать, если:

- синие светодиоды (2 и 3) непрерывно горят;
- передний правый светодиод (1) несколько раз быстро промигает красным, потом раз в секунду будет мигать зеленым.

6. Ошибка теста самодиагностики (self test): правый передний светодиод (1) постоянно мигает пятью короткими зелеными импульсами раз в 3-4 секунды. В этом случае перезагрузите квадрокоптер.

9.4. Команды управления квадрокоптером

	Взлет квадрокоптера вверх. Примечание. Любая программа должна включать в себя блоки Взлет и Посадка.
	Посадка квадрокоптера. Примечание. Любая программа должна включать в себя блоки Взлет и Посадка.
	Экстренная посадка квадрокоптера.
	Квадрокоптер летит вперед на заданное количество метров.
	Квадрокоптер летит вперед заданное количество времени. Одна секунда равна одному метру.
	Поворачивает квадрокоптер на заданное количество градусов.
	Устанавливает направление квадрокоптера: вперед, назад, влево, вправо.

Изменить координату по x на 0.1 метров	Перемещение квадрокоптера вправо/влево относительно текущего положения.
Изменить координату по y на 0.1 метров	Перемещение квадрокоптера вперед/назад относительно текущего положения.
Изменить координату по z на 0.1 метров	Перемещение квадрокоптера вверх/вниз относительно текущего положения.
Лететь к x: 0 y: 0 z: 0 (метры)	Задаёт направление полета относительно текущего положения.
Статус	Передаёт факт подключения квадрокоптера к RobboScratch
Координата по x (метры)	Передаёт координату X текущего положения квадрокоптера в метрах.
Координата по y (метры)	Передаёт координату Y текущего положения квадрокоптера в метрах.
Высота (метры)	Передаёт высоту (координату Z) текущего положения квадрокоптера в метрах.
Угол поворота (градусы)	Передаёт угол последнего поворота квадрокоптера в градусах.
Направление	Передаёт направление квадрокоптера. Доступны варианты: вперед, назад, влево, вправо.

9.5. Подключение квадрокоптера к RobboScratch

Шаг 1. Подключите пропеллеры к квадрокоптеру.

Шаг 2. Вставьте Bluetooth адаптер в USB порт компьютера.

Шаг 3. Положите квадрокоптер на ровную поверхность пропеллерами вверх и нажмите кнопку включения.

