



ИНЖЕНЕРНАЯ
И ИТ-ПОДГОТОВКА
ШКОЛЬНИКОВ

Д. Г. Копосов



РОБОТОТЕХНИКА Конструктор SPIKE

LEGO® Education SPIKE™ Prime

5—8 классы

УДК 372.862
ББК 32.97
К65

К65 Копосов Д. Г. Робототехника. Конструктор SPIKE.
5–8 классы : учебное пособие : [издание в pdf-формате] /
Д. Г. Копосов. — М. : Просвещение, 2021. — 176 с. :
ил. — ISBN 978-5-09-087108-2. — Текст электронный.

Учебное пособие знакомит учащихся с практическими задачами образовательной робототехники. Может использоваться с учебниками по информатике, технологии, для подготовки учебных итоговых проектов, а также в рамках внеурочной деятельности, на занятиях кружков и творческих объединений.

Для организации и выполнения практических заданий необходима платформа LEGO® Education SPIKE™ Prime (базовый и ресурсный наборы). Используется среда программирования LEGO® Education SPIKE™, базирующаяся на языке Scratch.

**УДК 372.862
ББК 32.97**

Учебное издание

Копосов Денис Геннадьевич

**Робототехника
Конструктор SPIKE**

5–8 классы

Учебное пособие

Редактор *Л. А. Осипова*

Оформление *Н. А. Новак*

Технический редактор *Е. В. Денюкова*

Корректор *О. Ч. Кохановская*

Компьютерная вёрстка: *Н. И. Беляева*

Формат 84×108/16. Усл. печ. л. 18,48.

Акционерное общество «Издательство «Просвещение»
Российская Федерация, 127473, г. Москва, ул. Краснопролетарская, д. 16,
стр. 3, этаж 4, помещение I

Адрес электронной почты «Горячей линии» — **vopros@prosv.ru**

ISBN 978-5-09-087108-2

© АО «Издательство «Просвещение», 2021
© Художественное оформление
АО «Издательство «Просвещение», 2021
Все права защищены

Содержание

Вместо предисловия	4
§ 1. Сначала получите первый опыт.....	5
§ 2. Зачем изучать технические характеристики	9
§ 3. Движение вдоль линии	13
§ 4. Движение вдоль линии. ПД-регулятор.....	22
§ 5. ПИД-регулятор	25
§ 6. Расположение датчиков.....	30
§ 7. Как проехать прямой угол	32
§ 8. Движение по энкодерам	35
§ 9. Здравствуй, одометр.....	47
§ 10. Робот всегда ошибается.....	51
§ 11. Остановка по одометру	53
§ 12. Движение вдоль линии. Один датчик.....	56
§ 13. Находим максимум и минимум.....	63
§ 14. Ищем объекты и определяем цвета.....	67
§ 15. Шаг вперёд, два шага назад.....	77
§ 16. Технология. Робототехника. Олимпиада.....	88
§ 17. Обнаружение объекта. Используем дальномер.....	96
§ 18. Технология. Робототехника. Олимпиадное задание	103
§ 19. Поиск объектов на линии.....	109
§ 20. Траектория. Повторяем и систематизируем.....	122
§ 21. Пробуем двигаться вдоль линии быстро.....	132
§ 22. Захват объекта и определение цвета.....	139
§ 23. Проекты, проекты, проекты... ..	149
§ 24. Функциональная схема робота.....	156
Сделай свой выбор... ..	162
Приложения	164
Приложение 1. Информация для организаторов	164
Приложение 2. Ответы на задание 3.....	165
Приложение 3. Модель для движения по линии.....	166
Приложение 4. Движение по линии (трасса «Биатлон»)	167
Приложение 5. Движение по линии с обнаружением объектов	169
Приложение 6. Программа для трассы «Кегельринг-квадро»	172

§ 1.

Сначала получите первый опыт

В жизни человек с выдержкой всегда побеждает.

А выдерживать надо не полчаса, а годами.

Перед вами самый новый, на момент написания этой книги конечно, робототехнический комплект SPIKE™ Prime, созданный известной мировой компанией.

Возможно, у вас и ваших друзей уже есть опыт работы с этим набором. Однако если вы в первый раз работаете со Спайком (мы будем называть робота именно так), то для начала выполните те задания и упражнения, которые есть в комплекте с программным обеспечением, по блокам:

- Отряд изобретателей;
- Запускаем бизнес;
- Полезные приспособления;
- Готовимся к соревнованиям.



Задание 1

Пройдите все миссии и выполните все задания в четырёх блоках учебных материалов (рис. 1, 2).



