

Lego mindstorms EV3

Изучаем датчик цвета



Датчик цвета может работать в трех различных режимах:

- * в режиме "Цвет" датчик может определить цвет поднесенного к нему предмета;
- * в режиме "Яркость отраженного света" датчик направляет световой луч на близкорасположенный предмет и по отраженному пучку определяет яркость предмета;
- * в режиме "Яркость внешнего освещения" датчик может определить - насколько ярко освещено пространство вокруг.

Режим "Цвет"

* В режиме "Цвет" датчик цвета достаточно точно умеет определять семь базовых цветов предметов, находящихся от него на расстоянии примерно в 1 см. Это следующие цвета: "черный"=1, "синий"=2, "зеленый"=3, "желтый"=4, "красный"=5, "белый"=6 и "коричневый"=7. Если предмет удален от датчика или некорректно определяется цвет предмета - датчик информирует об этом состоянием "Без цвета"=0.

Оранжевая палитра, программный блок "Переключатель"

- * **Задача №1:** необходимо написать программу, называющую цвета предметов, подносимых к датчику цвета.
- * В решении Задачи №1 нам поможет программный блок "Переключатель" Оранжевой палитры. Этот блок в зависимости от настроек выбирает для выполнения программные блоки, расположенные в одном из своих контейнеров. Рассмотрим настройку этого блока в режиме работы с датчиком цвета.
- * Создадим новую программу "lesson-C", установим в программе блок "Переключатель", выберем режим "Датчик цвета" - "Измерение" - "Цвет" (Рис. 1). В отличие от программного блока "Ожидание", программный блок "Переключатель" не ждет, пока наступит определенное событие, а проверяет текущее состояние и выполняет программные блоки, находящиеся в контейнере, сопоставленном текущему состоянию.

