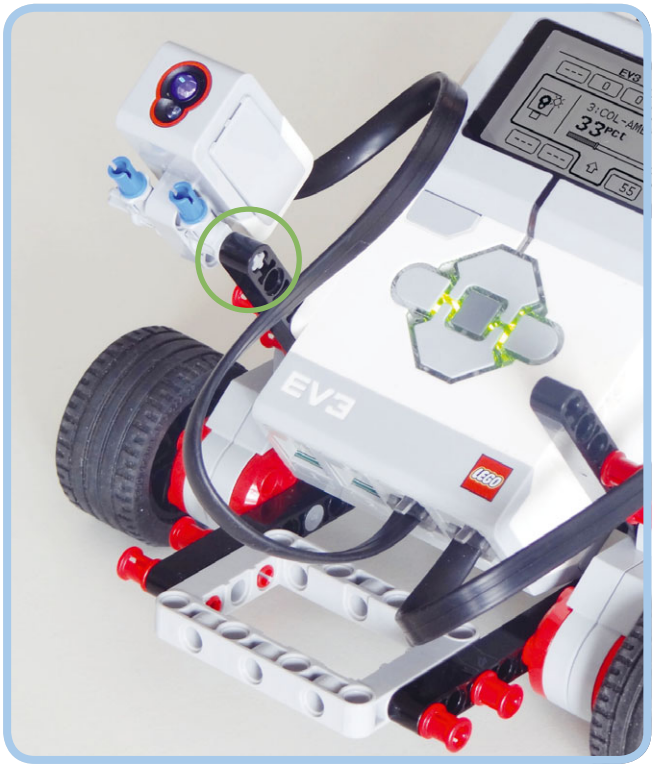

Рис. 7.16. Снимите датчик цвета с передней части робота и прикрепите его сбоку. Для этого вам не понадобятся дополнительные детали LEGO. Кабель остается подключенным к входному порту 3

учитывал, включен свет в комнате или выключен. Для полноценной работы рекомендуется закрепить датчик цвета так, как показано на рис. 7.16.

Измерение яркости внешнего освещения

Запустите программу **Port View** на модуле EV3 и выберите пункт **COL-AMBIENT**, чтобы просмотреть показания датчика о яркости окружающего освещения. Значение колеблется от 0% (очень темно) до 100% (очень ярко). Если вы, к примеру, прикроете датчик рукой, он передаст значение ниже 5%, а если поставите его рядом с лампой, то значение может составить около 70%.

В этой программе яркость окружающего света можно измерить теми же способами, которые применялись для замеров яркости отраженного света, но теперь вы будете использовать режим **Датчик цвета** ▸ **Сравнение** ▸ **Яркость внешнего освещения** (Color Sensor ▸ Compare ▸ Ambient Light Intensity). Чтобы определить, к примеру, включен ли свет в комнате, робот может использовать блок **Переключатель** (Switch), следящий, превышает ли показание датчика пороговое значение. В этой ситуации пороговым должно быть среднее значение между показаниями датчика, полученными при включенном свете и при выключенном.

ПРИМЕЧАНИЕ **В режиме Яркость внешнего освещения (Ambient Light Intensity) датчик излучает тусклый синий свет, но это никак не влияет на точность измерения. При этом отраженный свет не измеряется; датчик фиксирует только свет, поступающий из окружающей среды.**

Программа MorseCode

Теперь мы создадим программу, которая позволяет управлять роботом в темной комнате с помощью световых сигналов. Вы запрограммируете робота ехать вправо, если он фиксирует свет более двух секунд, и влево, если фиксируется короткий световой сигнал. Это позволяет управлять направлением движения робота путем включения и выключения света в комнате. Это упрощенная форма *азбуки Морзе* — способа связи, распространенного до изобретения телефона.

Вначале используется блок **Ожидание** (Wait), чтобы подождать поступления светового сигнала. После поступления сигнала выполняется ожидание с помощью другого блока **Ожидание** (Wait) в течение еще двух секунд, а затем программа использует блок **Переключатель** (Switch), чтобы определить, горит ли все еще свет. Если да, то робот поворачивает направо; в противном случае он поворачивает налево.

Создайте программу MorseCode, показанную на рис. 7.17, и запустите ее на роботе в темной комнате. Кроме того, вы

ПРАКТИКУМ № 36: УТРЕННИЙ БУДИЛЬНИК!

Сложность: **Время:**

Можете ли вы сделать так, чтобы ваш робот издавал сигнал тревоги, когда встает солнце? Поместите робота рядом с окном. В программе должен присутствовать блок **Ожидание** (Wait), который приостановит выполнение программы до тех пор, пока яркость внешнего освещения не поднимется выше порогового значения, которое вы вычислили. При достижении порога робот должен циклично воспроизводить громкий звук, пока вы не нажмете кнопку датчика касания, который действует как кнопка отключения.

СОВЕТ **Робот обычно выключается, если вы не используете его в течение 30 минут, поэтому он не разбудит вас утром. Чтобы ваш робот-будильник сработал, перейдите на вкладку Settings (Настройки) на экране модуля EV3; выберите пункт Sleep (Сон), а затем пункт Never (Никогда). На следующий день не забудьте вернуть параметру Sleep (Сон) значение 30 минут, чтобы батарея не разрядилась, если вы забудете выключить робота.**

*Рис. 7.17. Программа MorseCode. И первый блок **Ожидание**, и блок **Переключатель** находятся в режиме **Датчик цвета ▸ Сравнение ▸ Яркость внешнего освещения***

можете попробовать протестировать ее в освещенной комнате, посылая световые сигналы ярким фонарем.

ПРИМЕЧАНИЕ В моей комнате значение измерений датчика при выключенном свете составляет 2, и 16%, когда свет включен. Поэтому я выбрал пороговое значение 9% для программы MorseCode. Для ваших условий вы должны определить собственное пороговое значение.

Дальнейшее изучение

Датчик цвета позволяет роботу «ощущать» окружающую среду путем обнаружения цвета, яркости отраженного света и яркости внешнего освещения. Робот использует информацию с этого датчика для выполнения задач различного вида. В этой главе вы запрограммировали EXPLOR3R на движение по трассе и реагирование на световые сигналы.

Вы также научились вычислять пороговые значения и сравнивать их с замерами датчика, чтобы ваш робот мог обнаруживать изменения в окружающей среде. Робот использует пороговые значения, чтобы определить, находится ли он на черной линии и включен ли свет в комнате. Пороговые значения применяются и для других датчиков, поэтому позднее мы снова будем их использовать.

Датчик цвета является универсальным устройством, и существует множество интересных способов его использования. Попытайтесь выполнить некоторые из практических задач, и вы увидите, что можно придумать!

ПРАКТИКУМ № 37: ЦВЕТОВЫЕ САЛКИ!

Сложность: **Время:**

Установите датчик цвета, как показано на рис. 7.16, и запрограммируйте робота на движение в разных направлениях в зависимости от того, какого цвета предмет вы подносите к датчику. Каждое движение робота должно длиться три секунды.

ПРАКТИКУМ № 38: СКАНЕР ОТПЕЧАТКОВ ПАЛЬЦЕВ!

Сложность: **Время:**

Можете ли вы разработать такую программу для робота, чтобы он поворачивал налево, если нажать кнопку датчика касания, и направо, если вы «активировали» датчик цвета? Снимите оба датчика с их креплений, чтобы вы могли держать их в руках. Затем подключите их к модулю EV3, используя самые длинные кабели, какие у вас есть. Как вы поступите, чтобы датчик цвета обнаруживал прикосновение ваших пальцев?

СОВЕТ **Что такое яркость отраженного света, когда вы поместите палец на датчик?**

ПРАКТИКУМ № 39: ЦВЕТОВОЙ ШАБЛОН!

Сложность: **Время:**

Дополните программу, которую вы сделали в практикуме № 37, чтобы робот начал реагировать на различные цветовые шаблоны. Например, если он фиксирует красный в течение двух секунд, пусть совершит резкий поворот влево, а если фиксирует красный в течение одной секунды, а затем синий в течение одной секунды, он должен совершить резкий поворот вправо.

СОВЕТ **Создайте программу аналогичную MorseCode.**

ПРАКТИКУМ № 40: ТРАССА С ПРЕПЯТСТВИЯМИ!

Сложность: **Время:**

Можете ли вы сделать робота, который движется по линии тестовой трассы и разворачивается, если встречает препятствие на пути? Снова закрепите датчик касания на роботе и прикрепите слева от него датчик цвета, как показано на рис. 7.18. Вы сможете установить его только на один из двух синих штифтов, но этого достаточно.

СОВЕТ **Измените блок Цикл (Loop) в программе движения по трассе так, чтобы цикл выполнялся, пока кнопка датчика касания нажата. Затем запрограммируйте робота на разворот, поиск линии трассы и движение по ней в противоположном направлении.**

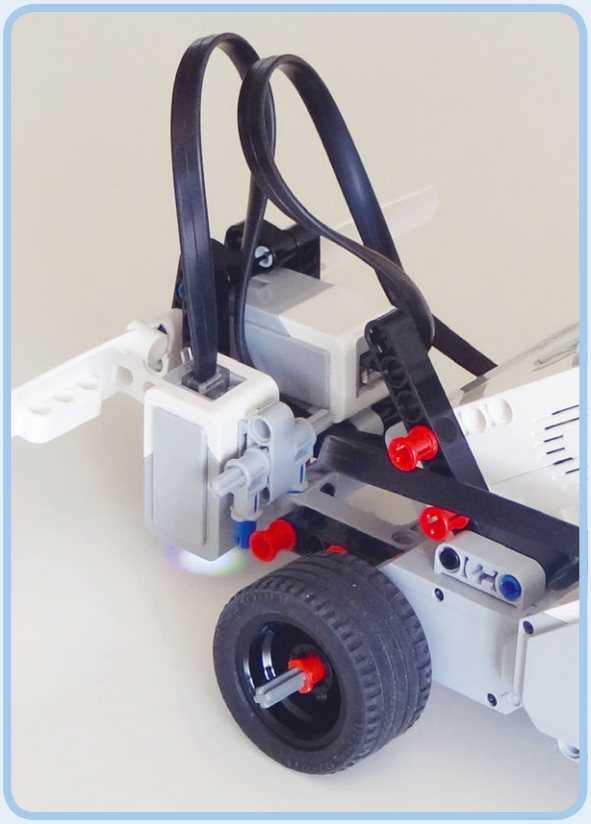

Рис. 7.18. Робот EXPLOR3R с двумя подключенными датчиками: цвета (порт 3) и касания (порт 1). Обратите внимание, что программа движения по трассе, которую вы создали ранее, должна работать с этой конфигурацией

ПРАКТИКУМ № 41: СУМАСШЕДШАЯ ДОРОЖКА!

Сложность: **Время:**

Откройте файл *Настраиваемая_трасса.pdf*, который можно скачать с сайта **eksmo.ru/files/Lego_Mindstorms_Primers.zip**, и распечатайте два угловых элемента трассы (две копии с. 3), трехсторонний перекресток (с. 3), круговой элемент (с. 17), желтый смайлик (с. 26) и зеленую звезду (с. 28), чтобы создать трассу, показанную на рис. 7.19. Запрограммируйте робота следовать по линии трассы до тех пор, пока он не встретит зеленую звезду, независимо от его исходного положения.

СОВЕТ **Запрограммируйте робота разворачиваться и следовать по трассе в противоположном направлении, если он встречает желтый смайлик.**

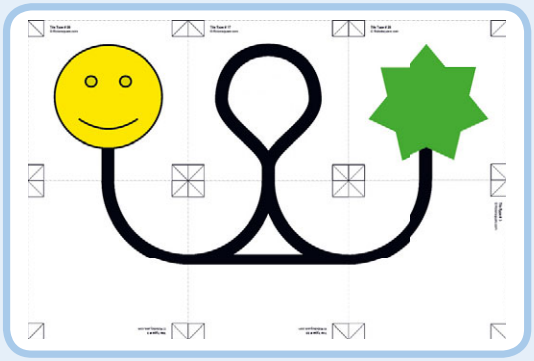

Рис. 7.19. Трасса для практикума № 41

СДЕЛАЙ САМ № 7: ДВЕРНОЙ ЗВОНОК!

Сборка: **Программирование:**

Можете ли вы сделать так, чтобы модуль EV3 воспроизводил звук, когда кто-то проходит через дверной проем? Установите датчик цвета на одной стороне дверной коробки и поместите фонарик на другой стороне, направив его прямо на датчик цвета. Как надо запрограммировать робота, чтобы он определял, когда кто-то проходит через дверь?

СОВЕТ **Когда кто-то блокирует свет от фонаря, проходя через дверной проем, яркость окружающего освещения падает.**

СДЕЛАЙ САМ № 8: ВКЛАД В БЕЗОПАСНОСТЬ!

Сборка: **Программирование:**

Смогли бы вы создать сейф для хранения ценностей, который бы открывался с помощью цветной карты безопасности, как показано на рис. 7.20? Один мотор используйте для того, чтобы перемещать каждый цветной квадрат на карточке безопасности мимо датчика цвета, а второй — чтобы открывать и закрывать сейф после успешного сканирования карты. Вы можете сканировать эту конкретную карту, используя следующие шаги.

1. Поверните колесо, пока датчик цвета не зафиксирует один из следующих цветов: красный, желтый, зеленый или синий.
2. Извлеките карту, если цвет не красный. Если красный — продолжайте.
3. Включите мотор, чтобы переместиться к следующему цвету.
4. Извлеките карту, если цвет не желтый. Если желтый — продолжайте.
5. Включите мотор, чтобы переместиться к следующему цвету.
6. И т. д.

СОВЕТ **Для каждого цветного квадрата используйте собственный блок Переключатель (Switch). Первый блок Переключатель (Switch) определяет, является ли первый цвет красным. Если это так (Истина (True)), мотор вращается и запускается следующий блок Переключатель (Switch), определяющий, является ли следующий цвет желтым. Если это так (Истина (True)), мотор вращается и так далее. Вариант Ложь (False) каждого блока Переключатель (Switch) должен содержать блоки, позволяющие извлечь карту.**

Рис. 7.20. Вы можете распечатать копию этой карты, скачав файл Карта_безопасности.pdf *с сайта* ***eksmo.ru/files/Lego_Mindstorms_Primers.zip***

8

Использование инфракрасного датчика

Инфракрасный датчик позволяет роботу «видеть» окрестности путем измерения приблизительного расстояния до объекта с помощью инфракрасных лучей. Кроме того, датчик собирает данные от *удаленного инфракрасного маяка*. Датчик определяет, какие кнопки на маяке вы нажимаете, приблизительное расстояние до маяка, а также примерное направление, или курс, от робота к маяку.

Вы можете реализовать каждую из этих функций в ваших программах с использованием инфракрасного датчика в четырех различных режимах: **Приближение** (Proximity), **Удаленный** (Remote), **Приближение маяка** (Beacon Proximity) и **Направление маяка** (Beacon Heading) (рис. 8.1). Инфракрасный свет от маяка можно увидеть, глядя на него через цифровую камеру, например смартфона. Кроме того, можно увидеть слабый свет светодиода датчика, если тот находится в режиме **Приближение** (Proximity). В этой главе вы узнаете, как используется каждый режим, как разработать программы для EXPLOR3R, чтобы он избегал препятствий, реагировал на удаленные команды, а также находил маяк.

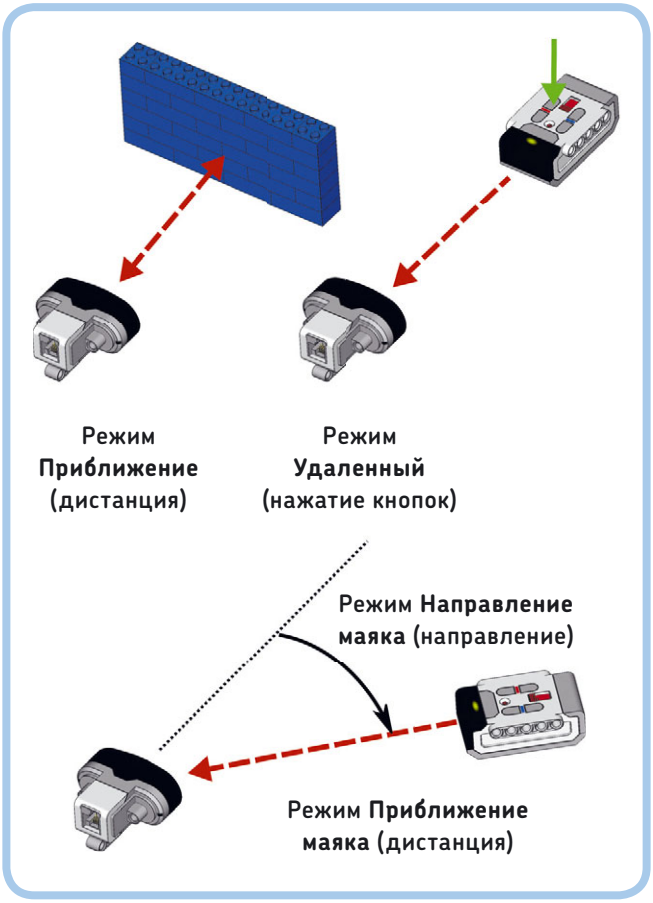

Рис. 8.1. Режимы работы инфракрасного датчика. Красные пунктирные линии обозначают невидимые лучи инфракрасного света. Если вы преградите путь между датчиком и маяком, датчик не сможет взять правильные замеры

Режим Приближение

В режиме **Приближение** (Proximity) робот измеряет расстояние от датчика до объекта. Но замеры производятся не в метрах или сантиметрах. Датчик определяет расстояние в процентах от 0% (очень близко) до 100% (очень далеко). Чтобы просмотреть показания датчика, запустите приложение **Port View** на модуле EV3, выберите входной порт 4, а затем — пункт **IR-PROX** (сокращение от словосочетания infrared proximity — инфракрасное приближение).

Значение определяется путем измерения, какая часть инфракрасных лучей, которые испускает датчик, отражается

обратно. Чем ближе объект к датчику, тем больше света возвращается. Некоторые поверхности отражают инфракрасный свет лучше, чем другие, поэтому кажется, что они находятся ближе. Например, расстояние до белой стены может казаться меньше, чем до черной, даже если они на самом деле на одинаковом расстоянии от датчика.

Избегание препятствий

Хотя датчик измеряет расстояние до объекта не совсем точно, он хорошо определяет, есть препятствие на пути или нет. Значение показаний датчика составляет 100%, когда он ничего не обнаружил, а если робот приближается к стене, значение уменьшается примерно до 30%. Чтобы робот избегал любые препятствия, нужно запрограммировать его на движение вперед, но, если он фиксирует объект, находящийся ближе, чем на 65% видимости (пороговое значение), он должен объехать его. В программе ProximityAvoid, показанной на рис. 8.2, используется блок **Ожидание** (Wait) для отслеживания значений менее 65%.

Совместное использование датчиков

Вы не ограничены применением только одного датчика. На самом деле, вы можете использовать все датчики, подключенные к EV3, в одной программе. Так вы получите более точные результаты замеров.

К примеру, инфракрасный датчик не всегда «видит» мелкие предметы на пути робота. Обнаружить их поможет датчик касания. С другой стороны, датчик касания не определит некоторые мягкие препятствия, например шторы, а инфракрасный датчик с этой задачей справится. Если комбинировать измерения датчиков, вероятность столкновения с препятствием у робота EXPLOR3R будет намного ниже.