Рис. 1. Первые два блока учебных материалов

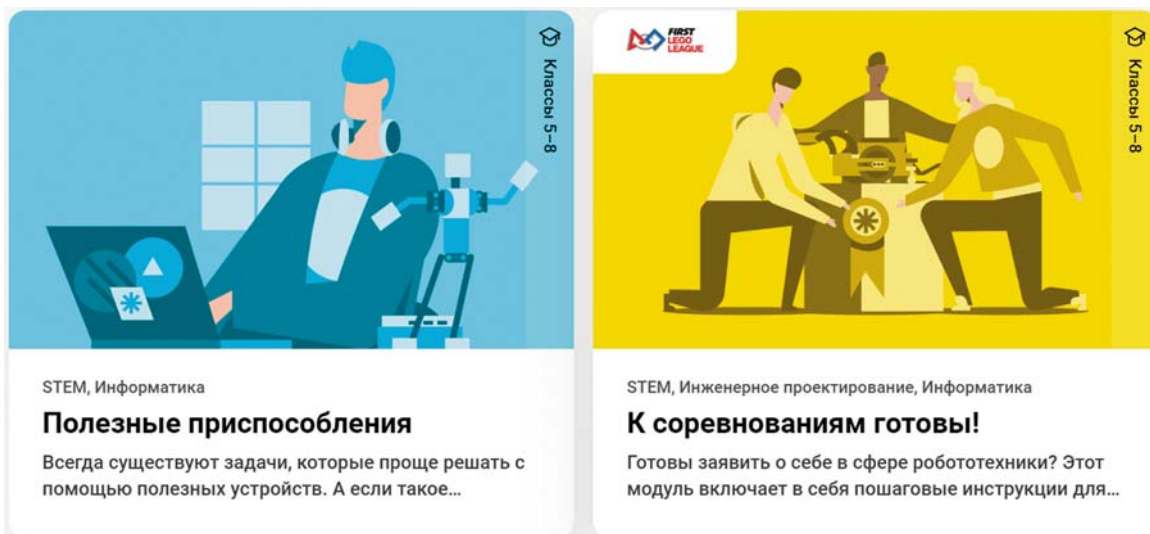


Рис. 2. Вторые два блока учебных материалов

Если вы выполнили все задания, собрали все модели, исследовали все представленные проблемы и что-то внутри говорит: «Было очень здорово и жалко, что всё закончилось», — вы готовы идти дальше. Это важно.

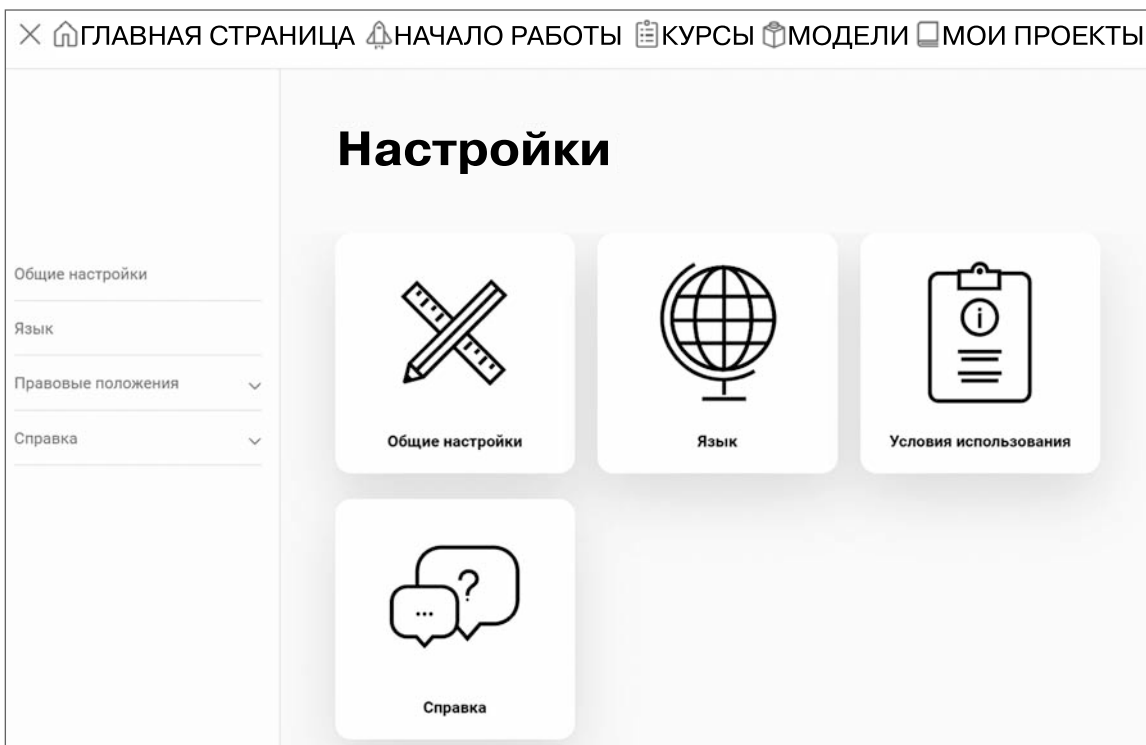


Рис. 3. Вкладка «Настройки»

Чтобы продолжить, мы должны разговаривать, как говорят, «на одном языке», важно не просто знать, что существует справочная система (рис. 3), но и регулярно ею пользоваться.



Задание 2

Изучите в справочной системе (рис. 4) информацию об аппаратном обеспечении Спайка: хабе, моторах, датчиках. Уделите особое внимание техническим характеристикам.