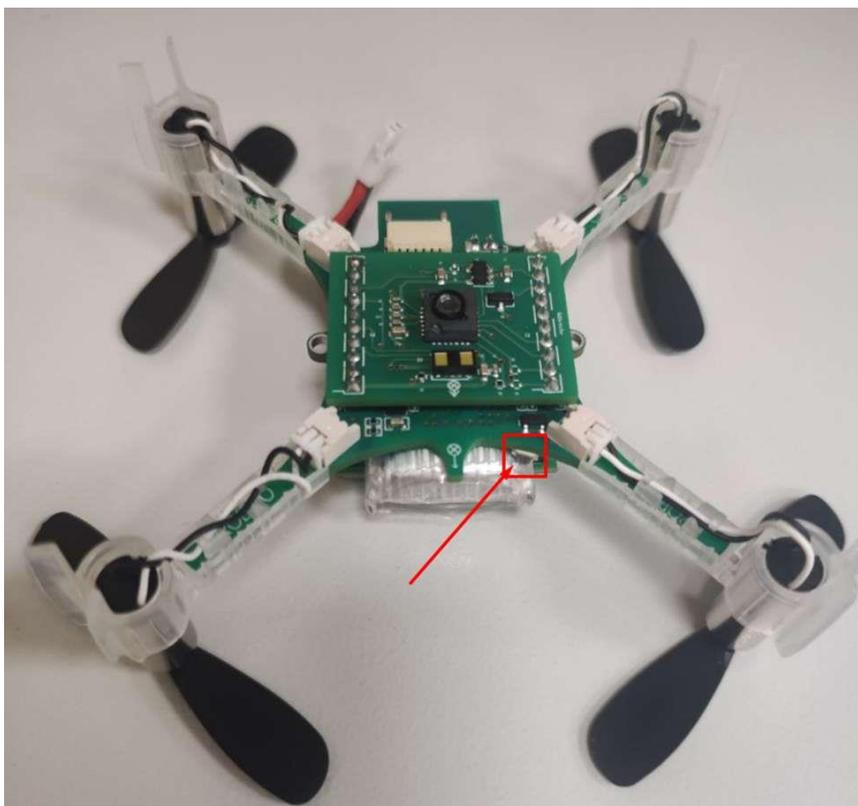


Рисунок 41 – Кнопка включения квадрокоптера

Квадрокоптер начнёт процедуру калибровки: прозвучит звуковой сигнал и прокрутятся все пропеллеры.

Внимание! Подождите завершения калибровки до перехода к следующему шагу.

Результат калибровки: 2 и 3 синий светодиод непрерывно горит, 1 светодиод мигает зеленым дважды в секунду.

Шаг 4. Запустите RobboScratch.

Шаг 5. Нажмите **Поиск устройств**.



Когда квадрокоптер подключится, кнопка Q станет зеленого цвета:  >



Квадрокоптер готов к взлету, при этом:

- синие светодиоды 2 и 3 непрерывно горят;
- передний правый светодиод 1 мигает зеленым дважды в секунду;
- 4 светодиод очень часто мигает красно-зеленым.

Шаг 6. Нажмите кнопку Q, чтобы вызвать меню квадрокоптера (рисунок 42).



Рисунок 42 – Меню квадрокоптера

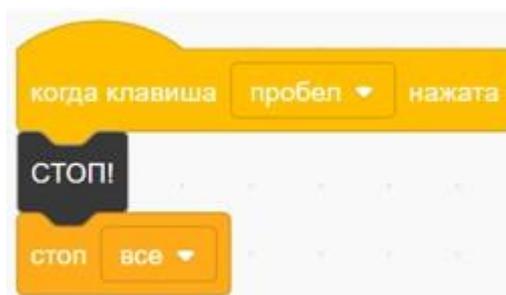
Если показания отображаются в меню, значит квадрокоптер подключился к RobboScratch. Чтобы взлететь, уровень заряда должен быть выше 40%.

9.6. Примеры программ взлета квадрокоптера

Любая программа должна включать в себя блоки **Взлет** и **Посадка**.

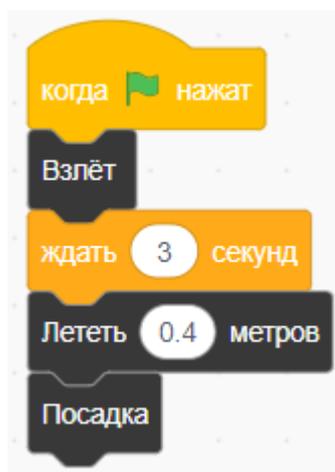
Если светодиод 4 прекратил мигать, значит квадрокоптер отключился от RobboScratch. Чтобы это исправить, нужно перезагрузить питание квадрокоптера и заново подключиться к RobboScratch (раздел 11.3).

Шаг 1. Составьте программу аварийной посадки:



Теперь для мгновенной посадки нужно будет нажать пробел.