Один из способов комбинации измерений датчиков заключается во вложении одного блока **Переключатель** (Switch) в другой, как показано на рис. 8.3. Подготовьте и запустите программу CombinedSensors, которая воспроизведет эту схему на практике, как показано на рис. 8.4.

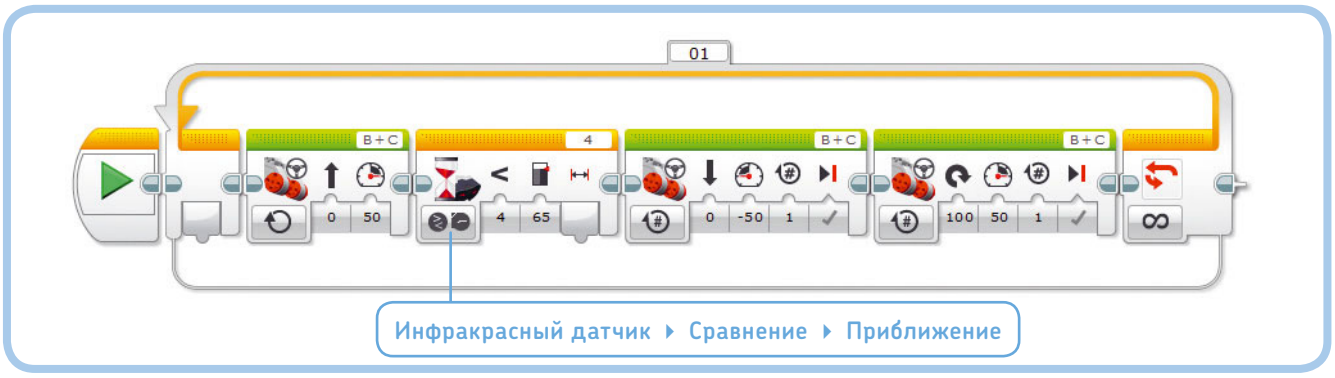

Рис. 8.2. Создайте новый проект под названием EXPLOR3R-IR и программу ProximityAvoid. Настройте блоки, как показано на рисунке. Обратите внимание на сходство с программой TouchAvoid, которую вы разработали в главе 6

ПРАКТИКУМ № 42: ИДУ НА СБЛИЖЕНИЕ!

Сложность: **Время:**

Запрограммируйте робота повторять слово «Обнаружено»*, если он заметил объект ближе 50%, в противном случае он должен произносить «Поиск»**. Потом попробуйте использовать другие пороговые значения, например 5 или 95%, чтобы разобраться, насколько близкие/далекие препятствия датчик может уверенно обнаруживать. Датчик не определяет точное расстояние, и вы увидите, что результаты варьируются в зависимости от того, какой тип объекта вы пытаетесь зафиксировать.

СОВЕТ **Вам нужно вложить блок Переключатель (Switch) в блок Цикл (Loop).**

ПРАКТИКУМ № 43: ТРИ ДАТЧИКА!

Сложность: **Время:**

Дополните программу CombinedSensors третьим датчиком. Задайте роботу такое поведение, чтобы он стоял на месте, если датчик цвета фиксирует синий объект, а при удалении синего объекта начинал движение, избегая препятствий.

* Пункт **Detected** в каталоге звуков Lego.
** Пункт **Searching** в каталоге звуков Lego.

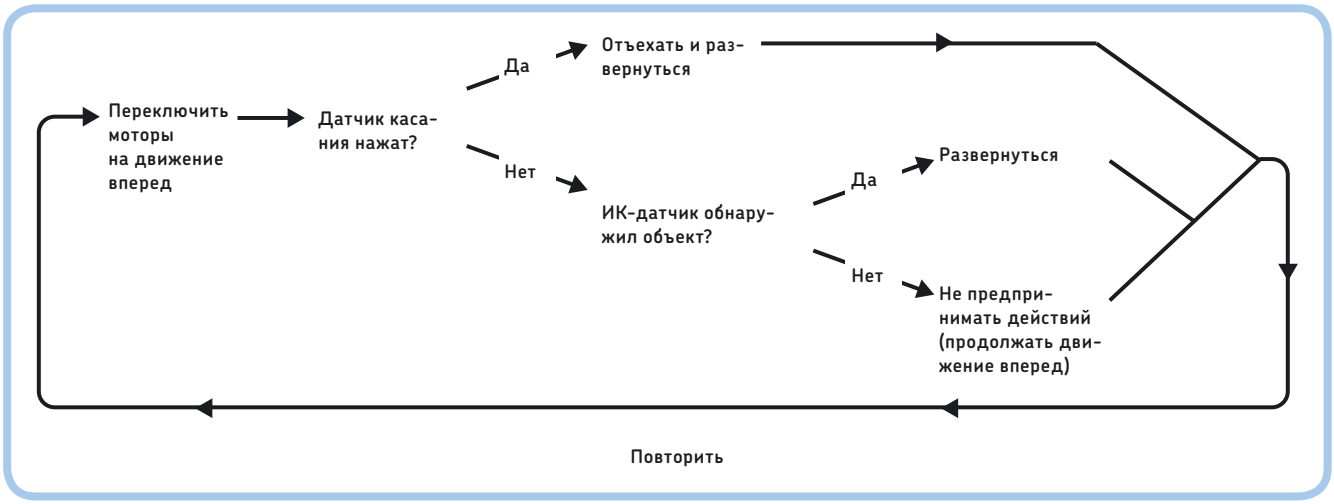

Рис. 8.3. Схема работы программы CombinedSensors

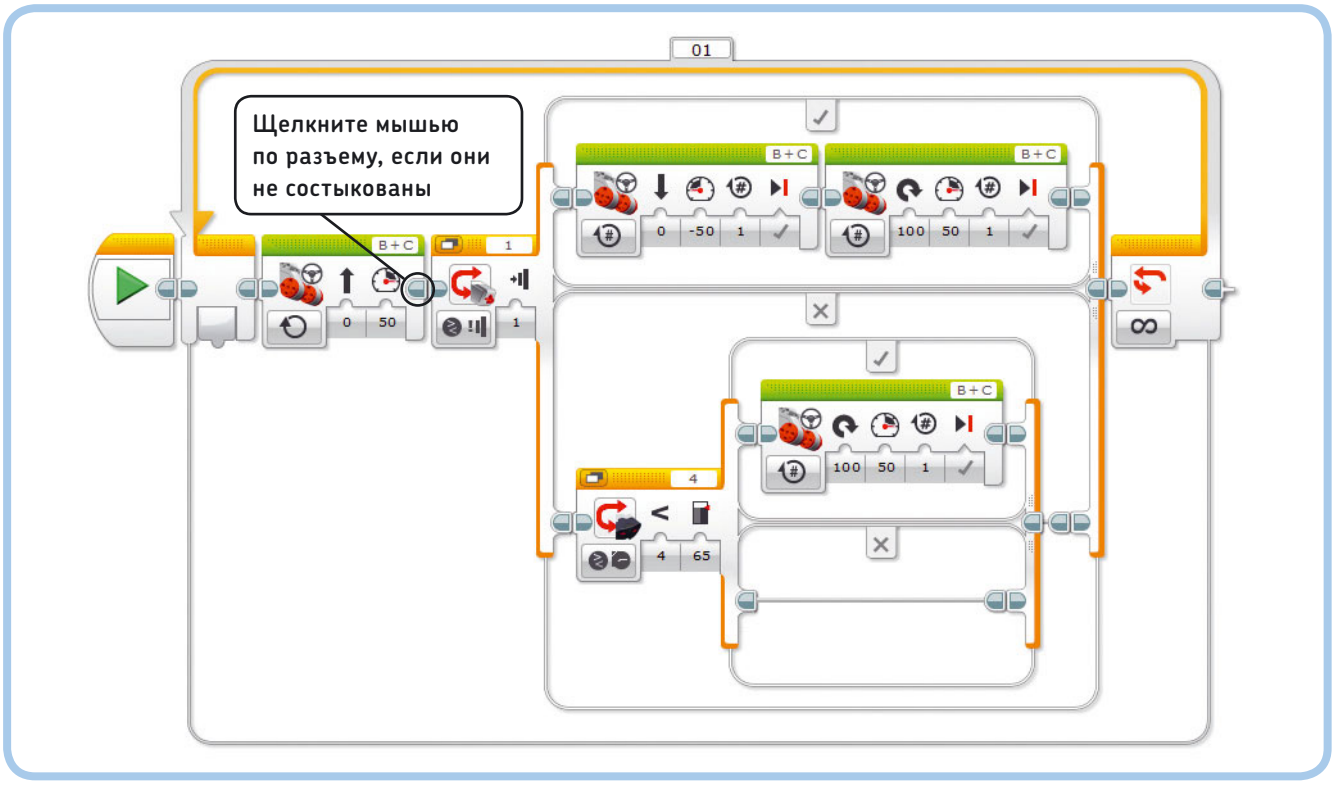

*Рис. 8.4. Программа CombinedSensors. Измените размер блоков **Переключатель** и при необходимости зациклите*

ПРИМЕЧАНИЕ Не забудьте подключить датчик касания, которым вы оборудовали робота в главе 6. Датчик касания должен быть подключен к входному порту 1.

Режим Удаленный

В режиме **Удаленный** (Remote) инфракрасный датчик фиксирует, какие кнопки нажаты на удаленном инфракрасном маяке, что позволяет разрабатывать программы, которые выполняют разные операции при нажатии каждой кнопки. Это похоже на то, как вы удаленно управляли роботом в главе 2. Приложение **IR Control**, предустановленное в модуле EV3, позволяет роботу резко поворачивать в разных направлениях согласно нажатиям кнопок на маяке. Датчик различает 12 комбинаций кнопок, или *идентификаторов*, как показано на рис. 8.5.

Блок **Переключатель** (Switch) в режиме **Измерение** (Measure) позволяет выбрать последовательность блоков, которую нужно запустить, для каждого значения, полученного от датчика, — каждого *идентификатора*. Программа CustomRemote использует этот переключатель, чтобы робот EXPLOR3R двигался вперед, когда нажаты две верхние кнопки (идентификатор 5). Робот поворачивает налево, когда нажата верхняя левая кнопка (идентификатор 1), и направо, если нажата верхняя правая кнопка (идентификатор 3). Если ни одна кнопка не нажата (идентификатор 0), робот прекращает движение. Бездействие — состояние по умолчанию, поэтому робот не двигается и при нажатии неопределенной комбинации кнопок.

Поскольку программа настроена для ожидания сигнала маяка на канале 1, вам нужно выбрать его на маяке (см. рис. 2.10). Если у вас есть еще один робот EV3, переключите его на другой канал (2, 3 или 4), чтобы избежать помех. Теперь создайте и запустите программу (рис. 8.6) и начните управлять вашим роботом дистанционно с помощью маяка.

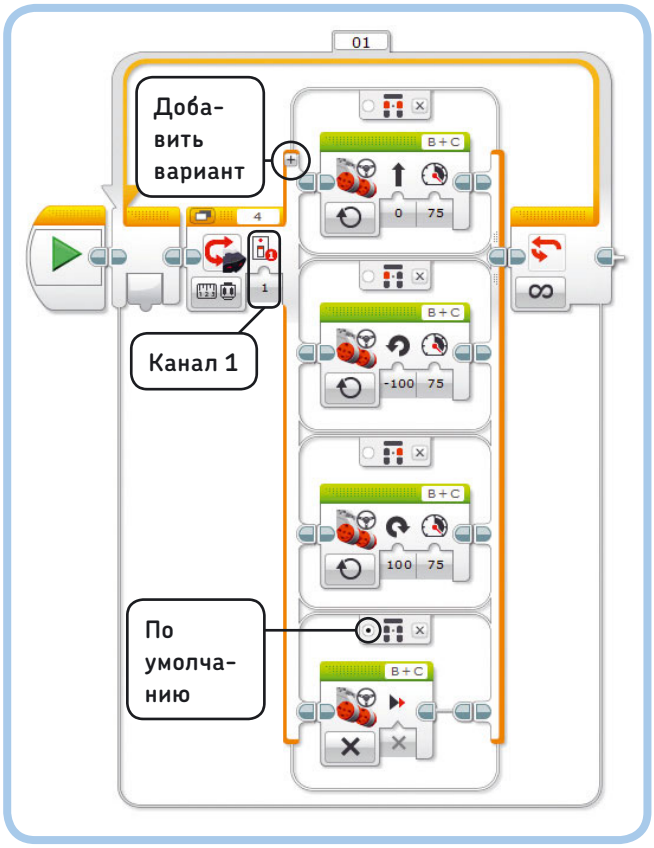

Рис. 8.6. Программа CustomRemote

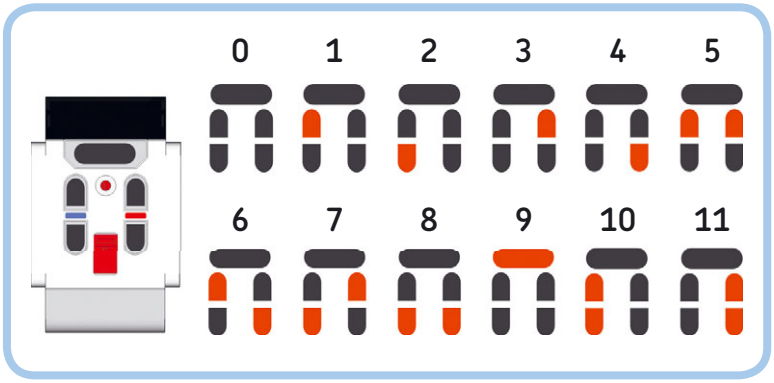

Рис. 8.5. Инфракрасный датчик определяет 12 комбинаций кнопок (идентификаторов), *которые можно нажать на удаленном инфракрасном маяке. Нажатые кнопки окрашены в красный цвет*

ПРАКТИКУМ № 44: РАЗБЛОКИРУЙ ДИСТАНЦИОННОЕ УПРАВЛЕНИЕ!

Сложность: **Время:**

Можете ли вы защитить вашу программу с помощью секретной комбинации кнопок? В программе CustomRemote прямо перед блоком **Цикл** (Loop) добавьте два блока **Ожидание** (Wait). Настройте эти блоки на ожидание нажатия идентификаторов 10 и 11 в указанном порядке, до того как остальная часть программы может быть выполнена. В качестве дополнительного задания попробуйте сделать программу более защищенной, используя способ, описанный в практикуме «Сделай сам № 8» на с. 110.

Этот способ особенно полезен (и весел!), потому что теперь вы можете сделать программу дистанционного управления для всех ваших роботов. Например, в главе 12 вы соберете гоночную машину EV3, которая ездит и управляется рулем иначе, чем EXPLOR3R. Стандартное приложение **IR Control** не сможет работать с этим роботом. Но вы можете решить эту проблему, создав собственную программу, позволяющую управлять этой машиной.

Режим Приближение маяка

Помимо определения кнопок, которые вы нажимаете на маяке, инфракрасный датчик может распознать мощность сигнала и направление, в котором сигнал поступает при нажатии любой кнопки. Робот может использовать эту информацию, чтобы определить расположение маяка и двигаться к нему.

В режиме **Приближение маяка** (Beacon Proximity) датчик использует мощность сигнала маяка для расчета относительного расстояния в диапазоне от 1% (маяк очень близко к датчику) до 100% (очень далеко). Для достижения наилучших результатов держите маяк примерно на такой же высоте или чуть выше датчика и направьте его в глаза робота (см. рис. 8.1).

Зеленый индикатор на маяке указывает на то, что он отсылает сигнал. Не имеет значения, какую кнопку вы нажимаете в режиме приближения или направления, но удобнее всего использовать кнопку в верхней части (идентификатор 9). Зеленый индикатор включается при однократном нажатии кнопки и выключается при повторном нажатии, поэтому вам не нужно удерживать кнопку.

Теперь вы создадите программу BeaconSearch1, согласно которой робот неоднократно произносит слово «Поиск»*, пока замеряемое инфракрасным датчиком расстояние составляет менее 10%. Другими словами, робот будет повторять слово «Поиск», пока маяк находится очень близко от датчика. Вы можете сделать это с помощью блока **Цикл** (Loop), настроенного, как показано на рис. 8.7. После завершения цикла робот говорит «Обнаружено»**.

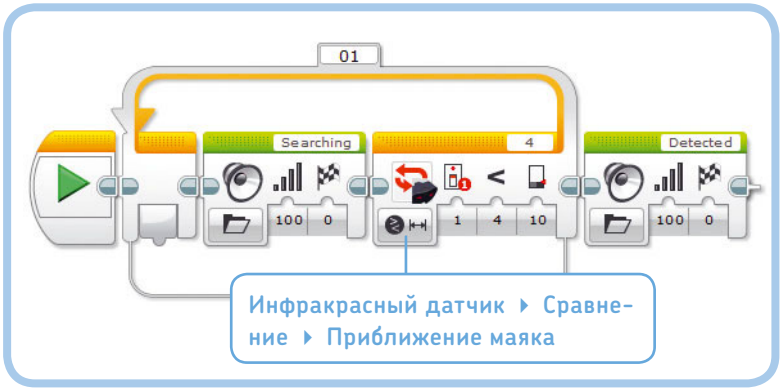

Рис. 8.7. Программа BeaconSearch1

ПРИМЕЧАНИЕ **Датчик может сообщать, обнаружен ли сигнал вообще. Вы можете сделать, к примеру, так, что робот остановится, если датчик перестал фиксировать сигнал. Подобная программа требует применения некоторых незнакомых вам блоков, которые мы рассмотрим в главе 14.**

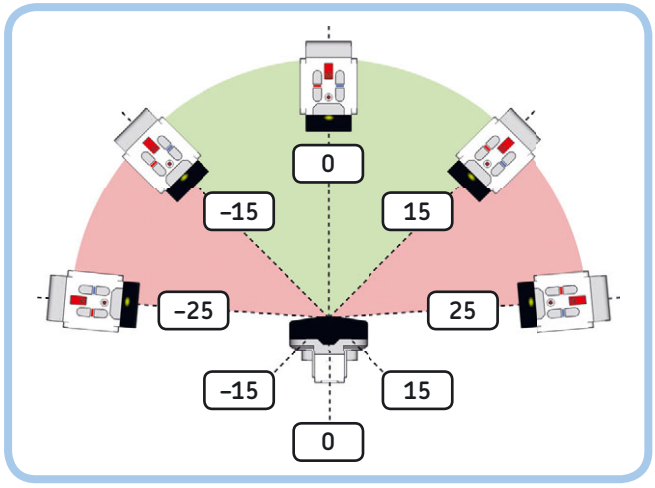

*Рис. 8.8. Диапазон допустимых значений в режиме **Направление маяка** находится в пределах от –25 до 25. Отрицательные значения указывают, что маяк слева от датчика; положительные — что он справа. Значение, близкое к нулю, говорит о том, что робот обнаружил маяк прямо впереди или позади себя*

Режим Направление маяка

Инфракрасный датчик может измерять направление сигнала маяка, используя режим **Направление маяка** (Beacon Heading). Этот режим дает роботу приблизительное представление об угле между маяком и датчиком. Результат измерения представляет собой число в диапазоне от –25 до 25, как показано

* Пункт **Searching** в каталоге звуков Lego.
** Пункт **Detected** в каталоге звуков Lego.

на рис. 8.8. Датчик не определяет угол с точностью, но делает это достаточно хорошо, чтобы распознать, находится ли маяк слева от робота (отрицательные значения) или справа (положительные значения).