×
🏠 ГЛАВНАЯ СТРАНИЦА
 🔔 НАЧАЛО РАБОТЫ
 📅 КУРСЫ
 📦 МОДЕЛИ
 📁 МОИ ПРОЕКТЫ

Справка

Основные характеристики

* Светодиодный матричный белый дисплей 5 x 5. * Шесть портов входа/выхода (порт A, B, C, D, E и F). * Автоматическое обнаружение датчиков и моторов. * Возможность свободного выбора портов при одновременном использовании нескольких моторов. * 6-осевой гироскопический датчик (3-осевой акселерометр и 3-осевой гироскоп). * Навигация с использованием трёх кнопок, включая интерфейс со световой индикацией. * Динамик. * Возможность подключения к устройствам через кабель микро-USB и Bluetooth. * Система подключения оборудования LEGO Power Functions 2.0 (LPF2) для подключения датчиков и моторов LEGO. * Операционная система MicroPython. * Центральный процессор ARM M4 с частотой 100 МГц, 320 Кб ОЗУ и 1 Мб ФЛЕШ-памяти. * Память ёмкостью 32 Мбайт для программ, звука и контента. * Открытая платформа.

Подробную информацию см. в разделе *Поддержка* на сайте LEGOeducation.ru.

Обзор аппаратного обеспечения

Большой Хаб Technic

Описание

Основные характеристики

Аккумуляторная батарея дл...

Средний угловой мотор Tech...

Большой угловой мотор Тес...

Датчик цвета Technic

Рис. 4. Справка о хабе Спайка



Задание 3

Найдите в справочной системе ответы на следующие вопросы.

1. Какова частота работы датчиков? Что она означает?
2. Через сколько подключений зарядного устройства к Спайку уровень заряда может упасть до 30 %?
3. Можно ли для обнаружения объектов использовать датчик цвета?
4. Перечислите цвета, которые способен распознавать датчик цвета.

5. Оцените, какое максимальное расстояние может проехать Спайк за одну минуту при использовании стандартных колёс из базового набора?

Давайте экспериментально оценим, какое расстояние может проехать робот на стандартных колёсах.

Во-первых, нужно определиться с моторами. В техническом описании приведены следующие данные:

- средний мотор: 185 об/мин $\pm 15\%$;
- большой мотор: 175 об/мин $\pm 15\%$.

Таким образом, нам нужно понять, какое расстояние проедет колесо за один оборот. Это мы можем определить, например, экспериментально, просто покатав колесо (рис. 5).

В итоге у нас получится, что за один оборот Спайк проедет 176 мм. Таким образом, при использовании:

- среднего мотора робот проедет 32,56 м $\pm 15\%$ (27,7–37,4 м);
- большого мотора: 30,8 м $\pm 15\%$ (26,2–35,4 м).

Ответы на первые четыре вопроса вы можете посмотреть в приложениях.

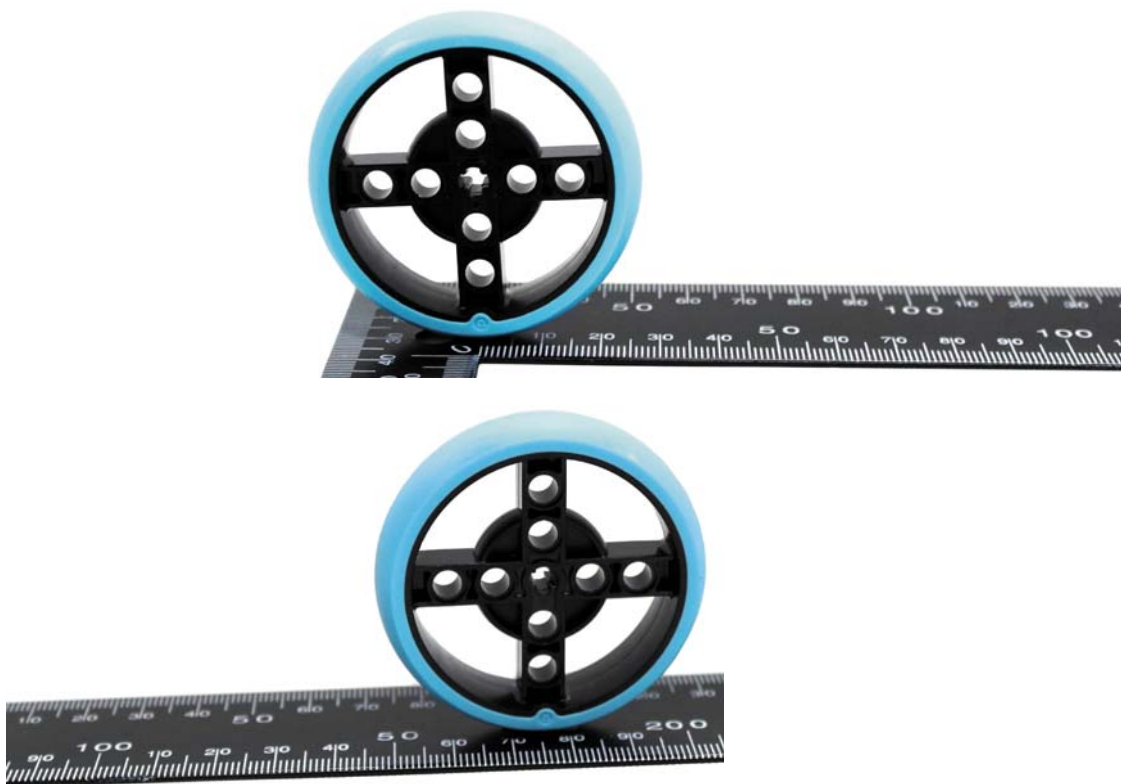


Рис. 5. Экспериментальное определение длины окружности колеса

§ 2.