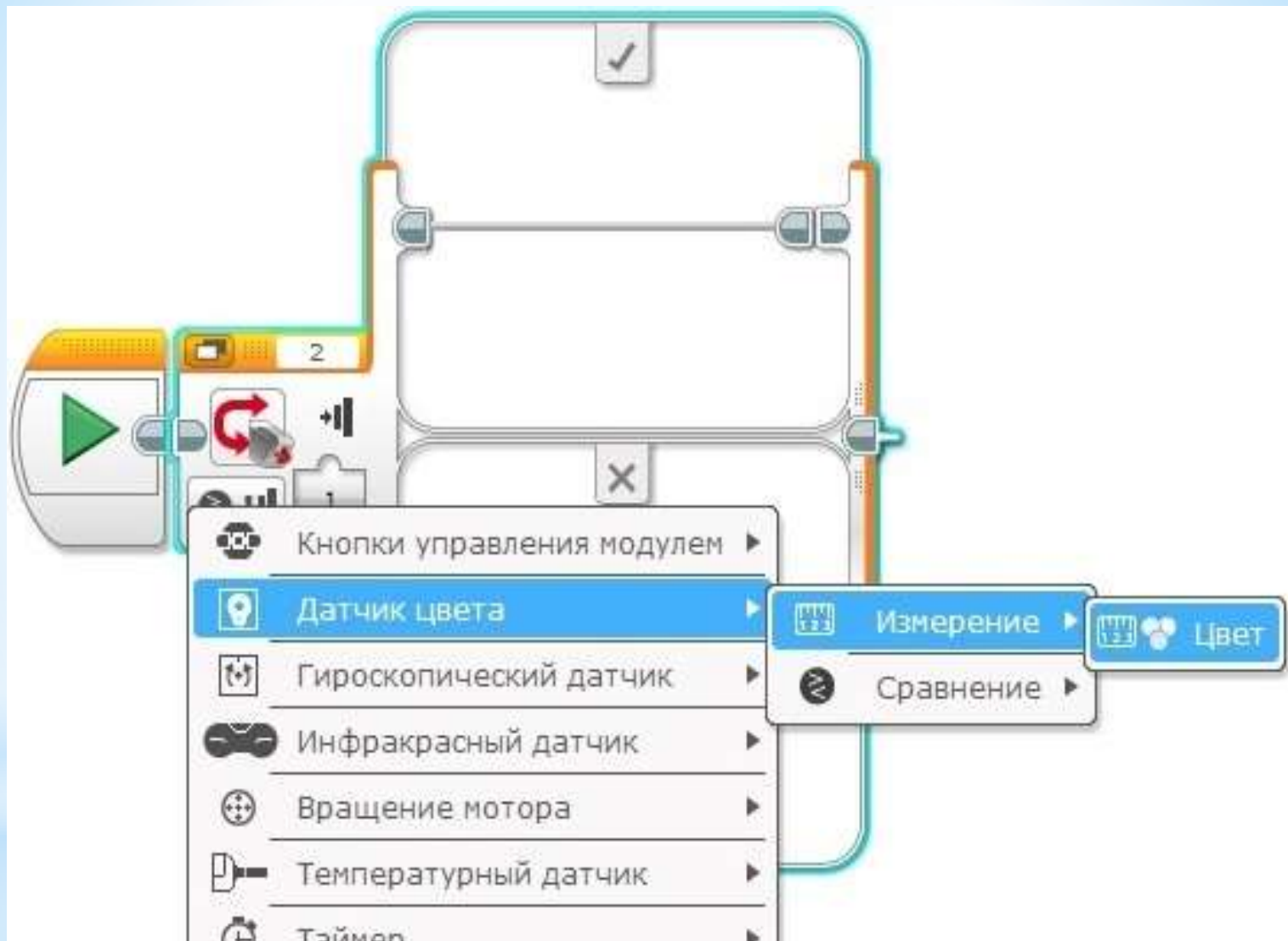


Рис. 1

Рассмотрим подробнее настройки программного блока "Переключатель":

- * выбранный режим устанавливает изображение датчика цвета в блоке (Рис. 2 поз. 1),
- * порт, к которому подключен датчик, отображается в соответствующем поле блока (Рис. 2 поз. 2),
- * в настройках каждого программного контейнера выбирается значение, в соответствии с которым будут выполняться программные блоки, вложенные в этот контейнер (Рис. 2 поз. 3),
- * один из контейнеров должен быть объявленным "Вариантом по умолчанию" - в случае, если значению, полученному от датчика, не соответствует ни один контейнер, то выполняется контейнер, объявленный "Вариантом по умолчанию" (Рис.2 поз. 4),
- * Кнопка "+" добавляет программный контейнер в блоке "Переключатель" (Рис. 2 поз. 5),
- * Программный блок "Переключатель" может автоматически растягиваться, чтобы вместить все блоки, помещаемые внутрь. С помощью меток, помеченных красными стрелками, можно самому изменять размеры блока (Рис.2).