Датчик способен обнаружить маяк во всех направлениях, даже если он находится позади, но самое надежное измерение получается в зеленой области, схематически представленной на рис. 8.8. Значение, зафиксированное датчиком, будет стремиться к нулю, если маяк находится прямо перед датчиком или прямо позади него, но оно равно нулю и тогда, когда датчик вообще не обнаруживает сигнал.

Информация о том, находится ли маяк слева или справа от робота, — все, что вам нужно, чтобы робот направился в сторону маяка. Когда датчик определяет, что маяк слева, робот поворачивает налево, а если справа, то направо. Вы можете использовать блок **Переключатель** (Switch), чтобы определять замер датчика меньше (<) 0, информирующий о том, что маяк находится слева.

Если постоянно корректировать направление движения, основываясь на измерениях датчика и одновременно двигаясь вперед, ваша программа успешно справится с поиском маяка. Удалите блоки **Звук** (Sound) из предыдущей программы и добавьте один блок **Переключатель** (Switch) и два блока **Рулевое управление** (Move Steering) для доработки программы BeaconSearch2, как показано на рис. 8.9. Используемый в программе блок **Цикл** (Loop) инструктирует робота продолжать поиск маяка до тех пор, пока расстояние до него будет не менее 10%; при достижении этого результата программа завершает работу.

ПРАКТИКУМ № 45: ПЛАВНОЕ ПРЕСЛЕДОВАНИЕ!

Сложность: **Время:**

Можете ли вы дополнить программу BeaconSearch2 так, чтобы робот двигался к маяку плавнее? Робот совершает плавные повороты (значение 25% параметра **Рулевое управление** (Move Steering)), если маяк находится в зеленой зоне (см. рис. 8.8), и резкие повороты (значение 50% параметра **Рулевое управление** (Move Steering)), если маяк находится в красной зоне.

СОВЕТ **Используйте способы, которые вы изучили в разделе «Плавное движение по трассе» в главе 7. Вам не нужно рассчитывать пороговые значения; они приведены на рис. 8.8.**

ПРИМЕЧАНИЕ **Не забудьте сделать так, чтобы маяк передавал сигнал непрерывно, либо удерживайте одну из кнопок нажатой, либо переключите кнопку в верхней части (идентификатор 9), чтобы зеленый индикатор горел постоянно.**

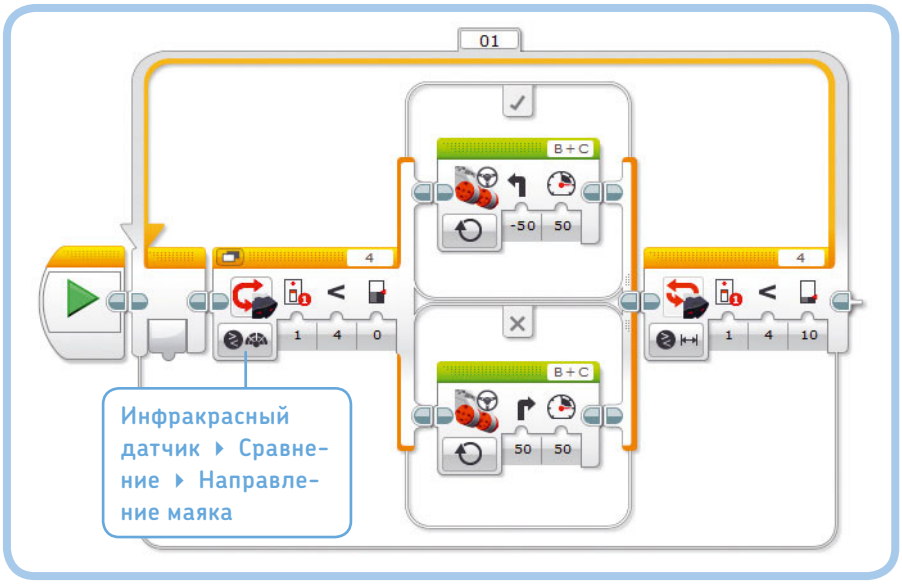

Рис. 8.9. Программа BeaconSearch2 заставляет робота двигаться к маяку. Когда маяк окажется близко — робот остановится. Если маяк все время перемещать, робот будет следовать за вами

Комбинированный режим работы датчика

Объединение нескольких режимов работы инфракрасного датчика в одной программе чревато тем, что ваш робот будет вести себя непредсказуемо, поскольку датчику нужно время, чтобы переключиться с одного режима на другой. **Приближение маяка** (Beacon Proximity) и **Направление маяка** (Beacon Heading) — это единственные режимы работы инфракрасного датчика, которые можно использовать вместе без задержек.

К примеру, нужно изменить блок **Цикл** (Loop) в программе CustomRemote (см. рис. 8.6), повторяя цикл до тех пор, пока датчик не определит значение приближения менее 10%. Блок **Переключатель** (Switch) определяет нажатие кнопки в режиме **Удаленный** (Remote), а затем блок **Цикл** (Loop) считывает значение приближения в режиме **Приближение** (Proximity). Но, поскольку датчик должен каждый раз переходить из одного режима в другой, программа будет работать очень медленно, и ваш робот не сможет должным образом реагировать на маяк. (Попробуйте, если вас это устраивает.)

Если время не имеет для ваших задач решающего значения, то использование различных режимов в одной программе — это замечательно. Например, программа MultiMode, показанная на рис. 8.10, работает именно так. Сначала она ждет, пока вы нажмете правую верхнюю кнопку на маяке (идентификатор 3), после чего воспроизводится звуковой сигнал. Потом робот говорит: «Да»*, если значение приближения ниже 30%; в противном случае он говорит: «Нет»**. Между звуковым сигналом и произнесенным словом будет пауза — это задержка, вызванная переключением режимов с **Удаленный** (Remote) на **Приближение** (Proximity).

Дальнейшее изучение

Инфракрасный датчик позволяет вашему роботу обнаруживать окружающие его объекты на расстоянии. В сочетании с удаленным инфракрасным маяком датчик может выступать

ПРАКТИКУМ № 46: СЛЕДУЙ ЗА МНОЙ!

Сложность: **Время:**

Можете ли вы запрограммировать робота EXPLOR3R следовать за вами по прямой, сохраняя постоянную дистанцию? Используйте инфракрасный датчик в режиме **Приближение** (Proximity) для определения расстояния до вашей руки (держите ее перед роботом). Робот должен следовать за вами, когда вы перемещаете руку; если вы поднесете руку ближе к роботу, он должен откатиться назад. Пусть робот стоит на месте, если он «видит» вашу руку на расстоянии от 35 до 45%.

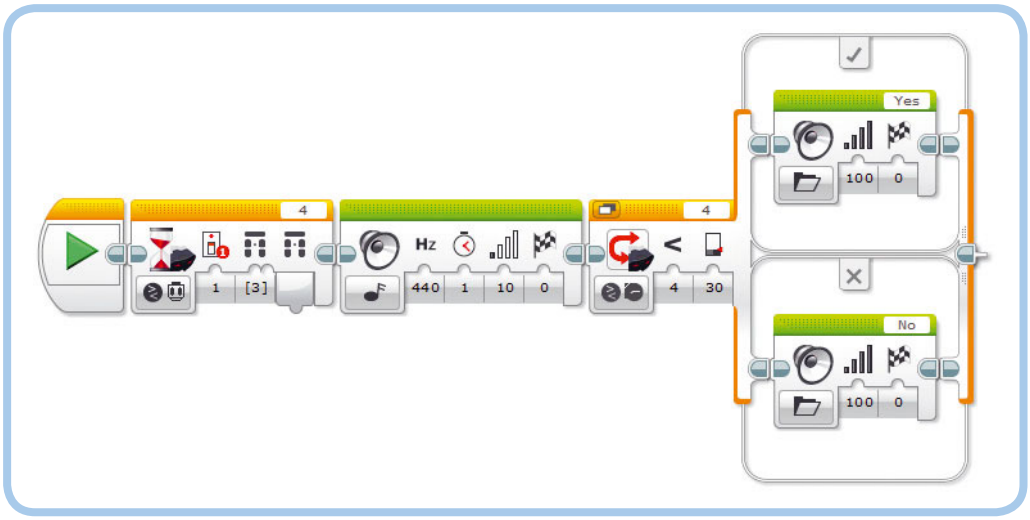

*Рис. 8.10. Программа MultiMode. В этом случае время выполнения программы не имеет решающего значения, поэтому допустимо использовать режимы **Удаленный** и **Приближение** в одной программе*

* Пункт **Yes** в каталоге звуков Lego.

** Пункт **No** в каталоге звуков Lego.

ПРАКТИКУМ № 47: АКУСТИЧЕСКИЙ ЛОКАТОР!

Сложность: **Время:**

Можете ли вы сделать, чтобы EV3 издавал звуки так, чтобы привести вас к маяку с закрытыми глазами? Настройте его издавать тональные звуки разной частоты и громкости в зависимости от местоположения маяка. Если маяк находится слева от датчика, следует воспроизвести звук низкой частоты (400 Гц), а если справа — звук высокой частоты (1000 Гц). Чем ближе вы подходите к маяку, тем громче должен быть звук.

СОВЕТ **Во-первых, используйте блок Переключатель (Switch), чтобы определить, с какой стороны находится маяк — слева или справа. Во-вторых, в обеих частях этого переключателя поместите блок Переключатель (Switch), чтобы определить, далеко находится маяк или близко. После этого у вас будет четыре варианта размещения блока Звук (Sound), каждый из которых настроен с возможностью воспроизведения одного из тонов: низкого и громкого, низкого и тихого, высокого и громкого и высокого и тихого.**

в качестве приемника инфракрасного сигнала и пеленгатора маяка. Кроме того, вы узнали, как объединить датчик касания и инфракрасный датчик, чтобы робот с большей эффективностью избегал препятствий. Чтобы еще больше усложнить программу, вы можете добавить датчик цвета. Прежде чем переходить к следующей главе, проверьте свои навыки по сборке и программированию, выполнив следующие практические задания!

СДЕЛАЙ САМ № 9: ЖЕЛЕЗНОДОРОЖНЫЙ ПЕРЕЕЗД!

Сборка: **Программирование:**

Сможете ли вы разработать автоматизированный железнодорожный переезд для модели поезда LEGO? Используйте мотор, чтобы поднимать шлагбаум, который препятствует автомобилю пересекать железную дорогу, когда проходит поезд. Используйте инфракрасный датчик или датчик цвета, чтобы определить, когда поезд приближается и шлагбаумы должны быть опущены, а когда они должны быть снова подняты.

СДЕЛАЙ САМ № 10: НАДЕЖНЫЙ СИГНАЛ ТРЕВОГИ!

Сборка: **Программирование:**

Можете ли вы применить все три датчика из набора MINDSTORMS EV3 для создания аварийного сигнала о вторжении, который срабатывает всегда? Используйте датчик касания для определения открытия двери (практикум «Сделай сам № 4» на с. 96), датчик цвета для обнаружения людей, перешагнувших порог (практикум «Сделай сам № 7» на с. 110), а также инфракрасный датчик в режиме **Приближение** (Proximity) для распознавания движения вблизи определенного объекта, например смартфона.

СОВЕТ **Настройте вашего робота и программу таким образом, чтобы вы (и только вы!) могли войти в комнату без тревожного сигнала.**

9

Использование кнопок модуля EV3 и датчиков вращения мотора

Помимо датчиков касания, цвета и инфракрасного датчика, EV3 содержит два типа встроенных датчиков: кнопки модуля и датчики вращения мотора. Вы можете использовать кнопки на модуле EV3, чтобы управлять программой в процессе ее работы. Например, программа может вывести запрос нажать кнопку, предлагая выбрать один из вариантов дальнейшего поведения робота.

Каждый мотор EV3 содержит встроенный датчик вращения, который определяет позицию мотора. Это позволяет точно управлять колесами или другими механизмами. Кроме того, датчик измеряет скорость вращения мотора, позволяя выявить ситуации, когда мотор вращается быстрее или медленнее, чем предполагалось.

Использование кнопок модуля EV3

Вы можете использовать кнопки «Вверх», «Вниз», «Влево», «Вправо» и центральную кнопку модуля EV3 в ваших программах по аналогии с датчиком касания. Например, вы можете сделать так, чтобы ваш робот реагировал воспроизведением звука на нажатие конкретной кнопки. А можно настроить робота на ожидание момента, когда кнопка будет отпущена или щелкнута (нажата и сразу же отпущена).

Еще один интересный способ использования кнопок модуля EV3заключается в создании экранного меню, позволяющего выбрать следующее действие в программе. Программа ButtonMenu на рис. 9.1 воспроизводит один из трех звуков в зависимости от того, какая кнопка нажата.

Два блока **Экран** (Display) отображают простое меню на экране, предлагая пользователю выбрать, следует ли роботу сказать: «Привет», «О'кей» или «Да»*. Затем блок **Ожидание** (Wait) (в режиме **Кнопки управления модулем** ▸ **Сравнение** (Brick Buttons ▸ Compare)) приостанавливает выполнение

* Пункты **Hello**, **Okay** и **Yes** в каталоге звуков Lego.

ПРАКТИКУМ № 48: ДЛИННОЕ СООБЩЕНИЕ!

Сложность: **Время:**

При отображении длинного сообщения на экране модуля EV3 оно может не вместиться полностью. Создайте программу, которая позволит использовать кнопку «Вниз» для прокрутки сообщения.

СОВЕТ **Научите робота показывать на экране какой-либо новый текст при каждом нажатии кнопки.**

ПРАКТИКУМ № 49: ПОЛЬЗОВАТЕЛЬСКОЕ МЕНЮ!

Сложность: **Время:**

Можете ли вы дополнить программу ButtonMenu, чтобы ваш робот выполнял какие-то полезные действия, помимо воспроизведения звуков? Возьмите три программы, которые вы сделали ранее, превратите их в контейнеры «Мой блок», а затем поместите их в блок **Переключатель** (Switch) программы ButtonMenu. Измените настройки блоков **Экран** (Display), чтобы они демонстрировали то, что происходит при нажатии на каждую кнопку.

СОВЕТ **Этот метод часто используется в соревнованиях робототехники, поскольку обеспечивает быстрый способ запуска различных программ. Чтобы изменить действия каждой подпрограммы, просто измените блоки во всех контейнерах «Мой блок».**

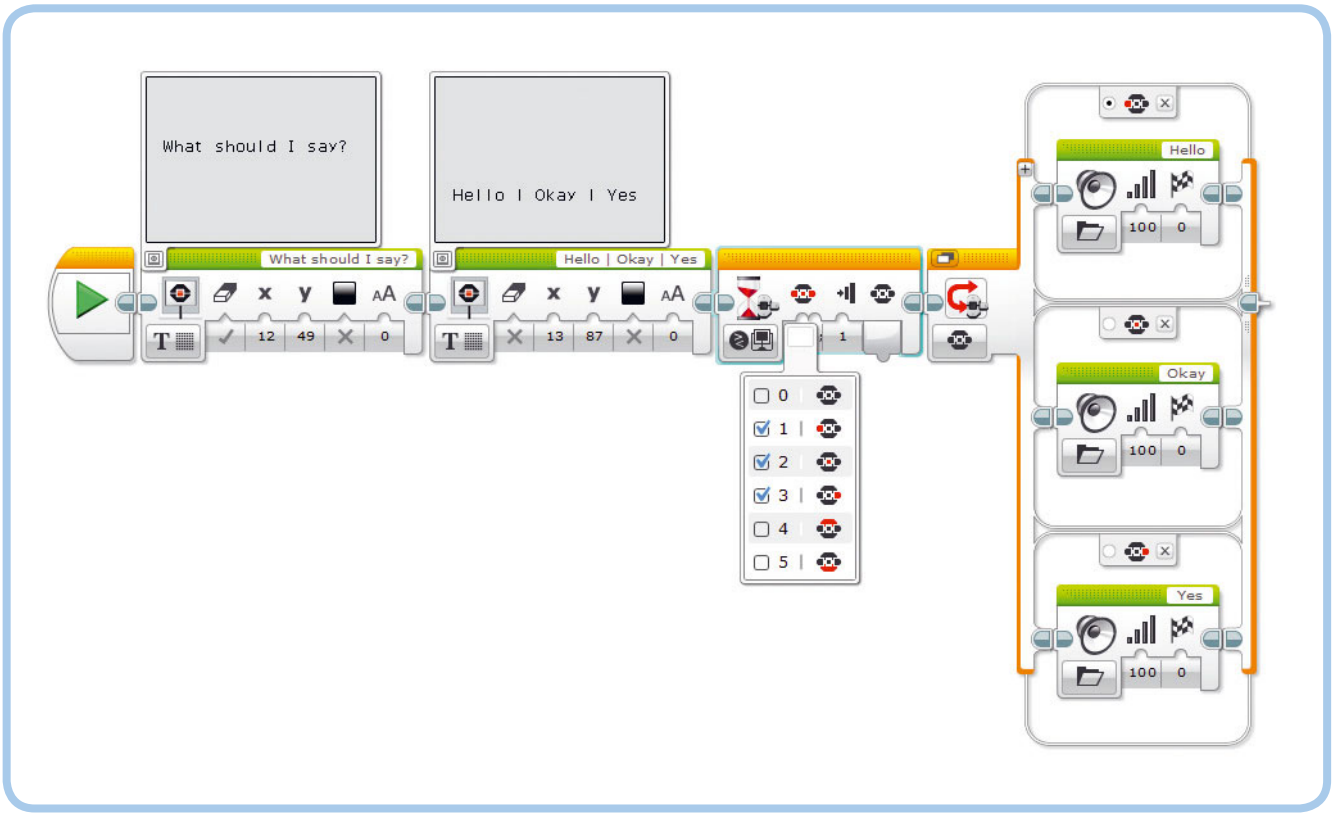

*Рис. 9.1. Программа ButtonMenu. Блок **Ожидание** настроен на режим **Кнопки управления модулем** ▸ **Сравнение** ▸ **Кнопки управления модулем**, а блок **Переключатель** — на режим **Кнопки управления модулем** ▸ **Измерение** ▸ **Кнопки управления модулем***

программы до тех пор, пока пользователь не нажмет одну из кнопок: «Влево», «Вправо» или центральную.

Затем блок **Переключатель** (Switch) (в режиме **Кнопки управления модулем** ▸ **Измерение** (Brick Buttons ▸ Measure)) определяет, какая кнопка нажата, и робот воспроизводит требуемый звук. После того как блок **Ожидание** (Wait) завершил работу, блок **Переключатель** (Switch) запускается настолько быстро, что уже во время нажатия кнопки он проверяет ее состояние. Даже если вы тотчас отпустили ее.

Использование датчика вращения мотора

Когда вы инструктируете робота двигаться вперед на три оборота мотора посредством блока **Рулевое управление** (Move Steering), он «знает», когда надо прекратить движение, поскольку датчик вращения в каждом моторе сообщает модулю EV3, сколько выполнено оборотов. Программа также может сообщить, насколько быстро в настоящее время крутится мотор, измеряя, как быстро изменяется его позиция.

Вы можете использовать блоки **Ожидание** (Wait), **Цикл** (Loop) и **Переключатель** (Switch) в режиме **Вращение мотора** (Motor Rotation) для измерения позиции мотора (режим **Градусы** (Degrees) или режим **Обороты** (Rotations)) и скорости вращения мотора (режим **Текущая мощность** (Current Power)).

Положение мотора

Положение мотора говорит вам о том, сколько он совершил оборотов с того момента, как вы запустили программу. Используйте приложение Port View в модуле EV3, перейдите к мотору B или C и вращайте моторы вручную, чтобы увидеть замеры датчика.