Зачем изучать технические характеристики

Хороший инженер должен состоять из четырёх частей: на 25 % быть теоретиком; на 25 % — художником; на 25 % — экспериментатором, исследовать свою машину; и на 25 % он должен быть изобретателем.

Как вы уже осознали, знание технических параметров оборудования позволяет делать предварительные оценки и прогнозы о том, как поведёт себя робот при тех или иных условиях. Можно, конечно, экспериментально всё проверить. Однако в разы сократить количество неэффективной работы поможет изучение технических характеристик блока управления, моторов и сенсоров. Рассмотрим датчик цвета.



Задание 4

Ещё раз прочитайте основные характеристики датчика цвета (рис. 6).

×
🏠 ГЛАВНАЯ СТРАНИЦА
 🔔 НАЧАЛО РАБОТЫ
 📅 КУРСЫ
 🧩 МОДЕЛИ
 📁 МОИ ПРОЕКТЫ

Работа с Приложением ▼
 Обзор аппаратного обеспечения ▲
 Большой Хаб Technic ▼
 Аккумуляторная батарея дл... ▼
 Средний угловой мотор Tech... ▼
 Большой угловой мотор Тес... ▼
 Датчик цвета Technic ▲
 Описание
 Основные характеристики

Справка

Датчик цвета Technic

Описание

Этот датчик способен обнаруживать цвет, улавливать отражённый свет, а также измерять уровень освещённости. Его также можно использовать в качестве источника света.

Основные характеристики

* Обнаружение цвета (форматы RGB, HSV и основные цвета LEGO®). * Улавливание отражённого света (для отслеживания линий). * Измерение уровня освещённости (темнота = 0 %; яркость = 100 %). * Излучение белого света. * Частота опроса датчика: 100 Гц. * Система подключения LEGO Power Functions 2.0 (LPF2) для подключения к Хабам LEGO. * Длина провода: 250 мм. * Оптимальное расстояние для измерения: 16 мм (в зависимости от размера, цвета и характеристик поверхностей объекта). *

Рис. 6. Справочная система

Как вы уже знаете, частота работы датчика 100 герц (Гц). То есть блок управления будет получать 100 числовых значений в секунду. Это может показаться достаточно большим значением. Но давайте узнаем, какое расстояние проедет робот между двумя измерениями.



Задание 5



Соберите модель робота, похожего на простую трёхколёсную тележку. Напишите программу, чтобы узнать, какое расстояние робот проедет за 1 с (рис. 7, 8).

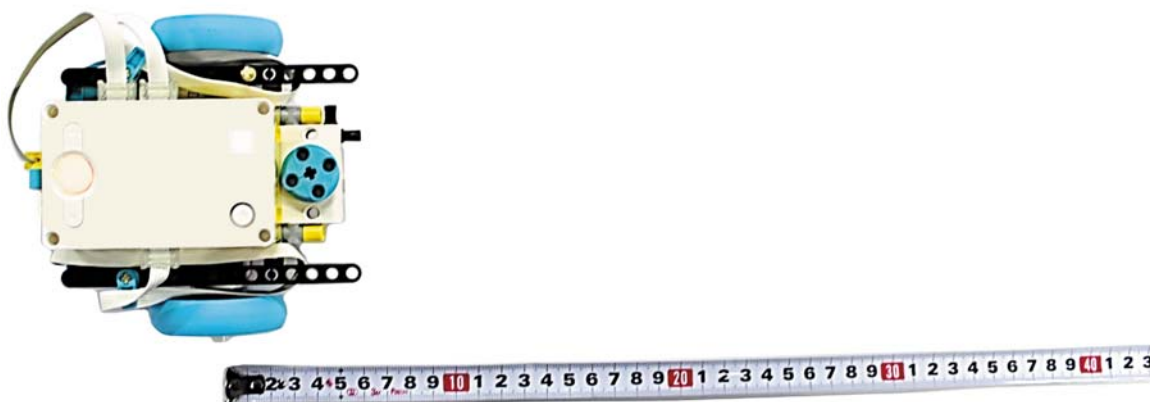


Рис. 7. Стартовое расположение робота

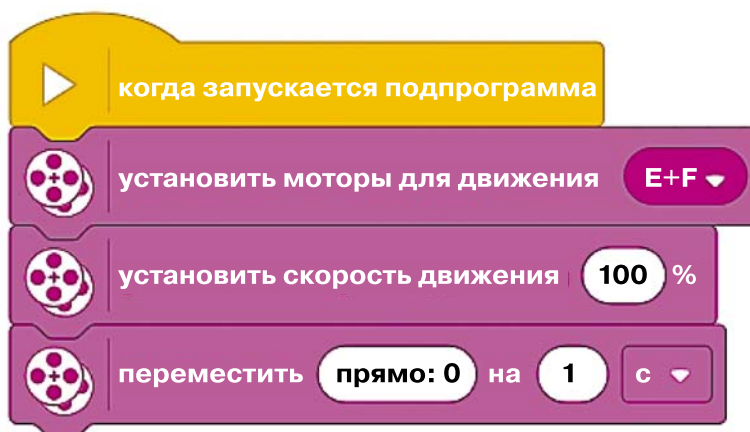


Рис. 8. Программа для задания

Наш робот за 1 с проехал примерно 40 см. За это время произведено 100 измерений. Следовательно, между измерениями он проедет 4 мм. Тогда показания датчика цвета при движении по линии будут изменяться не очень плавно. Давайте узнаем, как именно. Сделаем несколько измерений яркости отражённого от поверхности света, при этом будем двигать робота относительно чёрной линии на 3 мм.