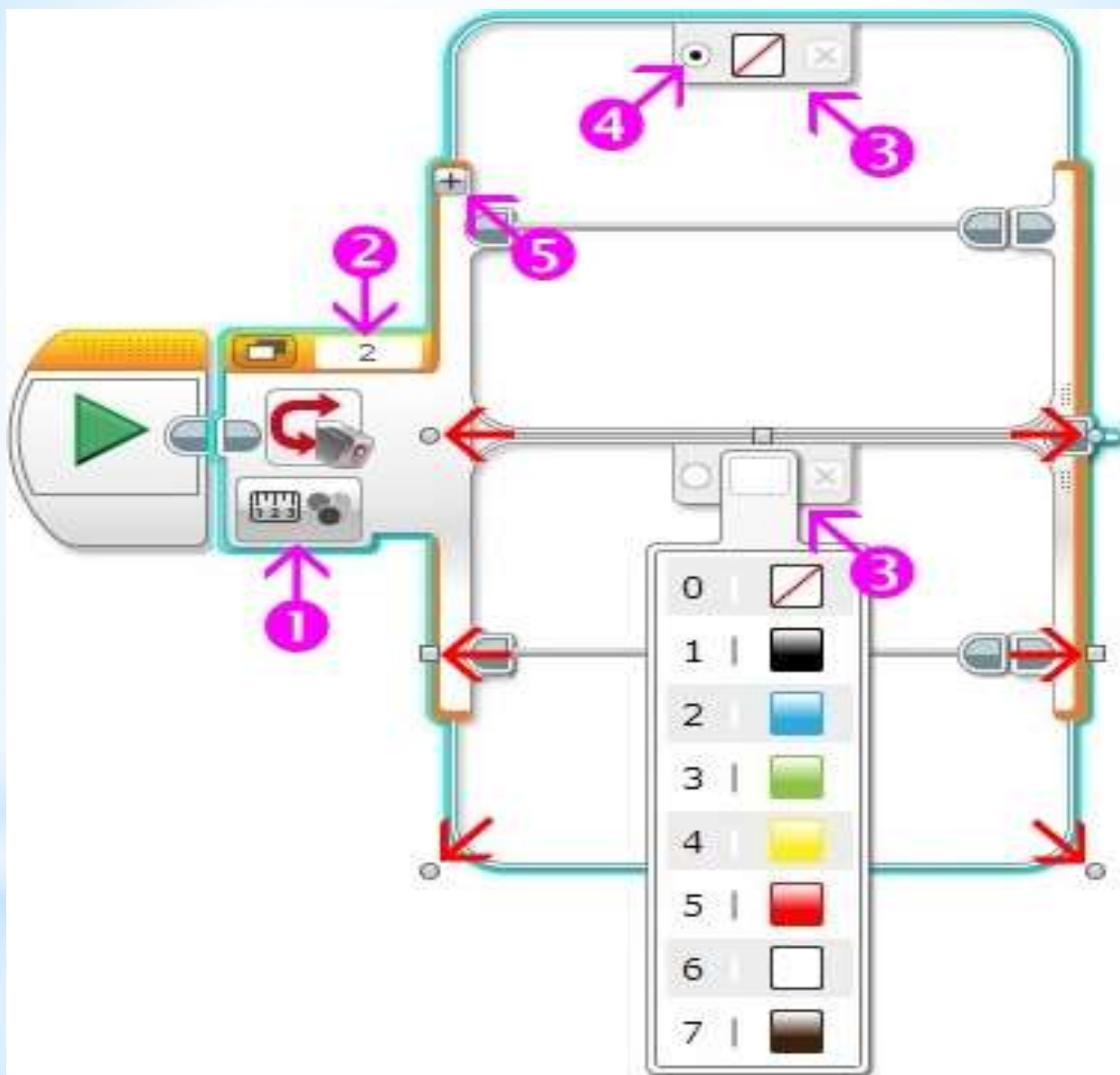


Рис. 2

Продолжим формирование программного блока "Переключатель":

- * создадим необходимое количество контейнеров, соответствующее количеству цветов для распознавания + вариант "Без цвета",
- * в настройках контейнеров установим распознаваемые цвета,
- * вариантом по умолчанию выберем вариант "Без цвета",
- * в каждый контейнер кроме варианта "Без цвета" (этот контейнер останется пустым) поместим программный блок "Звук" зеленой палитры.
- * каждому цвету сопоставим соответствующий звуковой файл.