Шаг 2. Составьте программу тестового взлета:

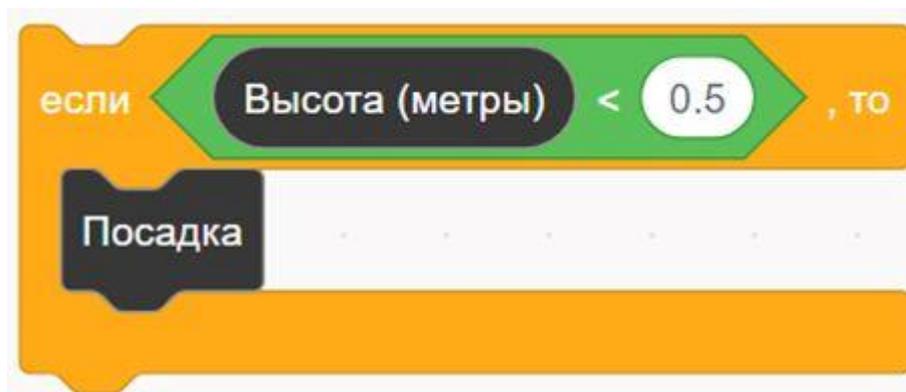


Квадрокоптер взлетит на высоту 30-35 см, продержится на этой высоте 3 секунды, пролетит вперед 40 см и совершит посадку.



Рисунок 43 – Направление полета «вперед» квадрокоптера

Шаг 3. Составьте программу автоматической посадки квадрокоптера.



Квадрокоптер совершит посадку при достижении высоты 50 см.

10. Управление танцующим роботом

10.1. Описание возможностей танцующего робота

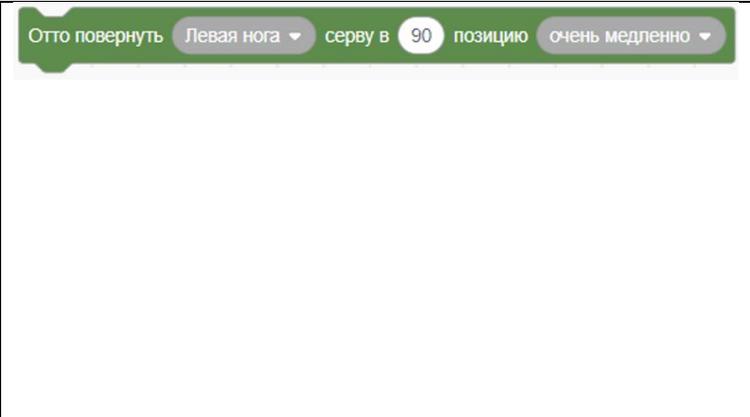
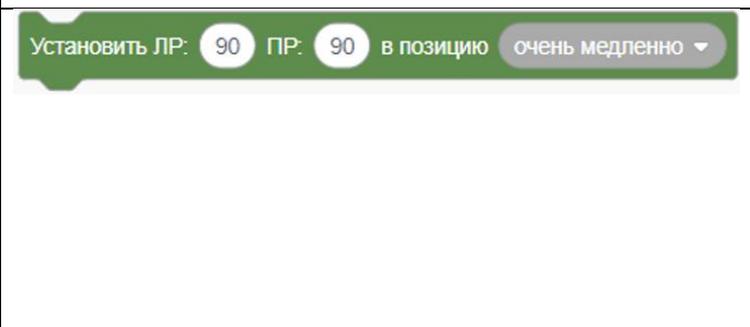
РОББО Танцующий робот — это интерактивный робот-друг, который может ходить, танцевать, издавать звуки или играть мелодии, избегать препятствия

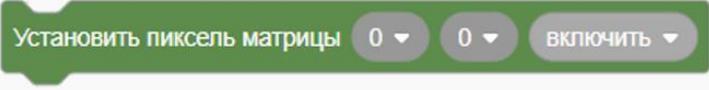
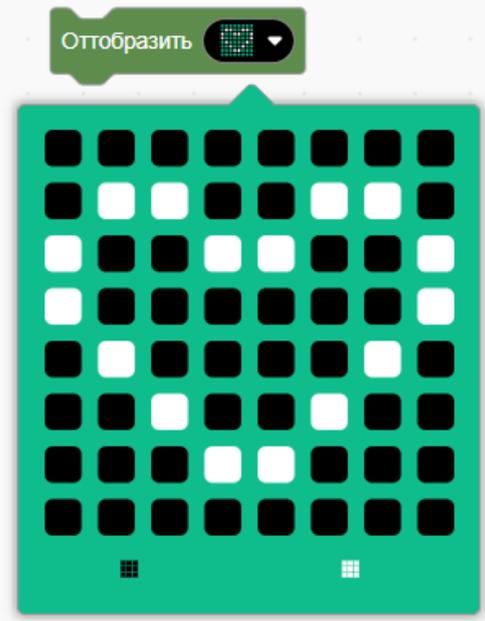
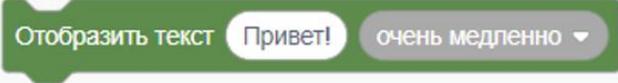
Значения с датчиков и кнопок показаны в меню танцующего робота.