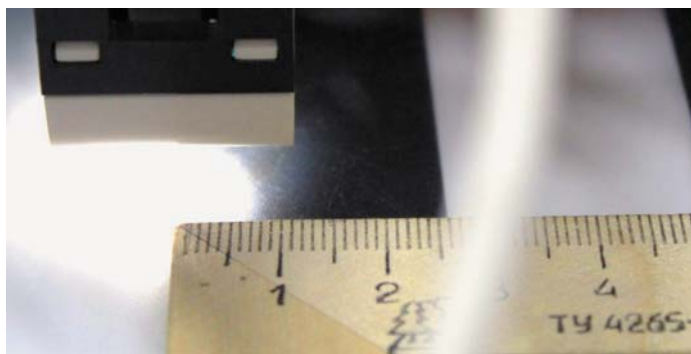


Рис. 9. Расположение датчика цвета над чёрной линией

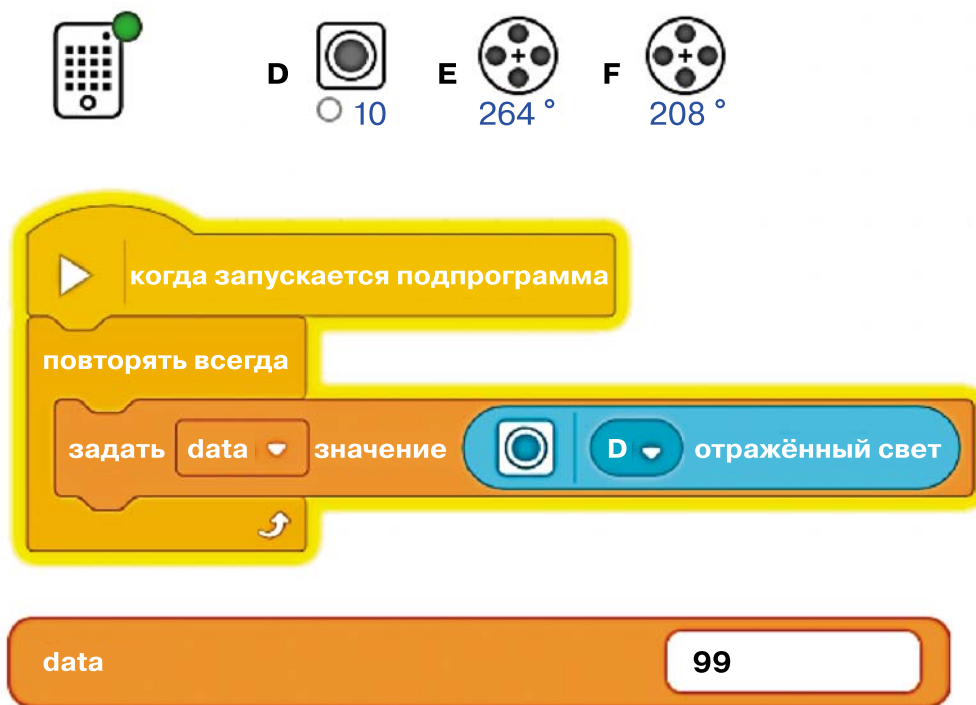


Рис. 10. Программа для задания



Задание 6



Соберите модель робота, похожего на простую трёхколёсную тележку. Фотографии модели представлены в приложении 3. Напишите программу, выводящую на информационный экран показания датчика цвета в режиме «*яркость отражённого света*» (рис. 9, 10). Сделайте серию измерений (значения можно записать в таблицу).

Иногда для получения представления о том, какие показания датчиков могут быть при тех или иных условиях, удобнее настроить отображения показаний датчиков в среде программирования так, как на рисунке 11.

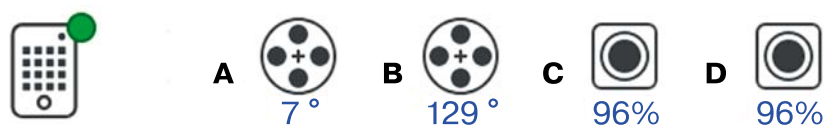


Рис. 11. Показания сенсоров, подключённых к хабу



Рис. 12. Настройки хаба



Задание 7

Настройте отображение показаний отражённого света, щёлкнув по изображению хаба с зелёным кружочком. Далее в информационной панели выберите нужный режим (рис. 12). Исследуйте, как изменятся показания датчика цвета при перемещении его над чёрной линией и белым полем.

Итак, благодаря знанию технических характеристик, небольшим экспериментам и пониманию того, что робот не способен останавливаться или изменять направление мгновенно (на это тоже уйдёт какое-то время), сделаем вывод, что вдоль тонкой (2 см) линии Спайк не сможет двигаться плавно на больших скоростях.

§ 3.

Движение вдоль линии

В основе научных закономерностей происходящих в природе процессов лежит логическое обобщение опыта, полученного из «проб и ошибок».

Мы плавно подошли к классической задаче робототехники — созданию и программированию робота, следующего вдоль тёмной линии на светлом поле.