При первом запуске программы Port View (или собственной программы) значение датчика вращения будет равно нулю. Значение становится положительным при вращении мотора вперед и отрицательным, если вы включите его в обратном направлении после позиции 0, как показано на рис. 9.2. Например, если вращать мотор вперед на 90 градусов, а затем назад на один оборот (360 градусов), мотор передаст данные о позиции –270 градусов.

Вы можете использовать замеры позиции мотора, чтобы создать программу, которая воспроизводит звук, если вы

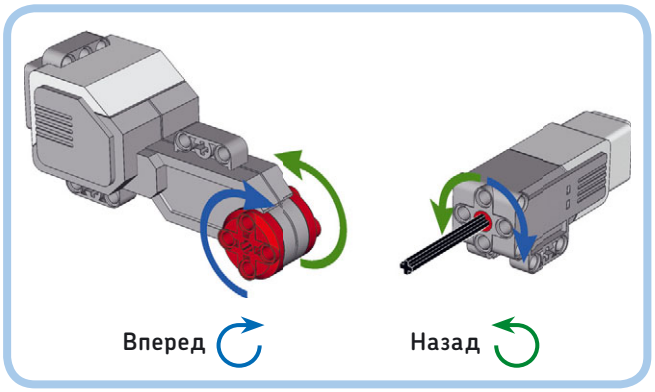

Рис. 9.2. Если вы программируете большой или средний мотор на движение вперед, он вращается в направлении, показанном синей стрелкой, и значение датчика вращения мотора становится положительным

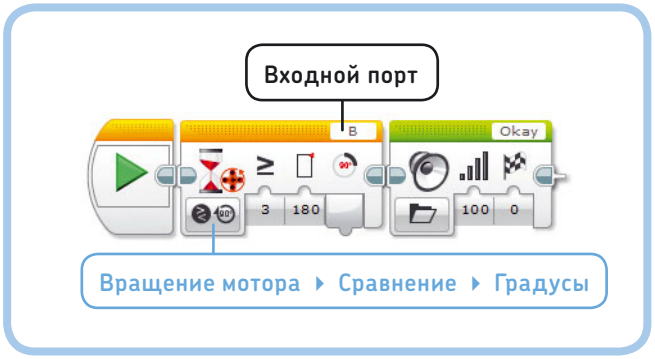

*Рис. 9.3. Программа HandRotate инструктирует робота произнести слово «О'кей», как только мотор B повернется в прямом направлении на 180 градусов. Обратите внимание, что блок **Ожидание** выполнит то же самое, если выбрать режим **Вращение мотора** ▸ **Сравнение** ▸ **Обороты** со значением параметра **Пороговое значение** равным 0,5*

вручную вращаете одно колесо на 180 градусов вперед, как показано на рис. 9.3. Блок **Ожидание** (Wait) в режиме **Вращение мотора** ▸ **Сравнение** ▸ **Градусы** (Motor Rotation ▸ Compare ▸ Degrees) ожидает, пока значение датчика вращения мотора не станет больше или равно (≥) 180 градусам. Поскольку эти датчики встроены в моторы EV3, они всегда подключены к выходным портам. В этой программе мотор используется на выходном порте B.

Сброс показаний датчика вращения мотора

Теперь предположим, что вы хотите повторить действия, описанные в программе HandRotate, с блоком **Цикл** (Loop) так, чтобы звук воспроизводился снова, когда вы провернете колесо еще на 180 градусов. Первый звук воспроизводится при вращении мотора вперед на 180 градусов. Но во время второго запуска цикла звук будет воспроизводиться сразу, потому что значение датчика уже больше 180 градусов, а это совсем не то, что нужно в результате.

Решение состоит в том, чтобы сбросить значение датчика вращения до нуля в начале цикла, используя для этого блок **Вращение мотора** (Motor Rotation) в режиме **Сброс** (Reset), как показано на рис. 9.4. Другие особенности этого и иных блоков датчиков описаны в главе 14.

Запустите программу и убедитесь, что вы слышите звук однократно каждый раз, когда колесо поворачивается на 180 градусов вперед.

Скорость вращения

Датчик вращения определяет, как быстро мотор вращается в диапазоне значений от –100 до 100%, основываясь на скорости, с которой изменяется позиция мотора. Значение положительно, когда мотор вращается вперед (синяя стрелка

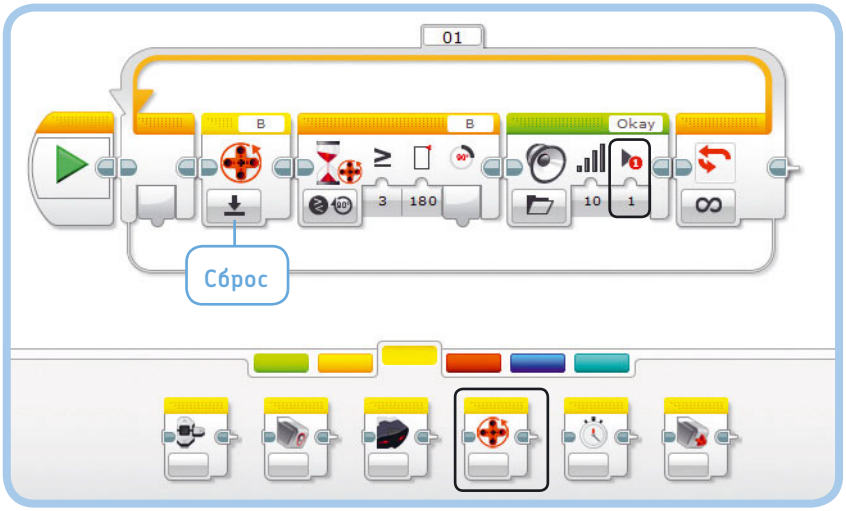

*Рис. 9.4. Программа HandRotateReset присваивает датчику значение 0 в начале каждого цикла с помощью блока **Вращение мотора**, работающего в режиме **Сброс**. Обратите внимание, что параметру **Тип воспроизведения** в блоке **Звук** присвоено значение **Воспроизвести один раз (1)**, поэтому программа не дожидается окончания воспроизведения звука*

ПРАКТИКУМ № 50: ВЕРНЕМСЯ К НАЧАЛУ!

Сложность: **Время:**

Можете ли вы создать программу, которая возвращает мотор в ту позицию, в какой он был при запуске программы? Робот должен предоставлять вам пять секунд, чтобы вы могли повернуть мотор в случайное положение вручную, а затем мотор должен вернуться к исходной точке. Используйте схему, показанную на рис. 9.5, в качестве руководства по разработке программы.

Рис. 9.5. Блок-схема для практикума № 50. Как робот определяет, что мотор был повернут в обратном направлении?

на рис. 9.2), и отрицательно, когда мотор вращается назад (зеленая стрелка). Нулевое значение датчика появится в том случае, когда мотор не вращается.

Для большого мотора значение **Текущая мощность** (Current Power), равное 50%, соответствует скорости вращения 85 оборотов в минуту (об/мин). Вы можете достичь этой скорости, вращая мотор вручную или с помощью блока **Рулевое управление** (Move Steering) со значением параметра **Мощность** (Power), равным 50%.

ПРИМЕЧАНИЕ **В режиме Текущая мощность (Current Power) измеряется скорость вращения; потребление тока или мощность не измеряется!**

Вычисление скорости вращения

Вы можете рассчитать скорость, измеренную в оборотах в минуту (об/мин), используя текущее значение параметра **Текущая мощность** (Current Power) следующим образом:

большой мотор
скорость вращения (об/мин) = значение датчика × 1,70

средний мотор
скорость вращения (об/мин) = значение датчика × 2,67

Например, если текущее значение параметра **Текущая мощность** (Current Power) большого мотора составляет 30%, мотор вращается со скоростью 30 × 1,70 = 51 оборот в минуту. Поскольку один оборот в минуту соответствует шести градусам в секунду, вы можете рассчитать скорость вращения в градусах в секунду (град/с) следующим образом:

скорость вращения (град/с) = скорость вращения (оборотов в минуту) × 6

В этом примере вы получите 51 × 6 = 306 градусов в секунду.

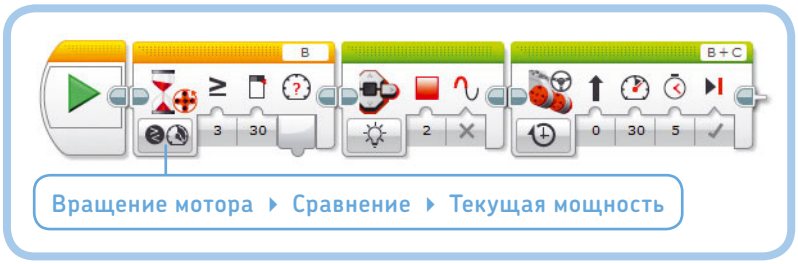

Рис. 9.6. Программа PushToStart

ПРАКТИКУМ № 51: ЦВЕТНАЯ СКОРОСТЬ!

Сложность: **Время:**

Создайте программу, которая непрерывно меняет цвет индикатора состояния модуля на зеленый, если мотор B вращается вперед, на оранжевый, если он вращается в обратном направлении, и на красный, если мотор остановлен. Поверните мотор B вручную, чтобы протестировать вашу программу.

СОВЕТ **Вам понадобится блок Цикл (Loop), два блока Переключатель (Switch) и три блока Индикатор состояния модуля (Brick Status Light).**

Измерение скорости вращения в программе

Для измерения скорости вращения в программе используется режим **Текущая мощность** (Current Power) датчика вращения, как показано на рис. 9.6. Программа PushToStart содержит блок **Ожидание** (Wait), настроенный на приостановку программы, пока мотор B не достигнет значения датчика 30% (51 об/мин). Затем запускается блок **Рулевое управление** (Move Steering). Запустите программу и вручную направляйте EXPLOR3R вперед, пока он не начнет двигаться самостоятельно.

Управление скоростью

До сих пор вы использовали по-разному настроенные зеленые блоки **Рулевое управление** (Move Steering), чтобы заставить робота двигаться. С помощью этих блоков моторы вращаются с постоянной, *регулируемой* скоростью. Когда вращение моторов замедляется из-за препятствия или наклонной поверхности, EV3 подает некоторую дополнительную мощность на мотор, чтобы сохранить нужную скорость вращения. Настройка параметра **Мощность** (Power) этих блоков фактически определяет *скорость*, которую моторы стараются поддерживать. То есть большой мотор, вращающийся на скорости 20% (34 оборота в минуту), при преодолении препятствия может потреблять больше энергии, чем при движении без затруднений на скорости 40% (68 оборотов в минуту).

Если вы не хотите, чтобы EV3 использовал эту дополнительную мощность для поддержания постоянной скорости вращения, вы можете использовать *нерегулируемую скорость*.

Регулирование скорости в действии

Чтобы увидеть разницу между регулируемой скоростью и нерегулируемой скоростью, давайте создадим программу, с помощью которой робот въезжает вверх по наклонной поверхности, например по столу, одна сторона которого приподнята. Сначала в течение трех секунд робот едет с нерегулируемой скоростью, а затем в течение трех секунд он движется с регулируемой скоростью.

Вам понадобятся два блока **Нерегулируемый мотор** (Unregulated Motor) (по одному для каждого мотора), которые доступны на вкладке **Дополнения** (Advanced) палитры программирования. Присвойте параметру **Мощность** (Power) значение 20%, как показано на рис. 9.7. После ожидания в течение трех секунд остановите моторы, присвоив параметру **Мощность** (Power) значение 0%. Для движения с регулируемой скоростью 20% (34 оборота в минуту) в программе используется блок **Рулевое управление** (Move Steering), который вы видели раньше.

Поместите робота EXPLOR3R на наклонную поверхность и запустите программу SteepSlope. Вы увидите, что в течение первых трех секунд робот едет вверх медленно, а в течение следующих трех секунд он движется быстрее.

На первом этапе движения EV3 включает оба мотора и в течение следующих трех секунд не взаимодействует с ними. Робот движется медленно, потому что для подъема в горку требуется больше энергии, чем для езды по ровной местности. На втором этапе движения датчик вращения сообщает роботу, что он движется медленно, и это вынуждает модуль EV3 передавать некоторое количество дополнительной мощности, чтобы скорость движения не снижалась.

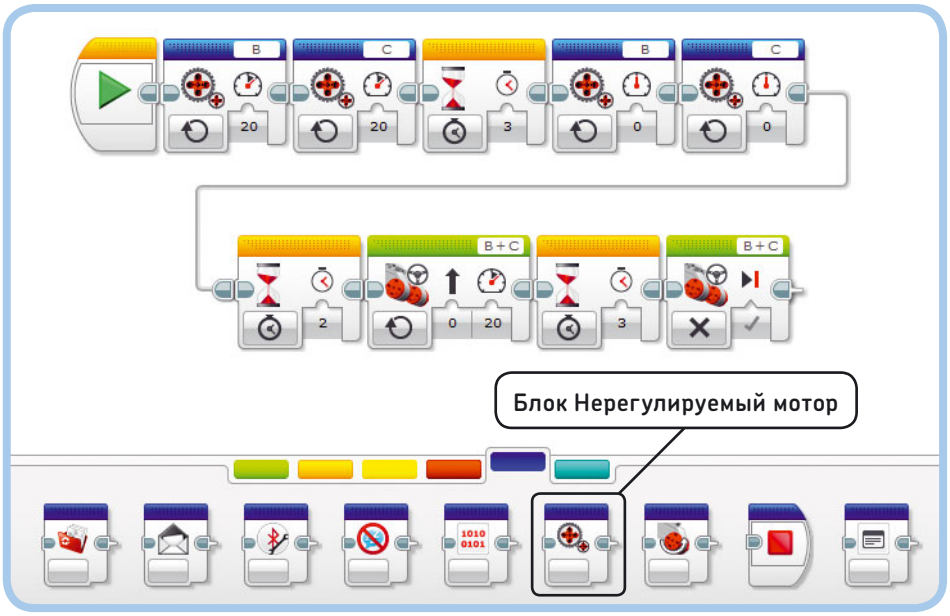

Рис. 9.7. Программа SteepSlope. Обратите внимание, что я использовал соединитель, чтобы разделить программу на две части для большей наглядности, но вам это делать необязательно

Остановка заглохшего мотора

Если во время движения замедлить одно из колес, когда запущен блок **Рулевое управление** (Move Steering), вы почувствуете, что робот пытается с усилием восстановить скорость вращения. Для колесных транспортных средств это хорошо. Но нежелательно для механизмов, которые не совершают полных оборотов, например для клешневого устройства.

Чтобы избежать этой проблемы, вы можете применить блок **Нерегулируемый мотор** (Unregulated Motor) и датчик вращения, определяющий,

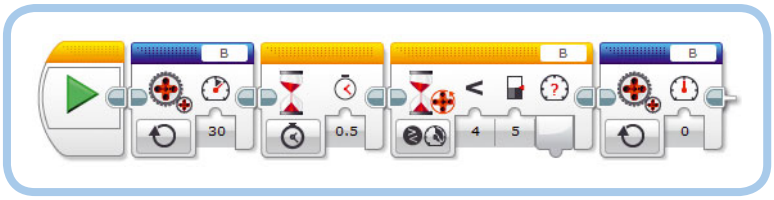

*Рис. 9.8. Программа WaitForStall крутит мотор B, пока он не заглохнет. Обратите внимание, что первый блок **Ожидание** требуется, чтобы предоставить мотору некоторое время для набора скорости, в противном случае при измерении скорости значение частоты вращения будет равно нулю, в результате чего программа немедленно завершит работу*

когда мотор заглох (блокируется), как показано в программе WaitForStall (рис. 9.8). Программа включает мотор B на 30% мощности и ожидает, когда скорость упадет ниже 5%, что свидетельствует о том, что мотор заглох. Если вы запустите эту программу на EXPLOR3R, ваш робот будет ездить по кругу, пока вы не остановите его, преградив ему путь.

Дальнейшее изучение

Теперь, когда вы научились работать со всеми датчиками из набора MINDSTORMS EV3, вы можете создавать роботов, которые взаимодействуют с окружающей средой. EXPLOR3R — это, разумеется, только один пример. По мере изучения этой книги вы научитесь собирать разных роботов с датчиками, и каждый из них будет применять датчики в своих целях.

До сих пор вы учились использовать компоненты, которые необходимы для создания работающих роботов: EV3, моторы, датчики и программное обеспечение. В следующих главах мы рассмотрим каждый компонент более подробно, так что у вас появится возможность создавать все более и более сложных (и интересных!) роботов. В следующей главе вы узнаете, как использовать детали конструктора Technic в наборе MINDSTORMS EV3 для создания собственных роботов.

Следующие практические задания помогут вам изучить больше возможностей датчиков, с которыми вы познакомились в этой главе.

ПРАКТИКУМ № 52: ДИСТАНЦИОННОЕ УПРАВЛЕНИЕ С ПОМОЩЬЮ МОДУЛЯ EV3!

Сложность: **Время:**

Снимите модуль EV3 с вашего робота (оставьте кабели подключенными) и создайте программу, которая позволяет перемещать робота с помощью кнопок на корпусе модуля EV3. Задайте роботу движение вперед при нажатии кнопки «Вверх», влево — при нажатии кнопки «Влево» и так далее.

СОВЕТ Используйте блок **Переключатель** (Switch) в режиме **Кнопки управления модулем** ▸ **Измерение** ▸ **Кнопки управления модулем** (Brick Buttons ▸ Measure ▸ Brick Buttons).

ПРАКТИКУМ № 53: СНИЖЕНИЕ СКОРОСТИ ПРИ ОБНАРУЖЕНИИ ПРЕПЯТСТВИЯ!

Сложность: **Время:**

Можете ли вы запрограммировать робота на движение по комнате и избегание препятствий без использования датчиков касания, цвета или инфракрасного датчика? Задайте роботу движение вперед, используя блоки **Нерегулируемый мотор** (Unregulated Motor), пока он не обнаружит препятствие. Затем робот должен отъехать, развернуться и продолжить движение в новом направлении.

СОВЕТ **Скорость вращения мотора снижается, когда робот натыкается на препятствие.**

СДЕЛАЙ САМ № 11: АВТОМАТИЧЕСКИЙ ДОМ!

Сборка: **Программирование:**

Вы когда-нибудь строили дома из обычных деталей LEGO? Теперь, когда вы знаете, как использовать моторы и датчики и как создавать рабочие программы, как насчет того, чтобы попытаться построить роботизированный дом с помощью EV3?

СОВЕТ **Используйте мотор для автоматического открывания двери, когда кто-то нажимает на дверной звонок (датчик касания), и установите сигнал о вторжении, который звучит, когда инфракрасный датчик обнаруживает кого-то. Используйте другой мотор, чтобы закрывать и открывать ставни, основываясь на уровне освещенности, измеренной с помощью датчика цвета.**

ЧАСТЬ III

Способы конструирования роботов

10

Конструирование с балками, осями, фиксаторами и моторами

Вы уже узнали довольно много о программировании роботов, но и учиться собирать роботов так же важно. Опыт сборки приходит с практикой, и в этой части книги вы получите основную информацию о конструировании роботов из деталей вашего набора MINDSTORMS EV3. В этой главе вы узнаете, как создавать прочные конструкции с помощью балок, рам, штифтов, фиксаторов и осей, которые вы можете увидеть на рис. 10.1. Вы поймете, как размерная сетка LEGO поможет вам собирать собственные конструкции из балок и фиксаторов. В главе 11 вы узнаете, как работают зубчатые колеса (шестеренки).

Каждая конструкция (но не все сразу), приведенная в этой главе, может быть построена с использованием деталей из вашего набора MINDSTORMS EV3. Изучите примеры, чтобы получить представление о том, насколько прочна каждая конструкция и какие из них будут полезны для ваших роботов.