Рисунок 44 – Меню танцующего робота

10.2. Команды управления танцующим роботом

 The screenshot shows a green command block with the text 'Отто повернуть'. Below it, there are several controls: a dropdown menu 'Левая нога', a text input 'серву в' followed by a circular dial set to '90', the text 'позицию', and another dropdown menu 'очень медленно'.	<p>Поворачивает сервопривод выбранной:</p> <ul style="list-style-type: none">• ноги робота• ступни робота• руки робота <p>на назначенное количество градусов с выбранной скоростью.</p>
 The screenshot shows a green command block with the text 'Установить ЛН: 90 ПН: 90 ЛС: 90 ПС: 90 в позицию'. Below it, there are four circular dials, each set to '90', followed by the text 'в позицию' and a dropdown menu 'очень медленно'.	<p>Устанавливает положение:</p> <ul style="list-style-type: none">• левой ноги• правой ноги• левой ступни• правой ступни <p>на назначенное количество градусов с выбранной скоростью.</p>
 The screenshot shows a green command block with the text 'Установить ЛР: 90 ПР: 90 в позицию'. Below it, there are two circular dials, each set to '90', followed by the text 'в позицию' and a dropdown menu 'очень медленно'.	<p>Устанавливает положение:</p> <ul style="list-style-type: none">• левой руки робота• правой руки робота <p>на назначенное количество градусов с выбранной скоростью.</p>

	<p>Воспроизводит звук из динамика назначенной тональности выбранное количество секунд.</p>
	<p>Включает/выключает выбранные пиксели матрицы.</p>
	<p>Отображает на матрице робота нарисованные в блоке фигуры.</p>
	<p>Отображает на матрице робота назначенный текст с выбранной скоростью.</p>
	
	<p>Передает значение уровня шума вокруг танцующего робота.</p>
	<p>Передает расстояние до танцующего робота.</p>

10.3. Подключение и программирование танцующего робота

Шаг 1. Подключите литиевый аккумулятор в разъем питания танцующего робота (Рисунок 45). При подаче питания все сервоприводы повернутся на 90°С.