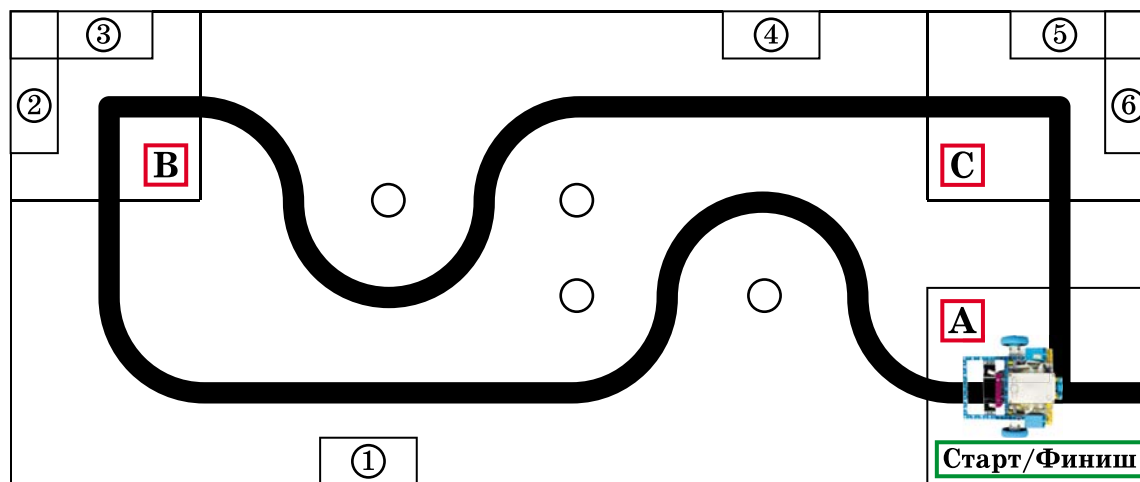


Рис. 13. Поле «Биатлон»

Для этого нам потребуется, во-первых, поле «Биатлон» (рис. 13), а во-вторых, собранный робот (рис. 14).

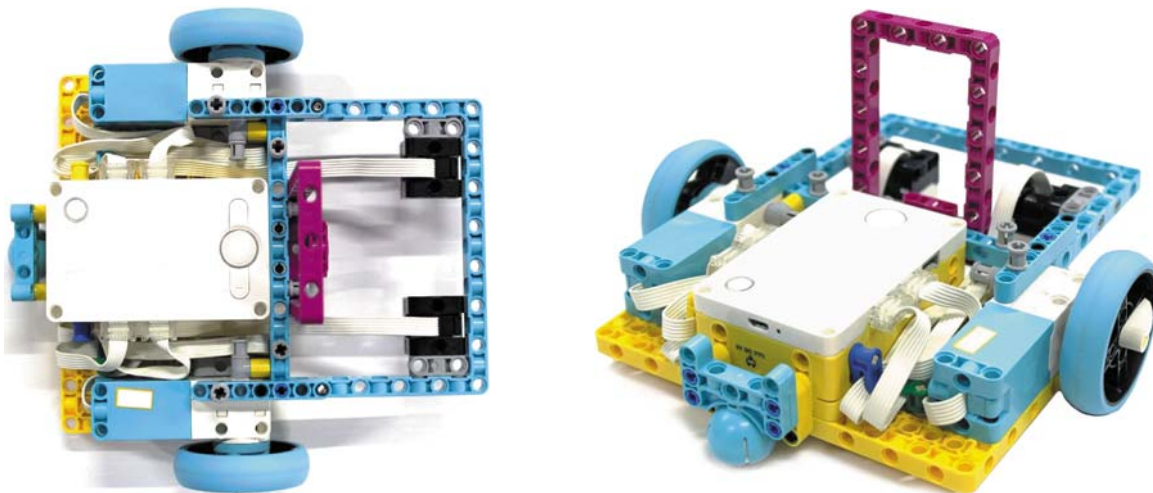


Рис. 14. Модель учебного робота для движения вдоль линии



Задание 8



Подготовьте поле «Биатлон» и соберите учебную модель робота. Дополнительные фотографии приведены в приложениях.



Чтобы робот Спайк двигался плавно вдоль линии, в программном обеспечении необходимо подключить расширения, в частности «Дополнительные моторы».



Задание 9

Подключите нужные расширения, как указано далее.



Для подключения расширений в созданный проект необходимо:

- 1) в левом нижнем углу нажать на иконку «Показать расширения» (рис. 15);
- 2) в появившемся окне «Расширения» кликнуть по расширению «Дополнительные моторы» (рис. 16);

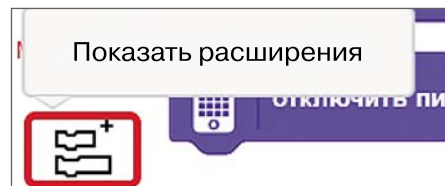


Рис. 15. Кнопка «Показать расширения»