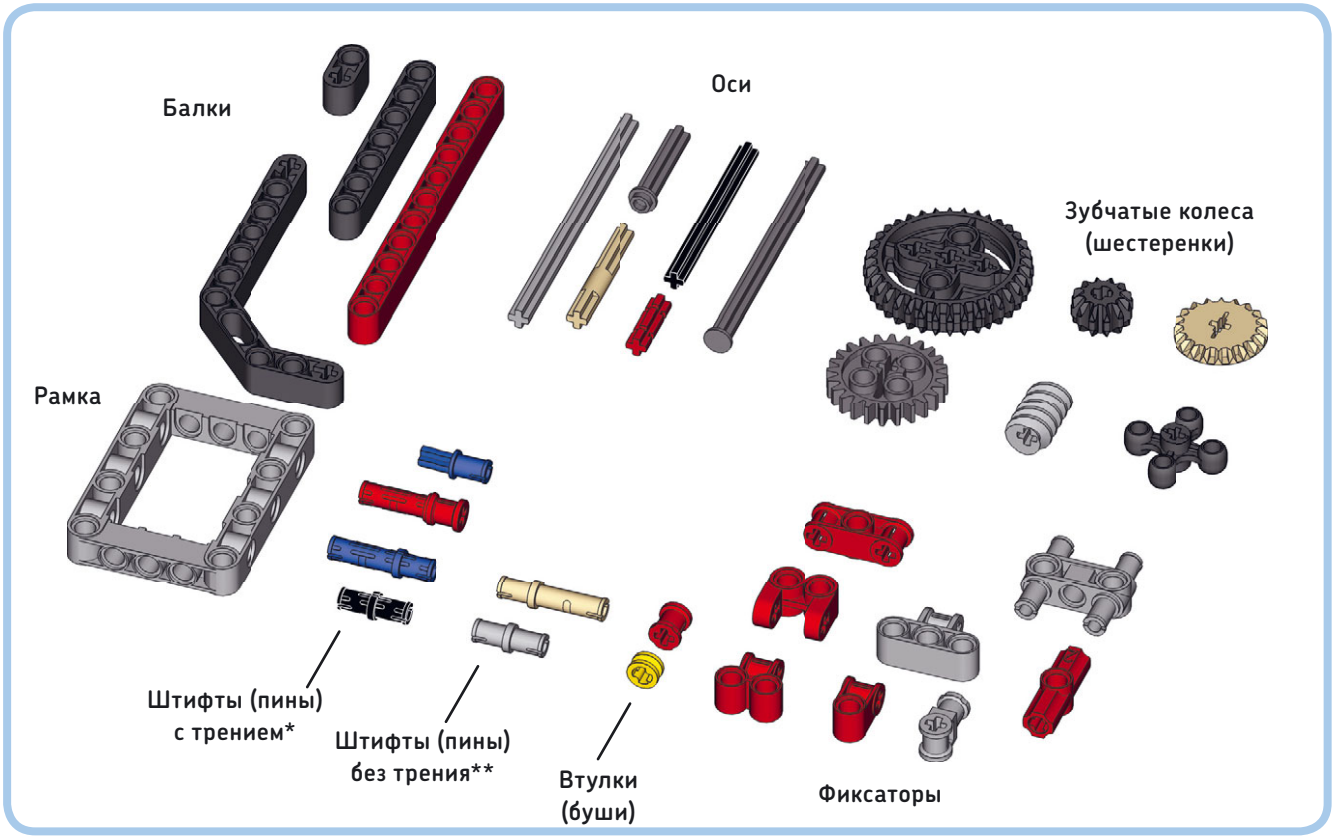

Рис. 10.1. Набор MINDSTORMS EV3 содержит множество различных балок, рам, осей, зубчатых колес, фиксаторов и штифтов. Полный список деталей вы найдете на внутренней стороне обложки

* Также называются фрикционными.

** Также называются бесфрикционными.

ПРИМЕЧАНИЕ **Если хотите, вы можете разобрать робота EXPLOR3R; для изучения остальной части книги он вам не понадобится.**

Использование балок и рамок

Вы уже многое знаете о применении моторов, датчиков и модуля EV3. *Балки* используются для создания конструкций, которые скрепляют все эти детали вместе. Длина балок и других деталей измеряется в *единицах* LEGO, иногда называемых *модулями*, как показано на рис. 10.2. Длина самой короткой прямой балки в наборе MINDSTORMS EV3 составляет две LEGO единицы, или 2M; самая длинная балка — 15M.

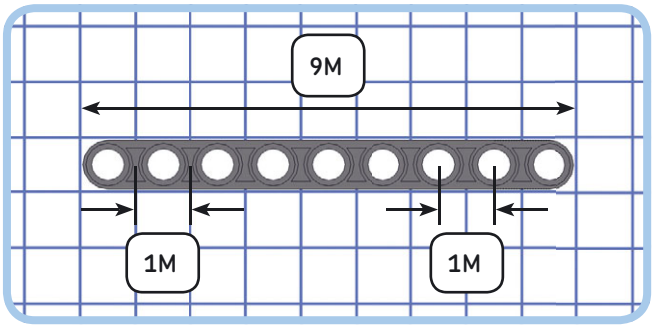

Рис. 10.2. Длина балок и других деталей измеряется в единицах LEGO. Расстояние между центральными точками двух соседних отверстий составляет одну LEGO единицу, или 1M. Следовательно, расстояние между центральными точками крайнего левого отверстия и крайнего правого отверстия этой балки длиной 9M составляет 8M

Удлинение балок

Балки можно увеличить в длину и ширину путем объединения нескольких балок с помощью штифтов с трением (с жестким креплением). Для неподвижного соединения вам потребуется не менее двух штифтов, и по крайней мере три отверстия должны перекрывать друг друга, как показано на рис. 10.3.

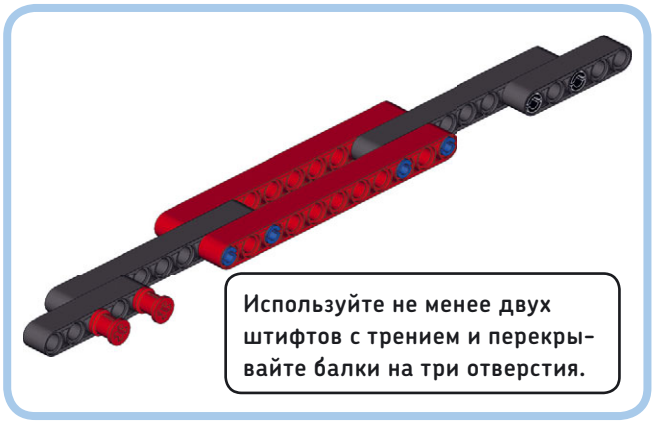

Рис. 10.3. Вы можете нарастить балки с помощью штифтов с трением (красные, синие и черные штифты в наборе)

Использование рамок

Набор MINDSTORMS EV3 содержит два типа рамок, они изображены на рис. 10.4. Рамки позволяют легко создавать большие конструкции со множеством отверстий для крепления других деталей, например балок и моторов. Рамки могут быть использованы для подсоединения балок под прямым углом.

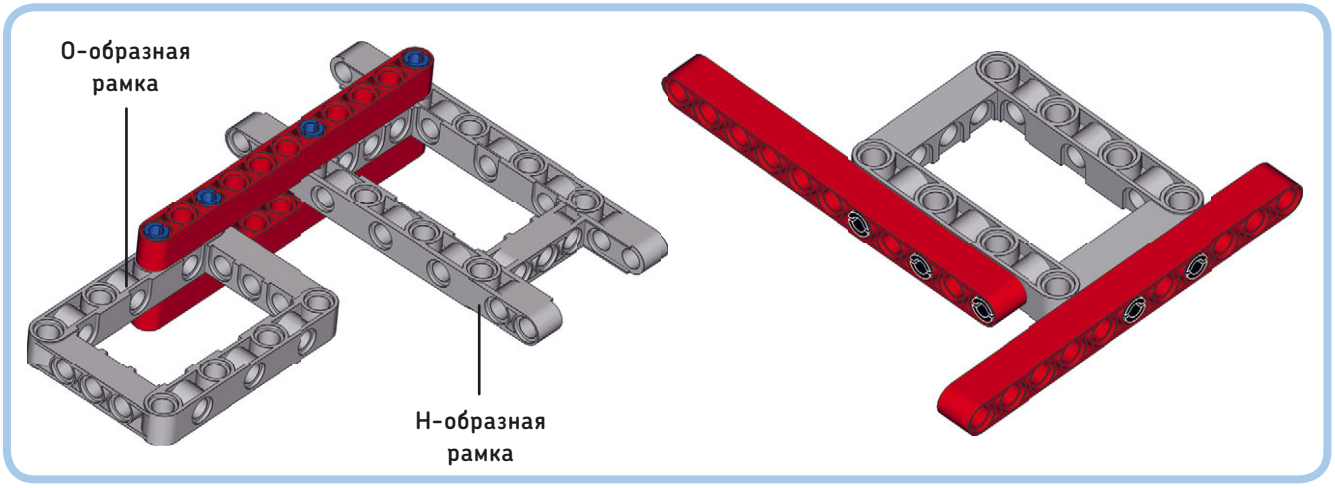

Рис. 10.4. Рамки используются для создания больших и прочных конструкций с большим количеством отверстий для прикрепления других деталей (слева) и для соединения балок под прямым углом (справа)

Использование балок для укрепления конструкций

Балки могут быть использованы не только для создания новых конструкций, но и для укрепления или усиления уже существующих. Например, рассмотрим верхнюю конструкцию на рис. 10.5, состоящую из двух рамок. Ее легко разрушить, потянув рамки в разные стороны, потому что они соединены только в двух местах.

Нижняя конструкция усилена двумя балками 3М. Такое соединение очень трудно разрушить и разъединить две рамки. Еще больше усилить такую конструкцию возможно, заменив балки 3М на балки 11М, которые охватывают всю ширину рамки.

Использование угловых балок

Набор MINDSTORMS EV3 содержит множество *угловых балок* различных типов и с разными углами. В том числе четыре типа угловых балок с углом 90 градусов (или с *прямым углом*), как показано на рис. 10.6. Балки такого типа используются для соединения прямых балок под прямым углом.

Кроме балок, согнутых под углом 90 градусов, ваш набор содержит два типа балок с углом 53,13 градуса,

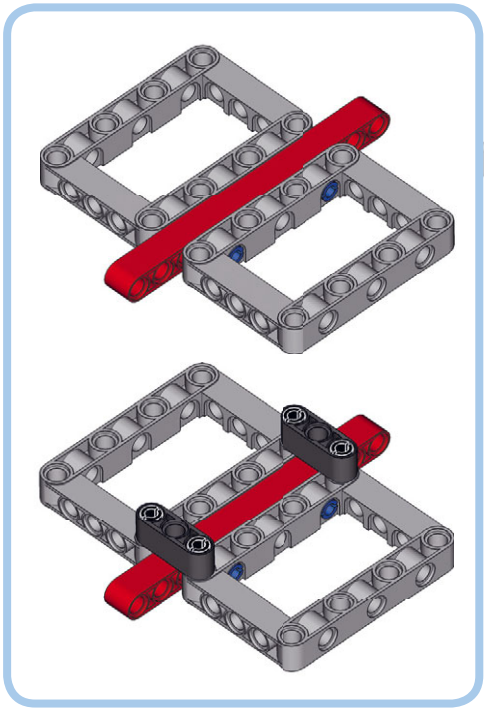

Рис. 10.5. Балки можно использовать для укрепления конструкций. Сравните прочность обеих конструкций, пытаясь разъединить две рамки. Вы поймете, что конструкция внизу гораздо прочнее

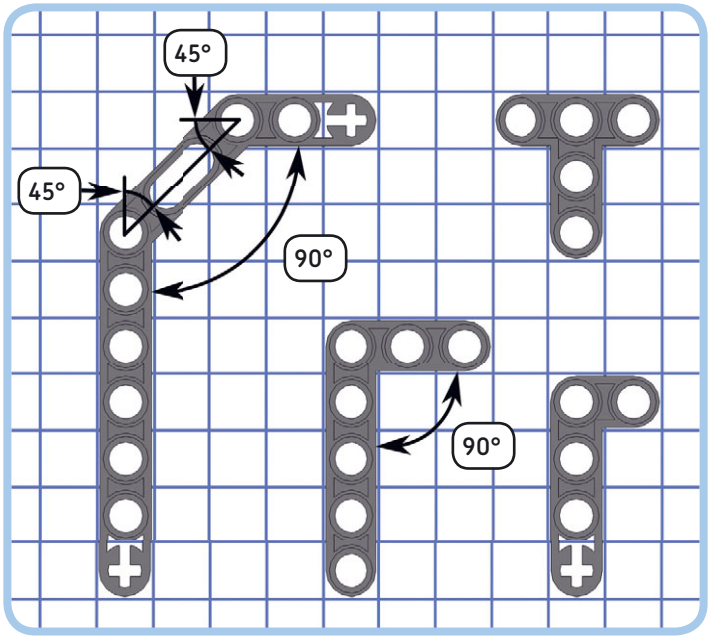

Рис. 10.6. Четыре типа балок с прямым углом (90 градусов)

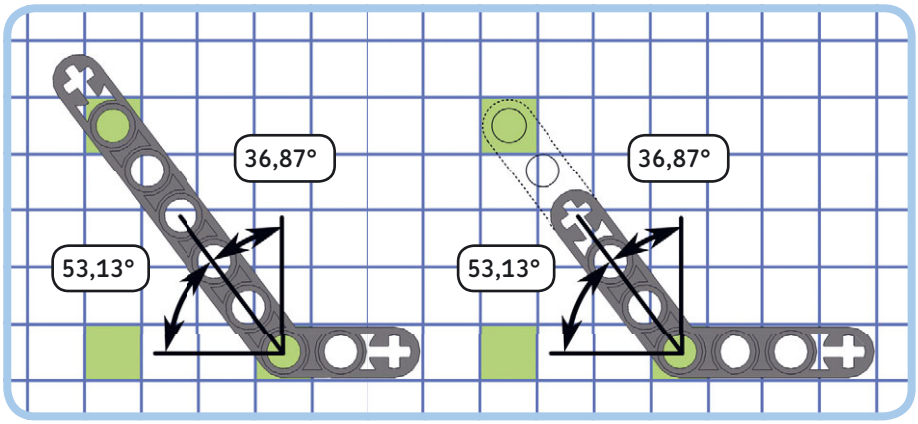

Рис. 10.7. Две балки с углом 53,13 градуса. Поскольку углы обеих балок одинаковы, вы можете нарастить более короткую балку (справа), присоединив прямую балку, чтобы ее можно было использовать для достижения таких же целей, как и длинную (слева)

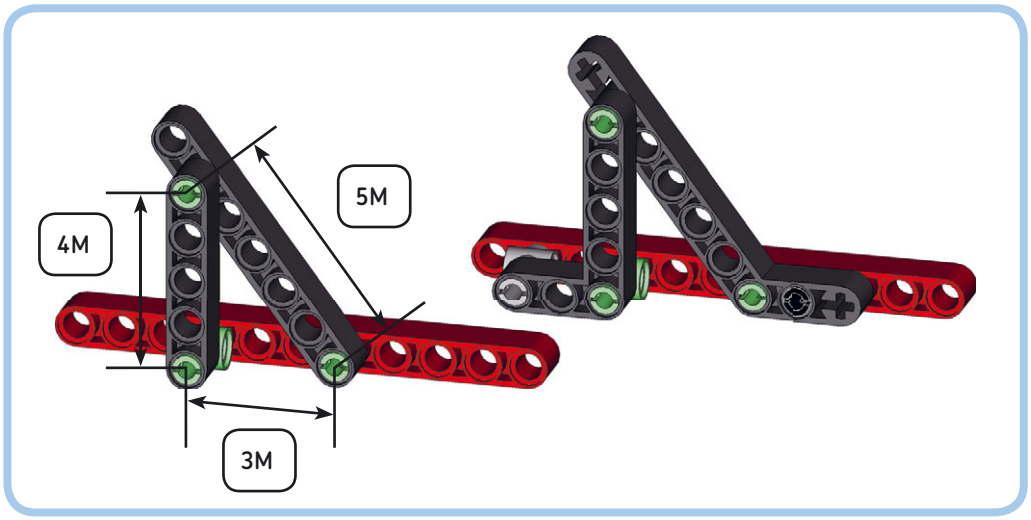

Рис. 10.8. Создание двух одинаковых прямоугольных треугольников с использованием прямых балок (слева) и угловых балок (справа). Штифты на самом деле не зеленого цвета, они окрашены для книги, чтобы показать точки соединения, отмеченные зеленым цветом на рис. 10.7. Обратите внимание, что длина стороны треугольника измеряется между центральными точками отверстий на каждом углу, а не путем подсчета количества отверстий на балке

которые изображены на рис. 10.7. Такой угол может показаться странным, но на самом деле его удобно использовать, потому что он может образовывать угол какого-нибудь общего треугольника. В частности, вы можете использовать этот угол, чтобы создать треугольник Пифагора (прямоугольный), стороны которого составляют 3M, 4M и 5M, как показано на рис. 10.8.

ПРАКТИКУМ № 54: БОЛЬШИЕ ТРЕУГОЛЬНИКИ!

Сложность: **Время:**

Существует еще один полезный прямоугольный треугольник, который можно собрать из деталей набора MINDSTORMS EV3. Его размер в два раза превышает размер треугольника, показанного на рис. 10.8. Можете ли вы собрать такой треугольник? А треугольник в три раза больше?

Размерная сетка LEGO

Слева на рис. 10.9 изображена сетка с квадратами размером 1M. Эта размерная сетка LEGO поможет вам в сборке прочных роботов. Если присоединять новые детали к балкам так, чтобы их отверстия совпадали с этой сеткой, позже будет легче добавлять новые детали (а).

Если вы отклонитесь от сетки, размещая детали под углом, соединение большого количества деталей становится затруднительным, так как детали LEGO имеют ограниченное количество фиксированных размеров (б). Такой тип соединения не стоит использовать для сборки несущей конструкции вашего робота, хотя его вполне можно применить для декоративных деталей, таких как хвост робота-животного.

Сборка без использования сетки чревата тем, что балки будут растягиваться или сгибаться, что может привести к повреждению деталей LEGO (в). Поэтому вы всегда должны избегать такого варианта сборки. Не используйте соединений, которые вы не можете сделать без небольшого сгибания или растягивания деталей. Если вы не уверены, нужно ли будет сгибать детали в угловой конструкции, лучше придерживаться сетки, используя для соединения прямые углы.

Вы можете положить на сетку балки с углом 53,13 градуса, чтобы определить соединительные отверстия, отмеченные зеленым на рис. 10.7. Таким образом, вы можете использовать угловые балки для создания прямоугольных конструкций, как показано на рис. 10.10.

ПРИМЕЧАНИЕ Свою копию сетки LEGO вы можете распечатать, открыв файл *Размерная_сетка.pdf*, и использовать распечатку для создания ваших собственных конструкций. Не забудьте выбрать *фактический размер*, или 100%, в настройках режима печати. Балка 15M на печатной схеме должна быть такого же размера, как реальная балка 15M. Если размер получился не такой, измените настройки масштаба и распечатайте сетку еще раз.