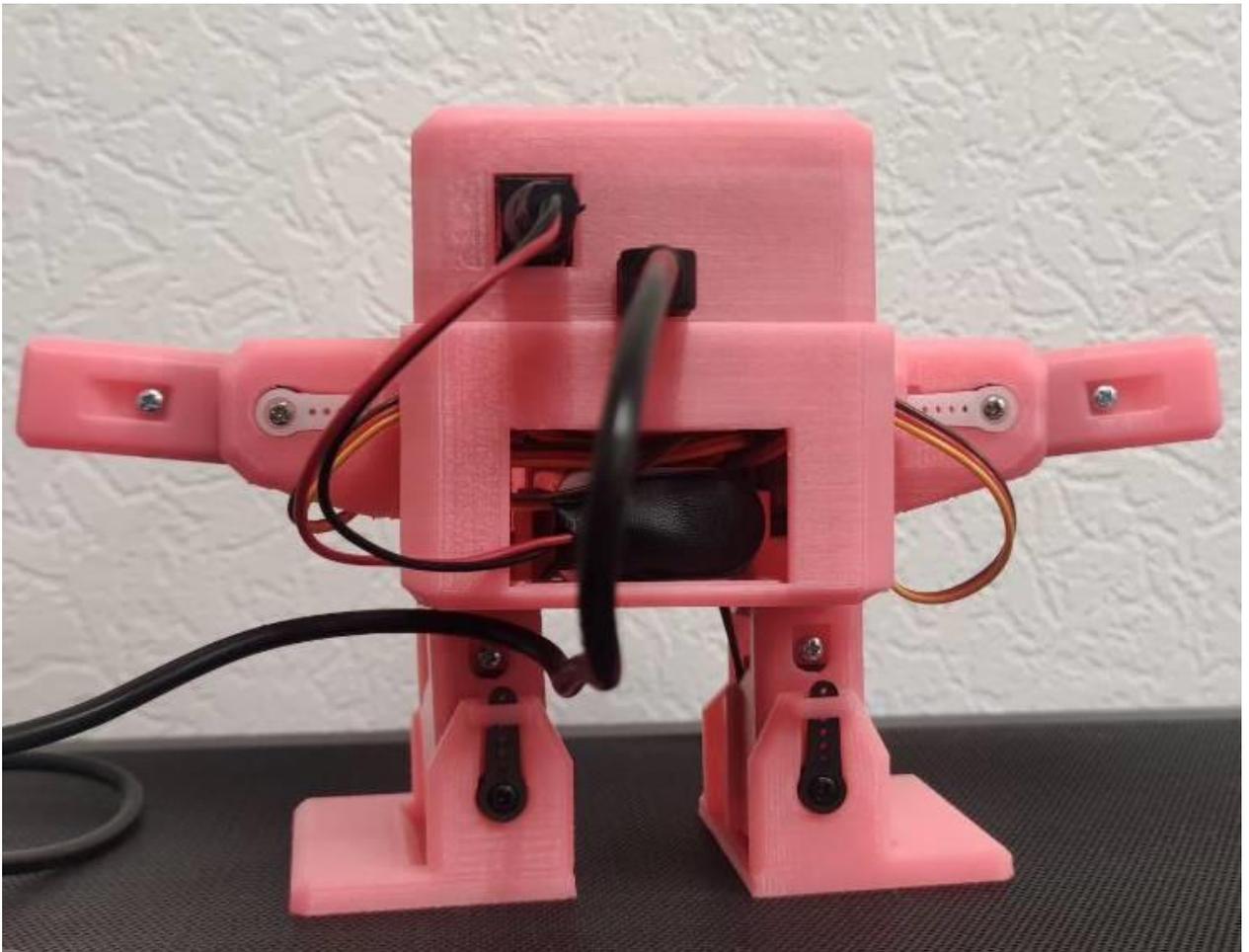


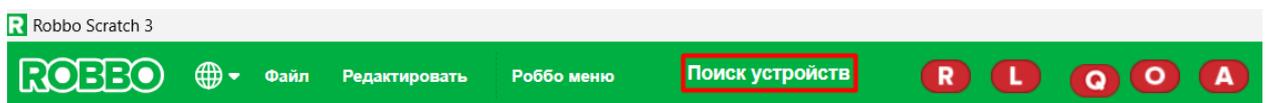
Рисунок 45 – Танцующий робот подключён к ПК по USB

Шаг 2. Подключите танцующего робота к компьютеру через miniUSB.

Шаг 3. Скачайте программу тестирования **otto_selftest.sb3**:

<https://drive.google.com/file/d/1CC1Y0PbTyT2aMoyZvQTbnkFbMcMuak6q/view>

Шаг 4. Запустите RobboScratch и нажмите **Поиск устройств**.



Когда Танцующий робот подключится, прозвучит звуковой сигнал и кнопка **O** станет зеленого цвета: > .