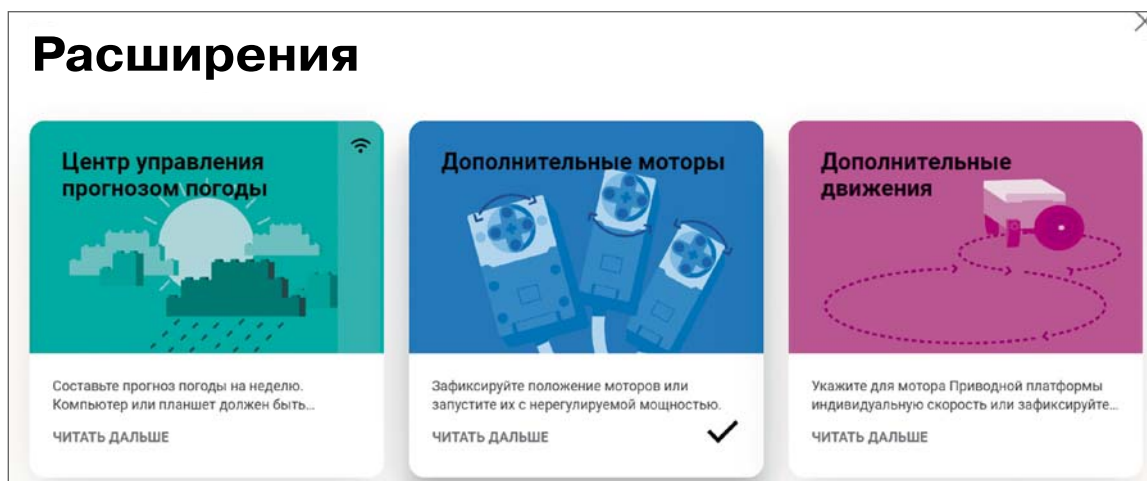


Рис. 16. Список расширений

3) когда расширение будет отмечено, можно закрыть это окно. В вашем распоряжении появились новые команды (рис. 17).

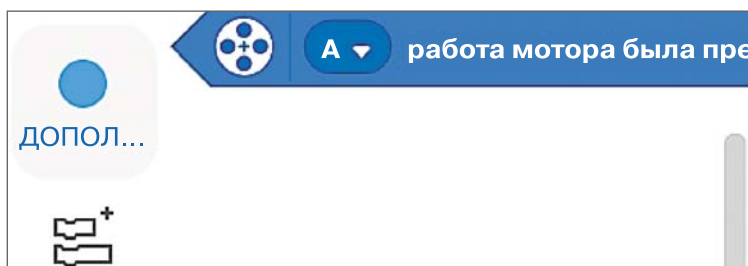


Рис. 17. Подключённые расширенные команды



Задание 10

Изучите информацию о дополнительных блоках управления моторами, представленную в справочной системе.

Пропорциональный регулятор (два датчика цвета)

Как вы поняли по модели робота, мы будем рассматривать движение по линии с помощью двух датчиков цвета, работающих в режиме отражённого света.



Задание 11

Проанализируйте примеры расположения двух датчиков над чёрной линией и их примерные показания, приведённые на рисунке 18.



Общая идея алгоритма может быть описана следующим образом:

- если оба датчика находятся над белым полем, то логично, что робот должен двигаться прямо;
- чем больше левый датчик находится над чёрной линией, тем сильнее робот поворачивает налево;
- чем больше правый датчик находится над чёрной линией, тем сильнее робот поворачивает направо.

Получается, что чем меньше показания датчика цвета, тем меньше скорость/мощность соответствующего мотора.

Вопрос в том, как описать зависимость формулой.

Вот какую идею мы применим. Посмотрите на рисунок 19. Когда показания датчиков равны, робот едет прямо. При этом у