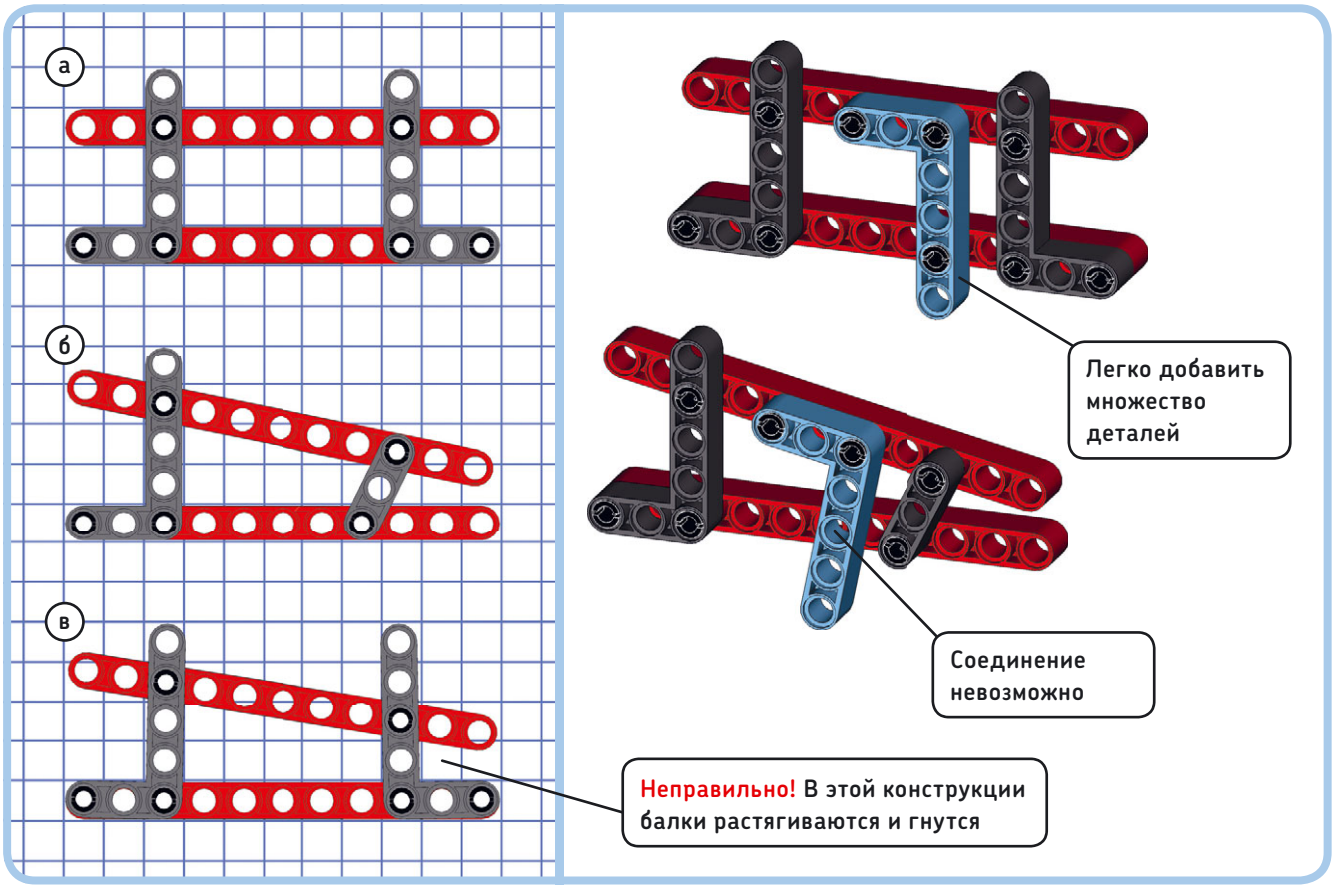

Рис. 10.9. Рекомендуется выстраивать конструкции на сетке (а). При необходимости можно собирать конструкции, отклоняясь от сетки (б) до тех пор, пока вы не найдете подходящее соединение балок (в)

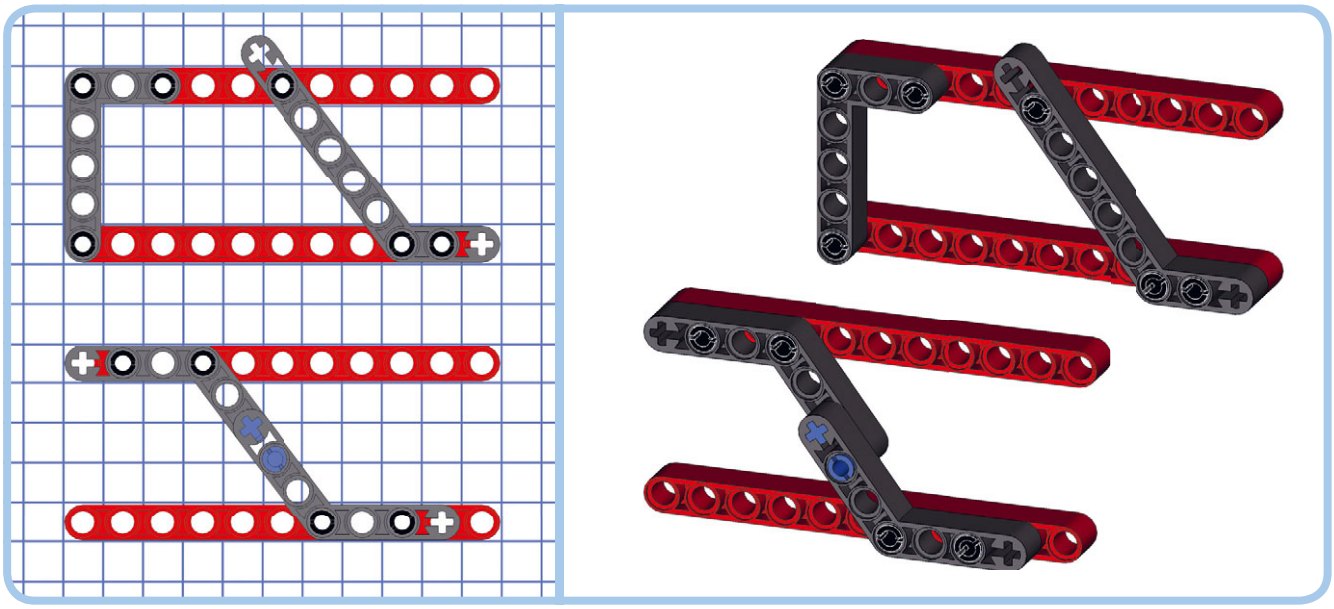

Рис. 10.10. Два способа использовать угловые балки 53,13 градусов, с раскладкой на сетке

ПРАКТИКУМ № 55: УГЛОВЫЕ КОМБИНАЦИИ!

Сложность: **Время:**

Можете ли вы найти комбинацию из двух угловых балок (53,13 градуса) для соединения двух параллельных балок 11М, показанных на рис. 10.11?

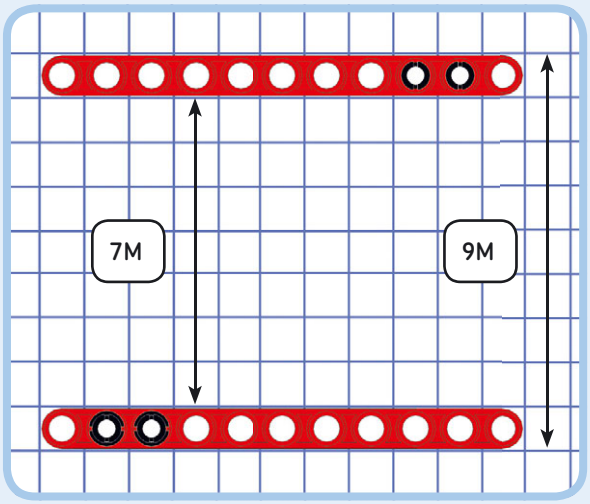

Рис. 10.11. Две прямые балки из практикума № 55

Использование осей и крестовых отверстий

Ось представляет собой стержень, на который можно установить вращающиеся детали, например колёса или шестеренки (зубчатые колеса). В круглых отверстиях оси свободно вращаются, но вы можете использовать их для неподвижных соединений, установив в крестовые отверстия, как показано на рис. 10.12. Самая короткая ось в наборе MINDSTORMS EV3 имеет длину 2М; самая длинная — 9М.

Чтобы ось не выпадала из круглого отверстия, добавьте втулку, чтобы закрепить ось, или используйте оси с ограничителем, как показано на рис. 10.13.

Осевой штифт с трением, изображенный на рис. 10.14, представляет собой полезный фиксатор. Один конец штифта с трением можно установить в круглое отверстие, где он будет вращаться с некоторым сопротивлением. Другой

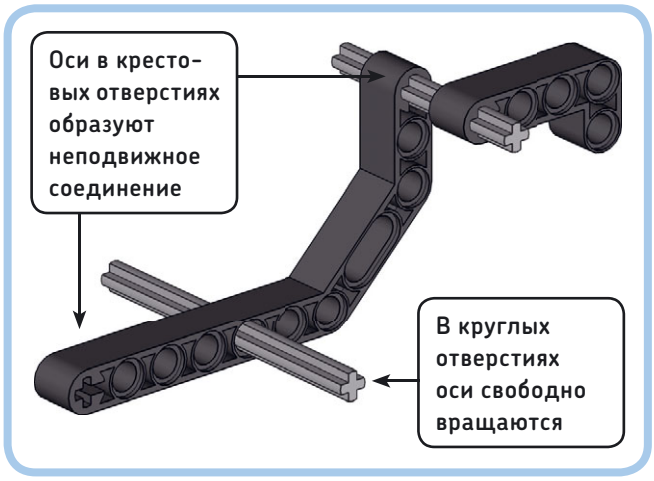

Рис. 10.12. Оси свободно вращаются в круглых отверстиях, а в крестовых образуют жесткое соединение

Рис. 10.13. Вы можете закрепить ось в отверстии балки с помощью втулки. Если ось имеет ограничитель на одном конце, вам понадобится только одна втулка

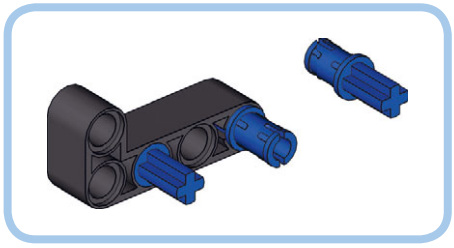

Рис. 10.14. Синие осевые штифты с трением соединяют круглое отверстие с крестообразным

конец представляет собой ось, которая может быть установлена в крестовое отверстие, где она будет оставаться неподвижной. Некоторые наборы LEGO содержат подобные желтовато-коричневые или серые осевые штифты без трения; такой штифт плавно вращается в круглом отверстии.

Использование фиксаторов

Для установки балок, осей, моторов и датчиков под разными углами вы можете использовать *фиксаторы*. Каждый тип фиксатора в наборе MINDSTORMS EV3 может быть использован разными способами. В этом разделе приведено несколько полезных примеров, с которых можно начать.

Удлинение осей

Некоторые фиксаторы можно использовать для соединения двух осей, как показано на рис. 10.15. Это позволяет совместить оси под углом или сделать их длиннее.

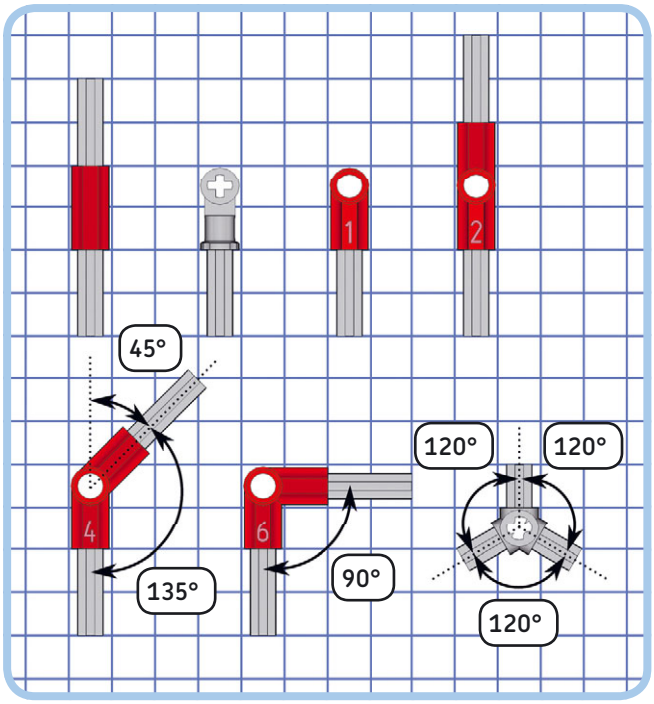

Рис. 10.15. Удлинение осей с помощью фиксаторов

Соединение параллельных балок

Чтобы соединить две параллельные балки, вы можете использовать как рамки или балки, так и комбинацию фиксаторов, как показано на рис. 10.16 и 10.17. Такое соединение будет уместно, когда ограничено пространство или если использование балок или рамок нецелесообразно. Используйте размерную сетку в качестве эталона для своих конструкций. Например, если вам необходимо соединить две параллельные балки длиной 3М, вы можете использовать вариант *f*, показанный на рис. 10.16.

В левой части рис. 10.16–10.18 показано, как комбинировать разные фиксаторы, а справа изображены примеры, иллюстрирующие эти комбинации.

ПРИМЕЧАНИЕ Эти примеры иллюстрируют, как соединить две балки с помощью фиксаторов. Те же принципы вы можете применить к другим деталям с отверстиями. Например, вы можете использовать эти комбинации фиксаторов, чтобы прикрепить датчик к отверстиям мотора или модуля EV3.

Соединение балок под прямым углом

Многие фиксаторы имеют круглые или крестообразные отверстия, расположенные перпендикулярно друг к другу. Это позволяет подсоединить балки под прямым углом или расположить их параллельно, но так, чтобы их отверстия находились перпендикулярно друг к другу, как показано на рис. 10.18.

Укрепление параллельных балок

Вы уже видели ранее, что с помощью балок можно укрепить конструкцию (см. рис. 10.5). Но в зависимости от ориентации отверстий балки может возникнуть необходимость добавить фиксаторы, прежде чем вы сможете подсоединить балки для укрепления, как показано на рис. 10.19.

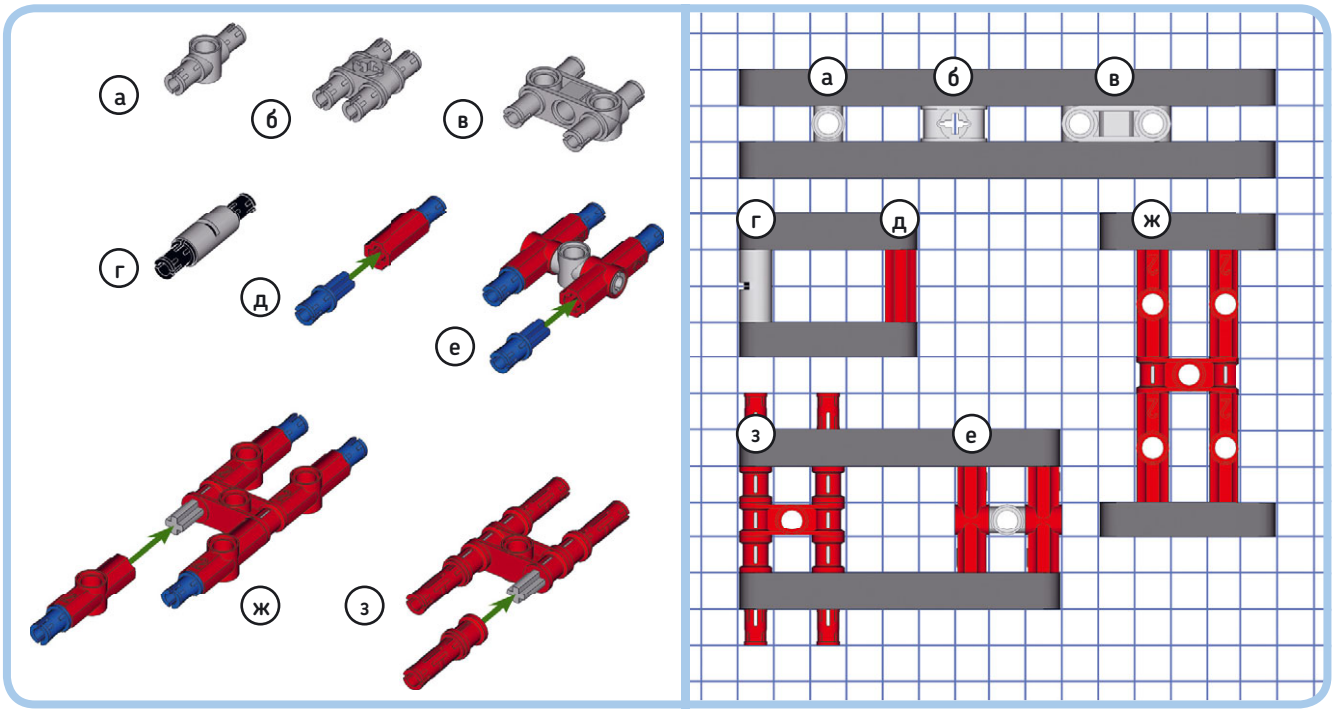

Рис. 10.16. Соединение параллельных балок в случае, если их отверстия обращены друг к другу

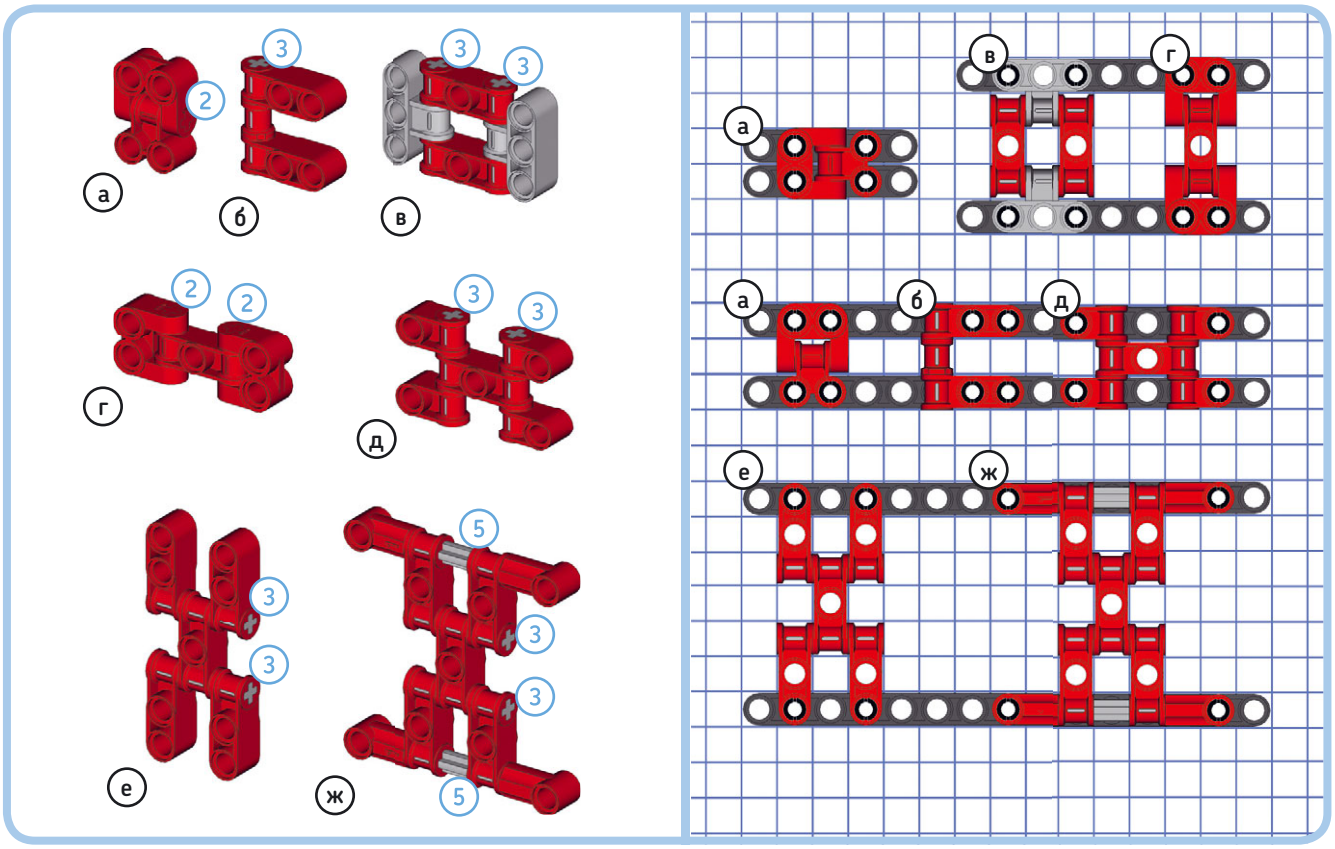

Рис. 10.17. Соединение параллельных балок, если их плоские стороны обращены друг к другу. Числа в кружочках обозначают длину осей, используемых в конструкциях