Шаг 5. Нажмите **Файл > Загрузить с компьютера** и откройте файл **otto_selftest.sb3**.

```
повторять всегда
если < Расстояние < 10 >, то
  Установить ЛН: 45 ПН: 90 ЛС: 45 ПС: 90 в позицию нормально
  Установить ЛР: 45 ПР: 90 в позицию нормально
  ждать 1 секунд
  Установить ЛН: 90 ПН: 90 ЛС: 90 ПС: 90 в позицию нормально
  Установить ЛР: 90 ПР: 90 в позицию нормально
  ждать 1 секунд
  Установить ЛН: 90 ПН: 45 ЛС: 90 ПС: 45 в позицию нормально
  Установить ЛР: 90 ПР: 45 в позицию нормально
  ждать 1 секунд
  Установить ЛН: 90 ПН: 90 ЛС: 90 ПС: 90 в позицию нормально
  Установить ЛР: 90 ПР: 90 в позицию нормально
  ждать 1 секунд
  Отобразить текст Hello! очень быстро
  Сыграть ноту 4 1/4 секунды
  ждать 1 секунд
  Нос R: 0 G: 0 B: 255
  ждать 1 секунд
  Нос R: 0 G: 255 B: 0
  ждать 1 секунд
  Нос R: 255 G: 0 B: 0
```

Шаг 6. Чтобы запустить программу проверки, поднесите руку к датчику (к глазам робота).

Проверьте, что действия танцующего робота соответствуют алгоритму.

Танцующий робот совершает действия в таком порядке:

- 1) сгибает, поворачивает левую ногу и ступню на 45*;
- 2) сгибает левую руку на 45*;
- 3) остановка на 1 секунду;
- 4) возвращает левую ногу и ступню в положение 90*;
- 5) возвращает левую руку в положение 90*;
- 6) остановка на 1 секунду;
- 7) сгибает правую ногу и ступню на 45*;
- 8) сгибает правую руку на 45*;
- 9) остановка на 1 секунду;
- 10) возвращает правую ногу и ступню в положение 90*;
- 11) возвращает правую руку в положение 90*;
- 12) остановка на 1 секунду;
- 13) выводит на экран матрицы надпись "Hello!";
- 14) звенит зуммером;
- 15) остановка на 1 секунду;
- 16) светодиод светит синим цветом;
- 17) остановка на 1 секунду;
- 18) светодиод светит зеленым цветом;
- 19) остановка на 1 секунду;
- 20) светодиод светит красным цветом.

Проверка танцующего робота завершена.

11. Управление набором «Схемотехника»

11.1. Описание возможностей

Набор «Схемотехника» состоит из:

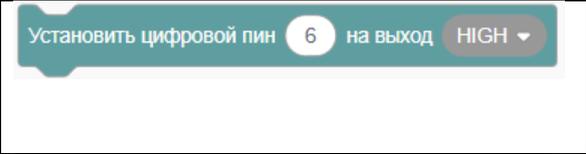
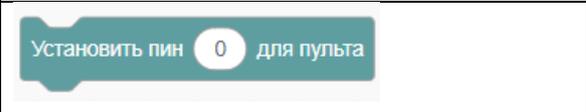
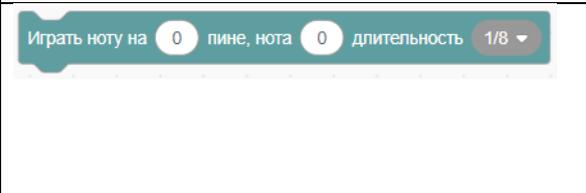
- микроконтроллера Arduino UNO;
- беспаячной макетной платы;
- набора из 100 компонентов для проектирования электронных схем.

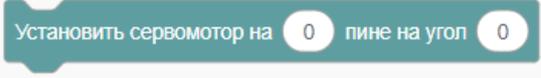
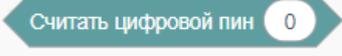
Значения дискретных и аналоговых входов показаны в меню Ардуино (Рисунок 46).