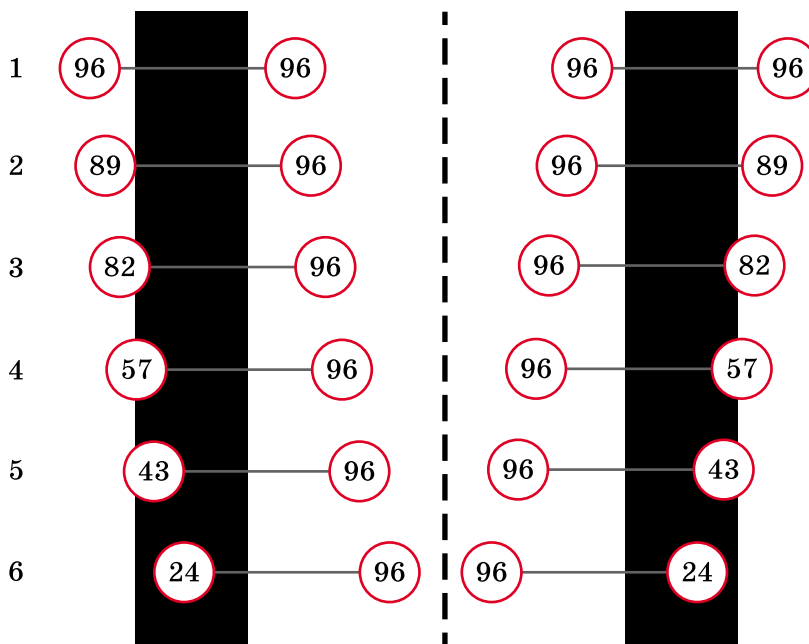


Рис. 18. Примеры расположения двух датчиков над линией

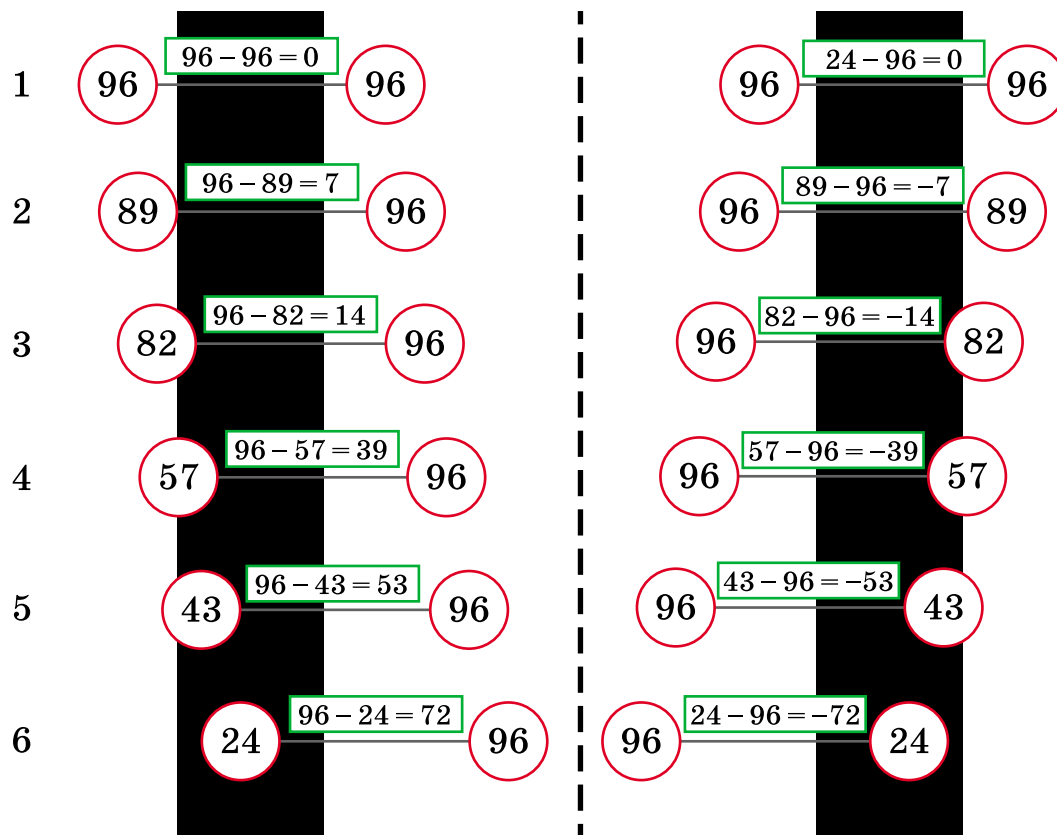


Рис. 19. Примеры расхождений в показаниях датчиков

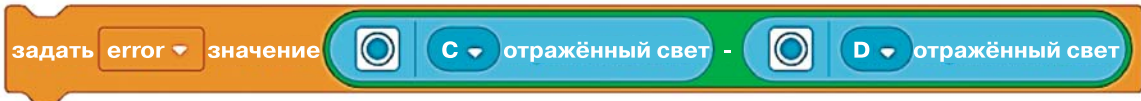
него должна быть какая-то скорость. Например, 50 % (это в процентах от максимальной).



Давайте перефразируем: когда разность показаний датчиков равна нулю, то робот должен двигаться прямо. Мы будем отслеживать разность показаний правого и левого датчиков. Другими словами, будем вычислять отклонение от нуля. 0, 7, 14, 39, 53, 72 — это всё отклонения, и -7, -14, -39, -53, -72 — это также отклонения (только в другую сторону). А сам нуль мы назовём целевым значением, так как робот (по нашей программе, конечно) должен стараться и стремиться, чтобы разность показаний была равна нулю.

Вместо термина «отклонение» часто используется слово «ошибка» (*error*). На языке Scratch это будет выглядеть, как на рисунке.





Проблема в том, что у нас ошибки могут быть очень большими. Например, 72 или -72! А ехать мы планировали при мощности 50 %. Что делать?

Логично учитывать не само отклонение, а его часть. Например, $1/3$ (одну треть) отклонения, или $1/5$ (одну пятую), или $3/10$ (три десятых). Мы подберём значение экспериментально.

Чтобы робот быстрее реагировал на изменения ситуации при движении по линии, будем к мощности правого мотора прибавлять часть отклонения, а из мощности левого — вычитать это значение.

Для этого используется термин «управляющее воздействие» (*control action*).



Итак, мы будем управлять мощностью моторов на основе отклонения от целевого значения. Говорят: «управлять по отклонению».

Давайте напишем итоговую программу управления роботом.



Задание 12

Создайте программу движения робота по линии (рис. 20). Протестируйте робота в работе. Пример, как робот движется по этой программе, по ссылке: <https://youtu.be/lzJpNVMsaGQ>

Несколько пояснений к программе:

- имена переменных должны отражать пояснение, зачем они используются: *error* — ошибка, *speedStart* — стартовая скорость, *controlAction* — управляющее воздействие;
- вначале обязательно следует задать первоначальные значения переменных, которые будут использоваться в вычислениях (*error* и *speedStart*);
- чтобы робот сразу не начинал движение, мы добавим ожидание нажатия на правую кнопку;