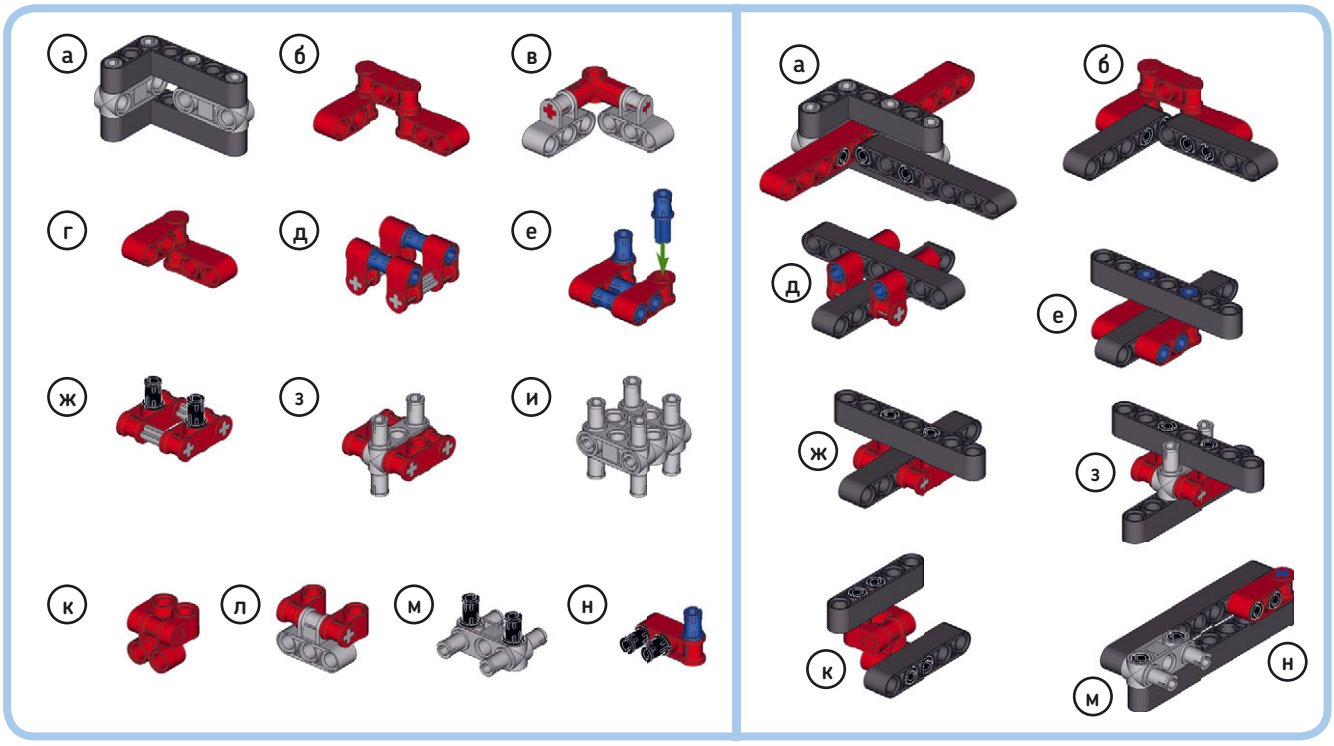

Рис. 10.18. Использование фиксаторов для подсоединения балок под прямым углом. Серые оси на рисунке имеют длину 3М

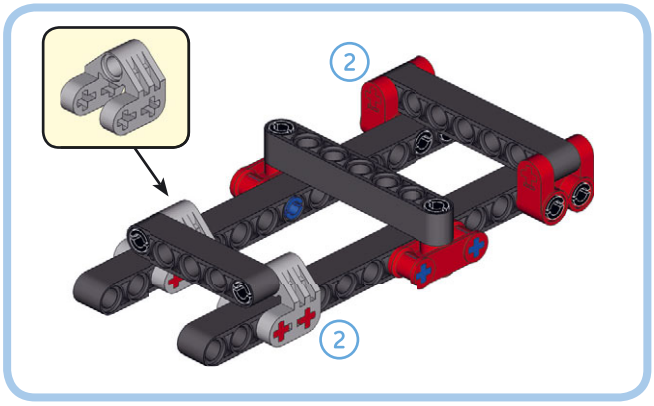

Рис. 10.19. Вы можете применить фиксаторы, чтобы создать точки крепления для балок, которые усилят конструкцию. Обратите внимание, что эти конструкции не являются жесткими, когда используются по отдельности. Этот способ укрепления конструкций нужно использовать так же, как изображенные на рис. 10.16

ПРАКТИКУМ № 56: КОНСТРУКТИВНЫЕ ФИКСАТОРЫ!

Сложность: **Время:**

Можете ли вы скомбинировать фиксаторы так, чтобы создать прочное крепление балок, показанных на рис. 10.20? Дополните примерами, взятыми с рис. 10.16–10.19, или создайте собственные комбинации.

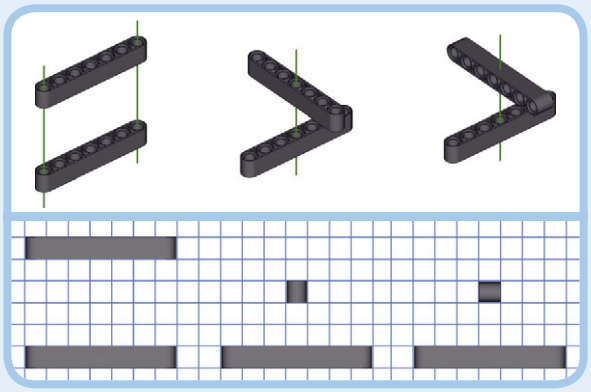

Рис. 10.20. Зеленые линии показывают, как две балки расположены по отношению друг к другу

Детали с половинчатым размером

Некоторые фиксаторы заканчиваются на половине клетки, как показано на рис. 10.21. При правильном использовании они предоставляют больше возможностей без ущерба для конструкции. Например, вы можете создавать конструкции, имеющие размер 7,5М, а не 7М или 8М.

Обратите внимание, что такие конструкции укреплять балками не так просто, поскольку расстояние между двумя отверстиями балки всегда должно быть целым числом. То есть применительно к данной конструкции невозможно использовать другую балку, чтобы соединить верхнюю балку с той, что в середине.

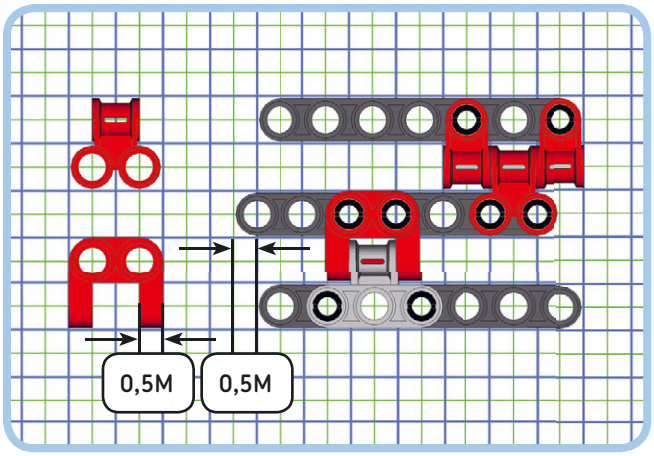

Рис. 10.21. Некоторые фиксаторы приводят к смещению деталей на 0,5М относительно квадрата размерной сетки. Балка в середине смещена на 0,5М относительно двух других балок

ПРАКТИКУМ № 57: ПОЛОВИНЧАТЫЕ БАЛКИ!

Сложность: **Время:**

Можете ли вы соединить две балки, чтобы сделать балку длиной 18,5М, с помощью фиксаторов?

СОВЕТ **Используйте фиксаторы, которые показаны на рис. 10.21.**

Использование тонких деталей

Большинство деталей в наборе MINDSTORMS EV3 имеют точную ширину, равную 1М, но есть и тонкие детали шириной 0,5М, как показано на рис. 10.22. Тонкие детали могут быть использованы, когда для установки обычных не хватает места. В наборе MINDSTORMS EV3 мало тонких деталей, поэтому возьмите их из набора LEGO Technic.

Одной из деталей, обозначенных здесь, является кулачок мотора. Он пригодится для создания вращающегося механизма, который нажимает кнопку датчика касания один раз за один оборот. Об этом вы подробнее узнаете в главе 13.

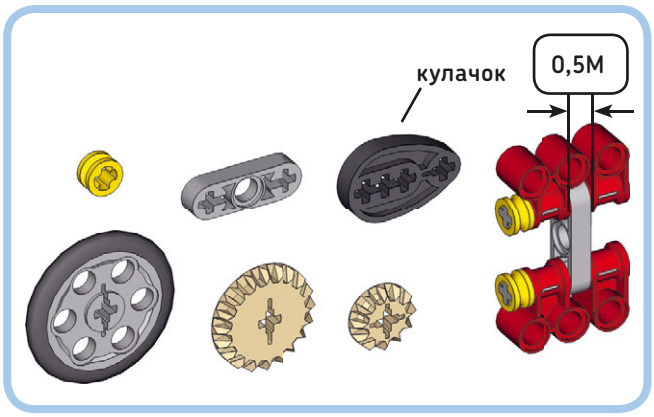

Рис. 10.22. Тонкие детали в наборе MINDSTORMS EV3

Создание гибких конструкций

Для создания шарниров и гибких механизмов вместо жестких конструкций нужно использовать штифты без трения (серые и желтовато-коричневые штифты в наборе MINDSTORMS EV3). Штифты без трения в механизме на рис. 10.23 позволяют легко повернуть зубчатое колесо.

Набор MINDSTORMS EV3 содержит тяги двух типов (6М и 9М), которые обычно используются для создания рулевых механизмов в автомобилях LEGO Technic. Эти тяги можно применить в некоторых конструкциях в качестве замены для балок. Хотя тяги и создают менее прочное соединение, чем балки, они могут быть использованы для соединения деталей, которые находятся в разных плоскостях. То есть, если вы расширите движущуюся балку предыдущего механизма, вы не сможете использовать балку для соединения с зубчатым колесом. Вместо нее можно использовать тягу, как показано на рис. 10.24.

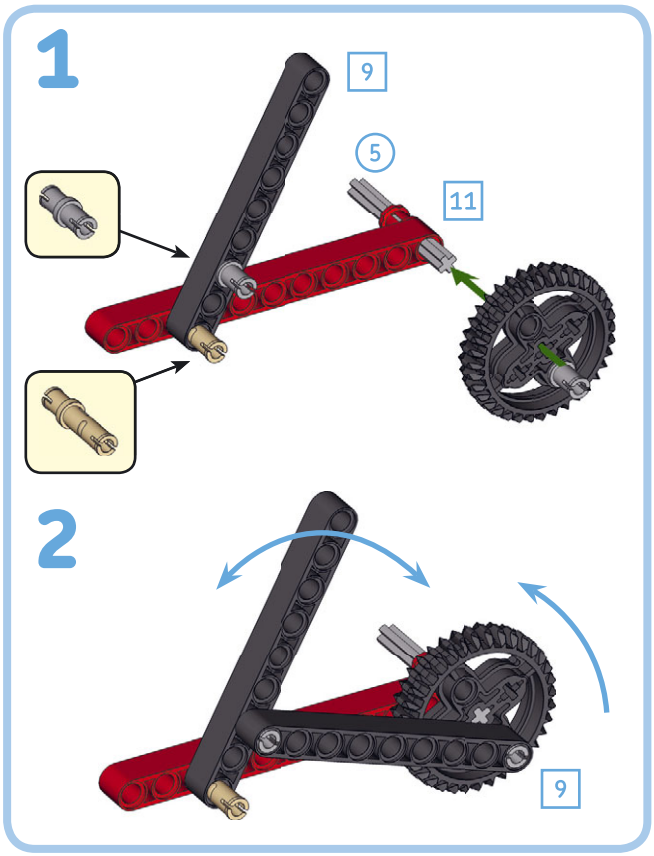

Рис. 10.23. В этой динамической конструкции используются штифты без трения, поэтому ее легко превратить в механизм, который заставляет балку двигаться вперед и назад. Для сравнения используйте штифты с трением, и вы увидите, что повернуть зубчатое колесо стало намного сложнее

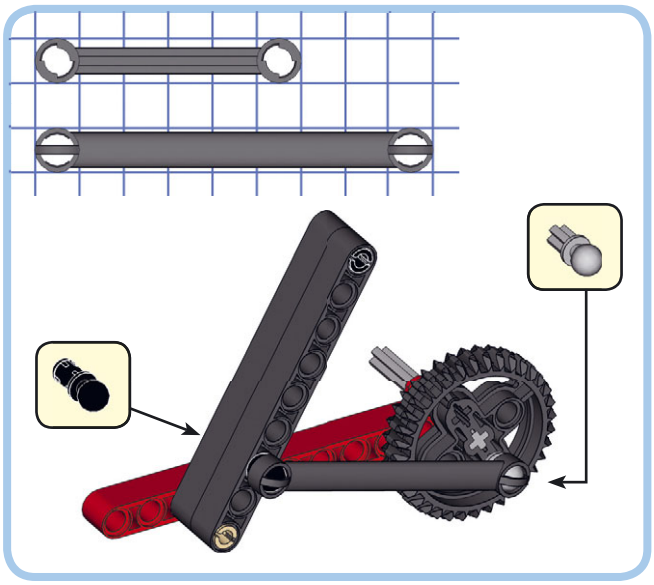

Рис. 10.24. В этой модифицированной версии динамической конструкции вместо балки используется тяга. Вы присоединяете тягу через круглое или крестообразное отверстие, используя штифт с трением и с шаром, как показано на рисунке

Конструкции с моторами и датчиками

Теперь мы рассмотрим, как, учитывая размер и форму моторов, использовать их в качестве центрального компонента вашего робота. Моторы довольно велики, и у них есть множество точек крепления для штифтов и осей. Поэтому чаще всего при создании такого механизма, как роботизированная клешня или гусеничный движитель, сборку лучше начинать как раз с мотора.

Такой способ позволяет тестировать каждый механизм, или *модуль*, по отдельности. После того как вы убедились, что все модули прекрасно работают по отдельности, вы можете собрать из них целого робота.

Конструкции с большим мотором

Геометрия большого мотора (рис. 10.25) позволяет легко соединять два мотора с использованием рамки и штифтов с трением. Это дает преимущество в создании транспортного робота, как показано на рис. 10.26. Все, что вам нужно сделать, это добавить колеса или гусеницы и модуль EV3.

Подключение колес и гусениц

Большой мотор достаточно мощный и быстрый, чтобы вращать колеса непосредственно, как показано на рис. 10.27. Вы также можете напрямую управлять гусеницами танков, присоединив их с помощью двух балок 13M. Рис. 10.28 демонстрирует базовую схему сборки. Другой пример можно увидеть в главе 18, там, где собирается SNATCH3R. Как подключить зубчатые колеса к большому мотору, вы узнаете в следующей главе.

Подключение балок к валу мотора

Подсоединить колеса и шестеренки к красному валу большого мотора вы можете с помощью крестового отверстия. Кроме того, чтобы балки и другие детали вращались, их можно вставить в круглые отверстия на валу. Например, чтобы создать возвратно-поступательный механизм, вы можете сделать так, чтобы мотор непрерывно вращал балку 3M, как показано на рис. 10.29.

Конструкции со средним мотором

Средний мотор (рис. 10.30) меньше по размеру, чем большой, что позволяет использовать его в небольших конструкциях, таких как рулевой механизм гоночного автомобиля.