Рисунок 46 - Меню набора «Схемотехника»

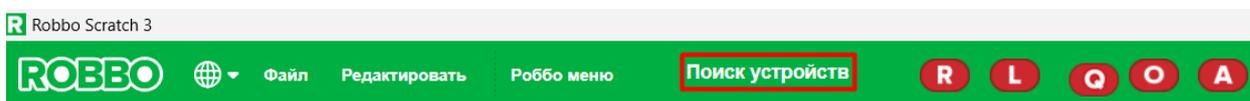
11.2. Команды управления

	Включает (положение HIGH) или выключает (положение LOW) выбранный цифровой пин.
	Определяет выбранный аналоговый пин на выход.
	Определяет выбранный пин для пульта.
	Определяет пины для датчика влажности и датчика температуры.
	Определяет динамик на выбранном пине. Воспроизводит звук из динамика назначенной тональности выбранное количество секунд.

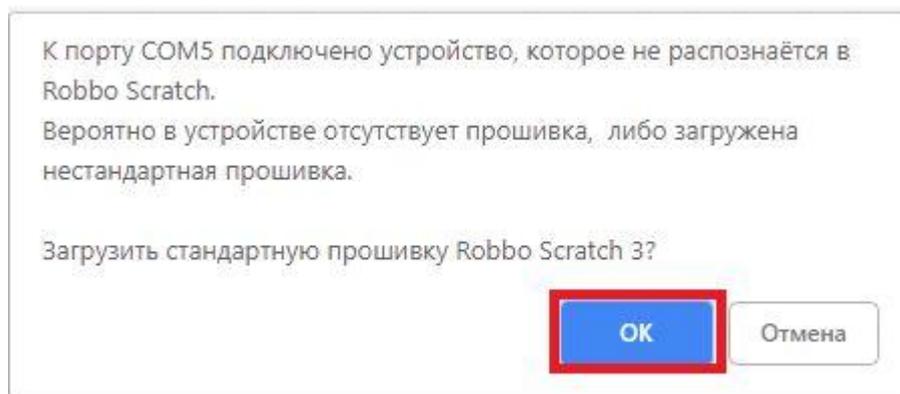
	<p>Определяет пин сервомотора и устанавливает его положения на назначенный угол.</p>
	<p>Определяет назначенный пин как цифровой.</p>
	<p>Определяет выбранный пин как аналоговый.</p>
	<p>Определяет выбранный пин 2 или 3 для ультразвукового дальномера (ультрасоника). Для модели HC-SR04 в поле после echo нужно написать 9.</p>
	<p>Проверяет значение с выбранного цифрового пина. Выдает ответ: 1 (да) или 0 (нет).</p>
	<p>Считывает значения с выбранного аналогового пина.</p>
	<p>Устанавливает курсор в положение 0 или 1.</p>
	<p>Отображает на дисплее назначенный текст.</p>
	<p>Включает или очищает дисплей. Выбор действия зависит от положения в программе.</p>

11.3. Подключение платы Arduino

Шаг 1. Откройте RobboScratch и нажмите **Поиск устройств**.



Шаг 2. В появившемся окне с предложением загрузки прошивки нажмите **ОК**. Не закрывайте программу до завершения прошивки.



После успешной прошивки кнопка А станет зеленого цвета:  > 

Arduino UNO готова к работе в RobboScratch. В случае возникновения ошибки переходите к следующим шагам.

Шаг 3. Закройте RobboScratch.

Шаг 4. Скачайте программу прошивки Arduino.ino:

http://files.robbo.ru/Software/Firmwares/Arduino/ARDUINO_firmware_v3_11_1_1_2019.zip

Шаг 5. Откройте файл Arduino.ino и нажмите , чтобы загрузить прошивку на плату. Дождитесь окончания загрузки.

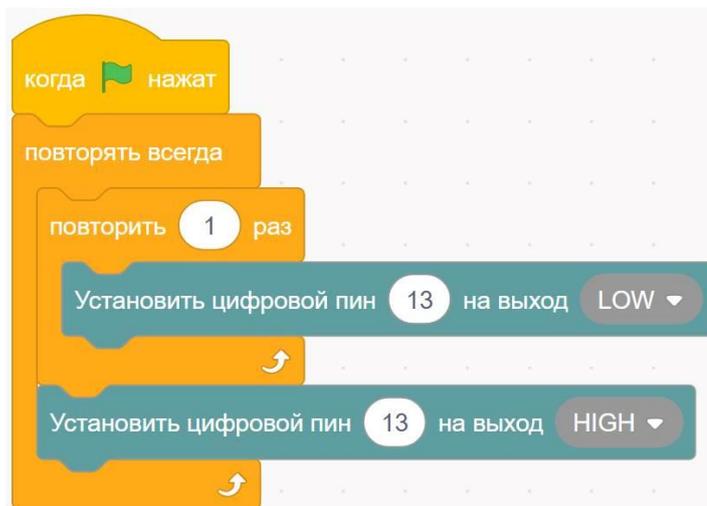
Шаг 6. Запустите RobboScratch и нажмите **Поиск устройств**.

Когда кнопка  станет зеленого цвета , Arduino UNO готова к работе в RobboScratch

11.4. Программирование Arduino

На плате Arduino есть светодиод, подключенный к пину №13. Он отмечен на плате английской буквой "L".

Составьте программу ниже, чтобы запрограммировать мигание светодиода без дополнительной сборки схемы.



Результат работы программы: светодиод мигает с частотой 1 раз в секунду.

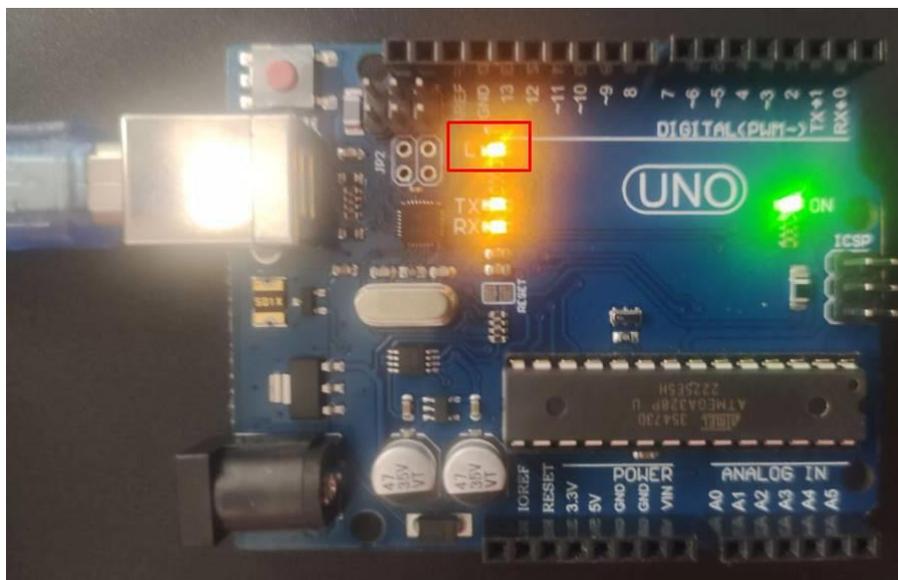


Рисунок 47 – Светодиод на плате Arduino UNO

Дополнительная информация

1. База знаний РОББО:
<https://support.robbo.world/help/ru>
2. Самоучитель по робоплатформе и лаборатории:
<https://robbo.ru/course/forum/>

Техническая поддержка

1. По телефону горячей линии: +7 (812) 565-35-93.
2. По адресу электронной почты: support@robbo.ru.
3. На сайте технической поддержки: <https://support.robbo.world>.
4. Обновления и материалы доступны на сайте:
<http://files.robbo.ru/Software/RobboScratch3.0/>.