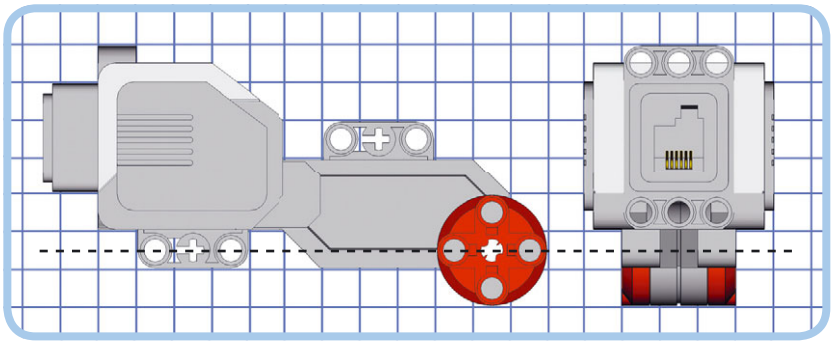

Рис. 10.25. Внешний вид большого мотора

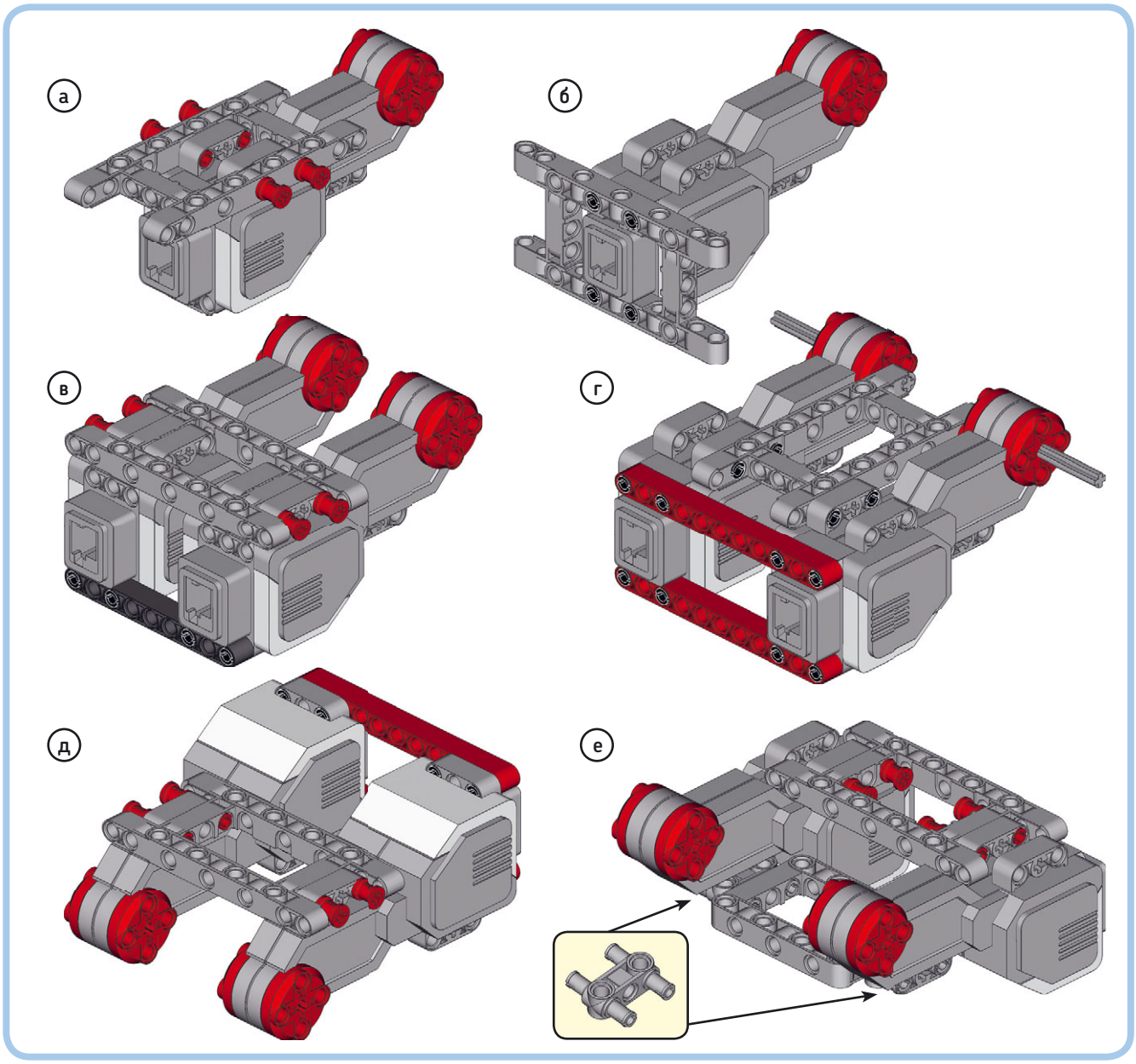

Рис. 10.26. Упростить крепление мотора к роботу можно, установив на него рамки. Например, вы можете создать основу для транспортного робота, объединив два больших мотора с помощью рамок, балок и штифтов с трением различными способами

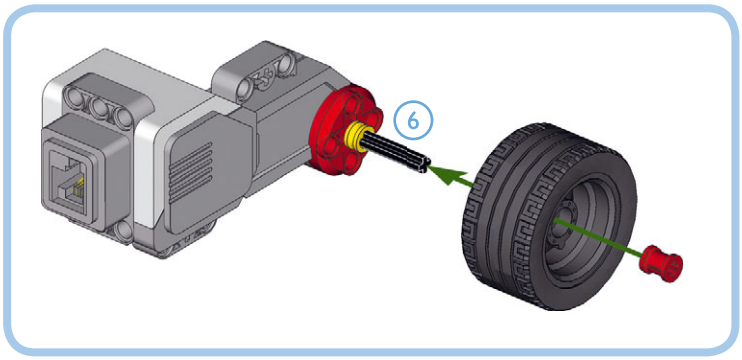

Рис. 10.27. Колеса можно подключить непосредственно к большому мотору с помощью оси 6М. Половинчатая втулка создает небольшое пространство между мотором и колесом, а обычная втулка нужна, чтобы колеса не соскальзывали с оси

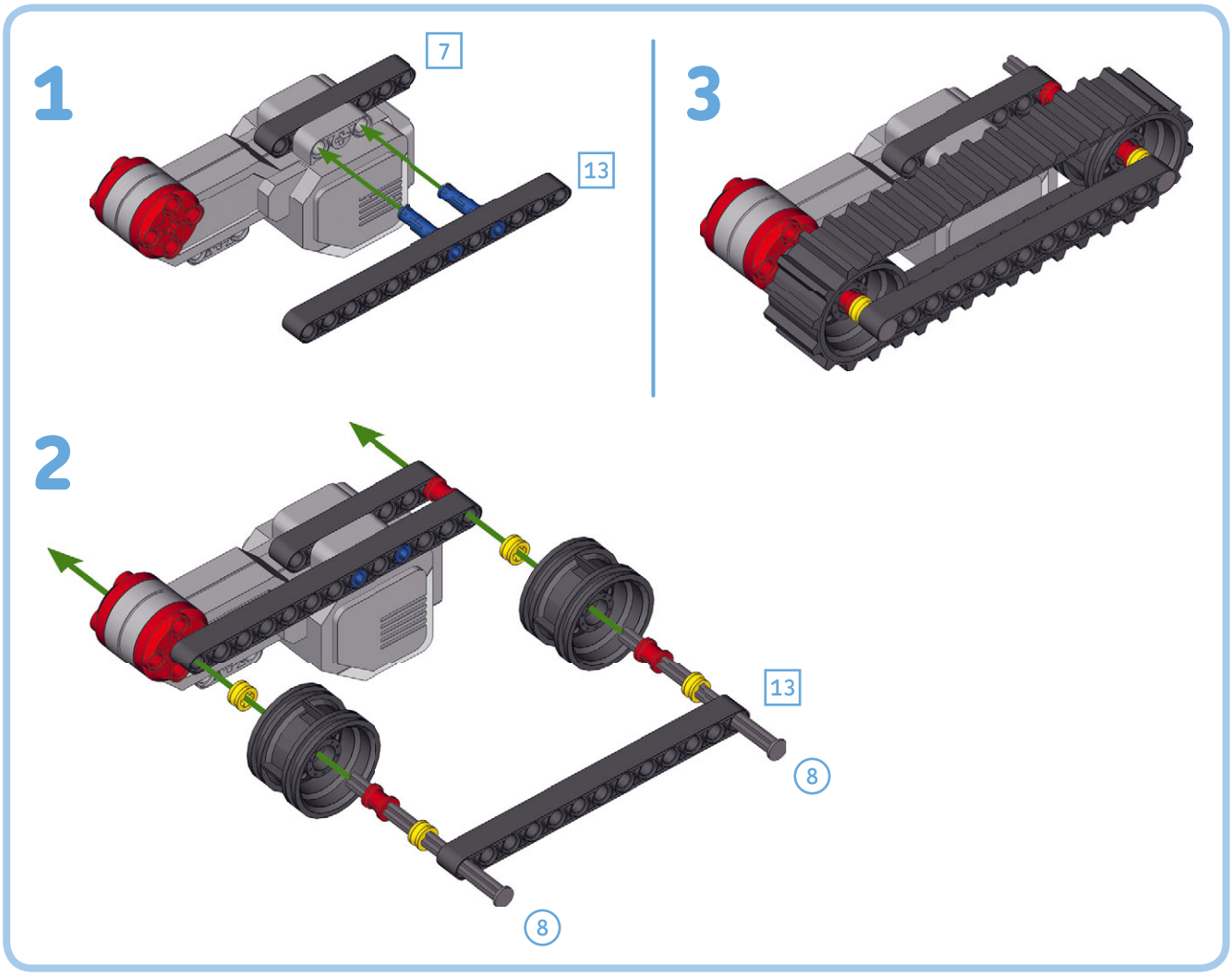

Рис. 10.28. Вы можете подключить диски гусеничного движителя к большому мотору с помощью двух балок длиной 13М и двух осей 8М с ограничителями

На передней стороне мотора есть круглые соединительные отверстия. А добавив рамку, можно создать больше точек подключения в задней части, как показано на рис. 10.31. Ну а если добавить рамку, как показано на рис. 10.32, вы сможете установить его между двумя большими моторами транспортного робота, например чтобы привести в движение механизм вилочного погрузчика.

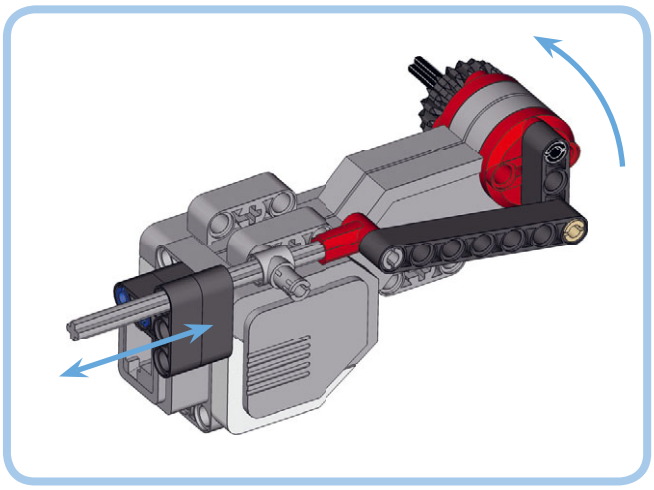

Рис. 10.29. Вы можете подключить шестеренки и колеса к валу мотора с помощью оси или фиксирующих балок, используя круглые отверстия. Этот механизм двигает серую ось 9М вперед и назад за один оборот вала мотора

Конструкции с датчиками

Каждый датчик в наборе MINDSTORMS EV3 имеет точки крепления для одной оси и двух штифтов, как показано на рис. 10.33. Кроме того, инфракрасный датчик имеет два круглых отверстия в задней части. Чтобы создать жесткое соединение, вам нужно использовать либо два штифта и балку, либо ось и балку с крестообразным отверстием.

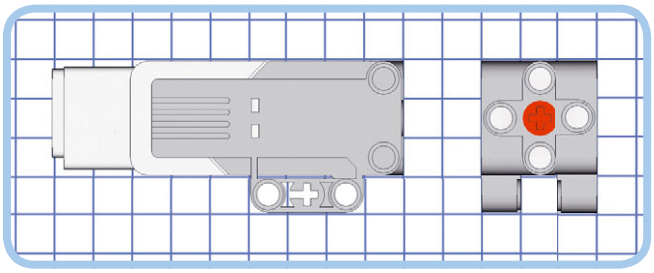

Рис. 10.30. Внешний вид среднего мотора

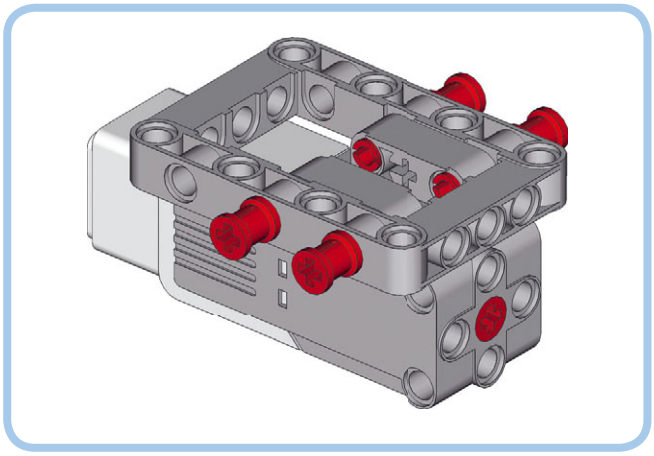

Рис. 10.31. Добавление точек подключения к среднему мотору с помощью рамки

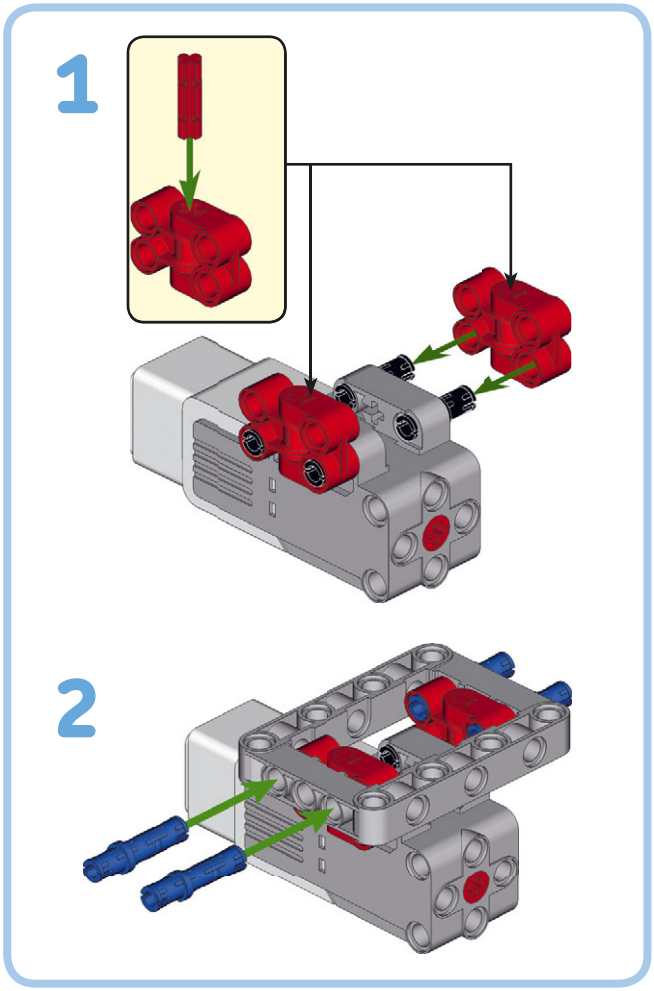

Рис. 10.32. Вы можете прикрепить средний мотор к раме так, что его легко будет установить между двумя большими моторами транспортного робота. Например, вы можете удалить рамку в примере е на рис. 10.26 и установить вместо нее эту конструкцию

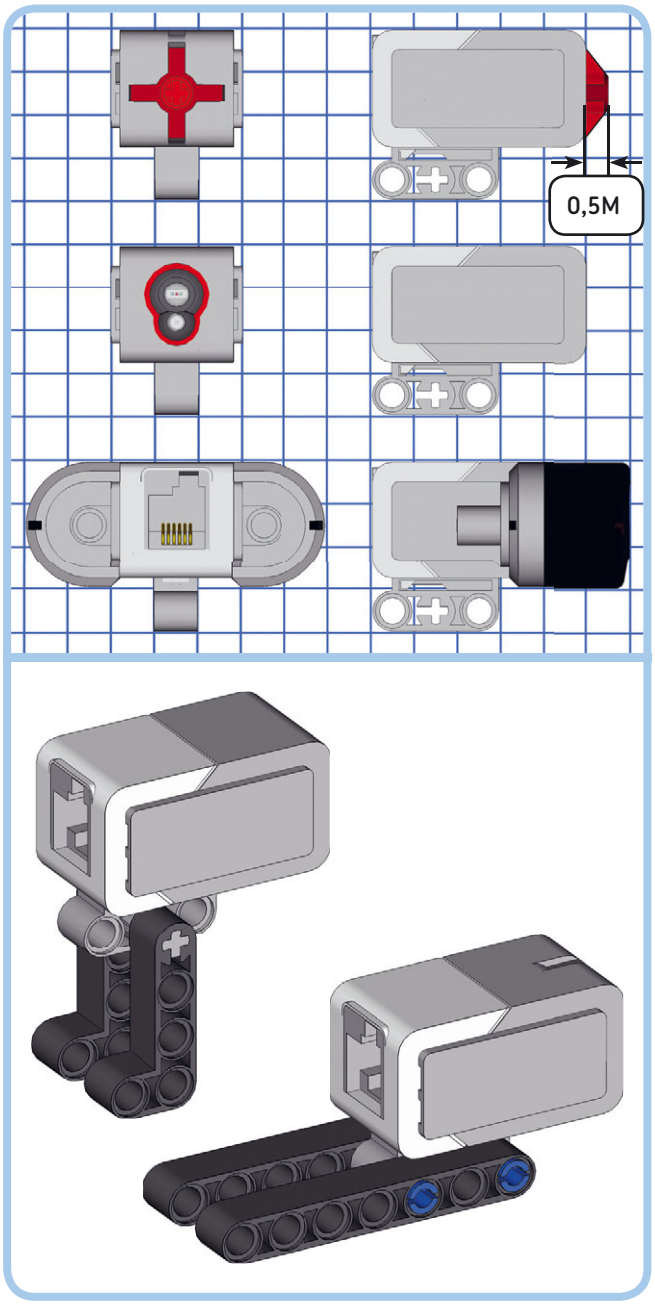

Рис. 10.33. Внешний вид датчиков в наборе MINDSTORMS EV3 (вверху) и присоединение их к роботу (внизу)

Прочие детали

В дополнение к деталям, обсуждаемым в этой части книги, набор содержит другие разнообразные элементы, такие как мечи и маски монстров, которые можно использовать для украшения ваших роботов. И наконец, ваш набор включает в себя детали магазина и механизма для стрельбы шариками. Вы найдете инструкции по строительству такой пушки, изучив раздел «Сборка и программирование» робота EV3RSTORM в программе EV3 (центральный робот на рис. 3.2 в главе 3).

Дальнейшее изучение

В этой главе вы научились использовать балки, рамки, штифты, оси, фиксаторы и моторы, с помощью которых создаются компоненты ваших роботов. Вы узнали, как размерная сетка LEGO может помочь в сборке прочных конструкций. Рецепта для создания идеального робота не существует. А лучшим способом получить опыт в разработке собственных роботов была и остается практика. Вы можете конструировать проекты, представленные в этой книге, а затем изменять их, создавая своих роботов, выполняя по возможности все задания «Сделай сам».

В следующей главе мы рассмотрим, как работают зубчатые колеса и как вы можете использовать их с моторами EV3.

СДЕЛАЙ САМ № 12: ПЕРЕСЯДЕМ НА ГУСЕНИЦЫ!

Сборка: **Программирование:**

Робот EXPLOR3R перемещается на двух основных колесах и опорном колесе сзади. Можете ли вы создать версию EXPLOR3R на гусеницах? Проверьте свое творение, управляя им с помощью инфракрасного маяка.

СОВЕТ **Используйте пример гусеничного движителя, показанный на рис. 10.28. Почему нужно убрать опорное колесо EXPLOR3R?**

СДЕЛАЙ САМ № 13: РОБОТ-УБОРЩИК!

Сборка: **Программирование:**

Можете ли вы создать робота, который ездит по столу, не падая с него? Сконструируйте робота, который будет сметать любые детали LEGO, оказавшиеся перед ним, с помощью среднего мотора, освобождая таким образом пространство во время движения.

Установите на переднюю часть робота инфракрасный датчик, закрепив его на расстоянии около 25 см от передних колес, и направьте его вниз, чтобы он «видел» поверхность стола. Как сделать так, чтобы робот определял, что он приближается к краю стола?

СОВЕТ **Если вы уже построили робота, описанного в практикуме «Сделай сам № 12» на с. 141, вы можете использовать его в качестве отправной точки для этого задания.**

СДЕЛАЙ САМ № 14: РОБОТ, ОТКРЫВАЮЩИЙ ШТОРЫ!

Сборка: **Программирование:**

Можете ли вы создать робота, который автоматически открывает шторы на восходе солнца и закрывает на закате? Используйте датчик цвета для измерения яркости окружающего света, а также добавьте возможность отмены автоматического поведения с помощью удаленного инфракрасного маяка.

СОВЕТ **Если вы используете только одно пороговое значение, робот может открывать и закрывать шторы по несколько раз, если уровень освещенности варьируется около порогового. Чтобы избежать этой проблемы, вы можете использовать два разных пороговых значения. Что робот должен делать, когда уровень освещенности находится между этими значениями?**