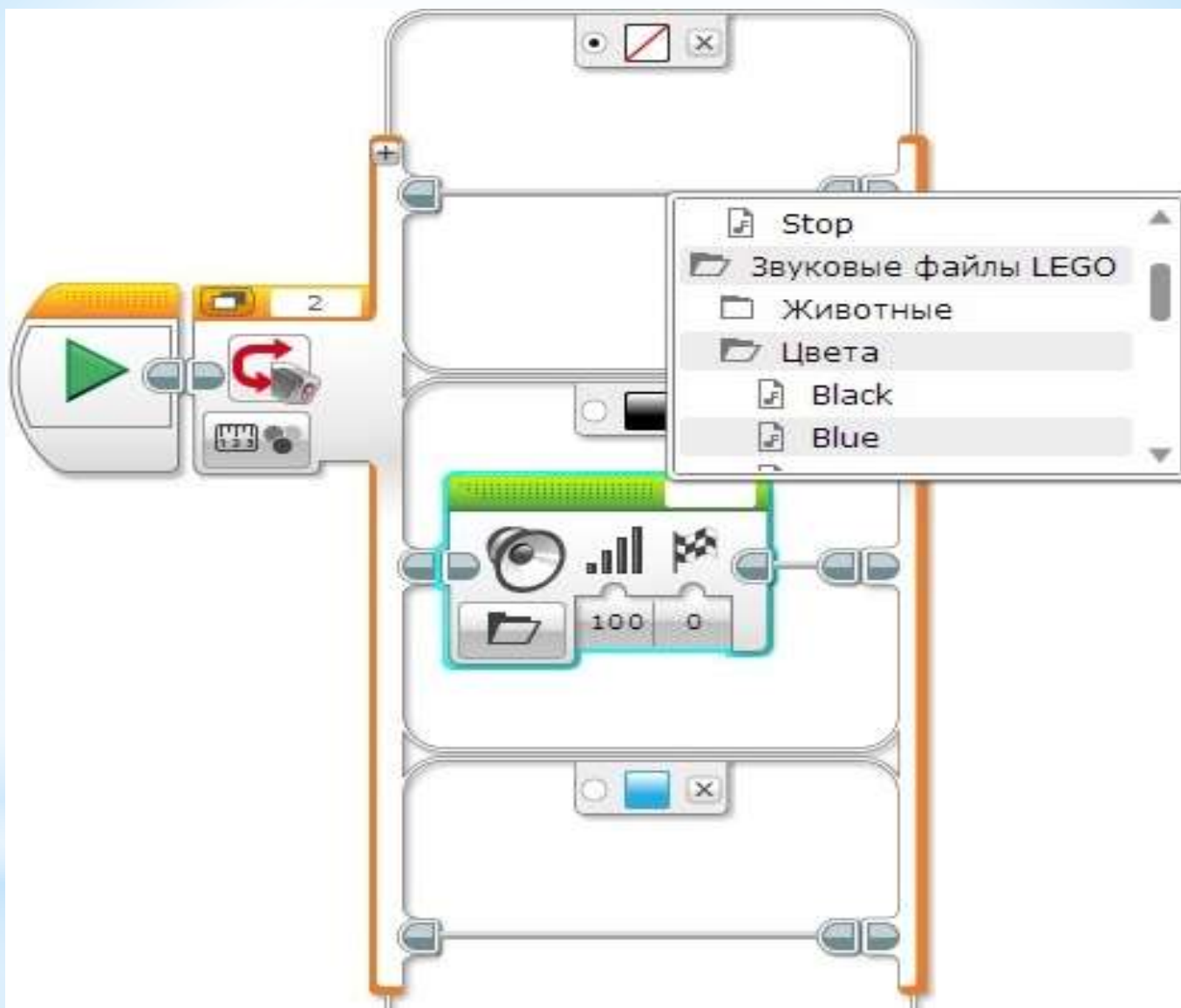


Рис. 3

- * Наш программный блок "Переключатель" значительно увеличился в размерах. Специальная кнопка (Рис. 4 поз. 1) позволяет переключить режим отображения блока на экране на "Вид с вкладками". Изменим размеры блока для комфортного визуального отображения.

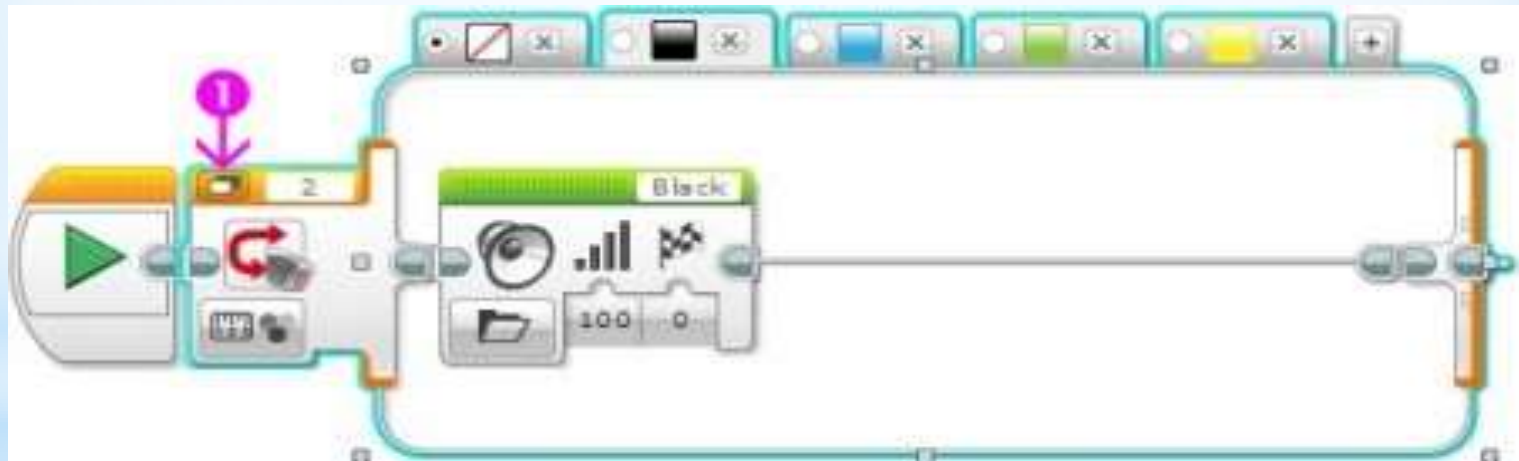


Рис. 4

- * Осталось вставить наш настроенный программный блок "Переключатель" внутрь программного блока "Цикл" Оранжевой палитры. Программа готова! Загрузим её в робота и протестируем работу! (Рис. 5)

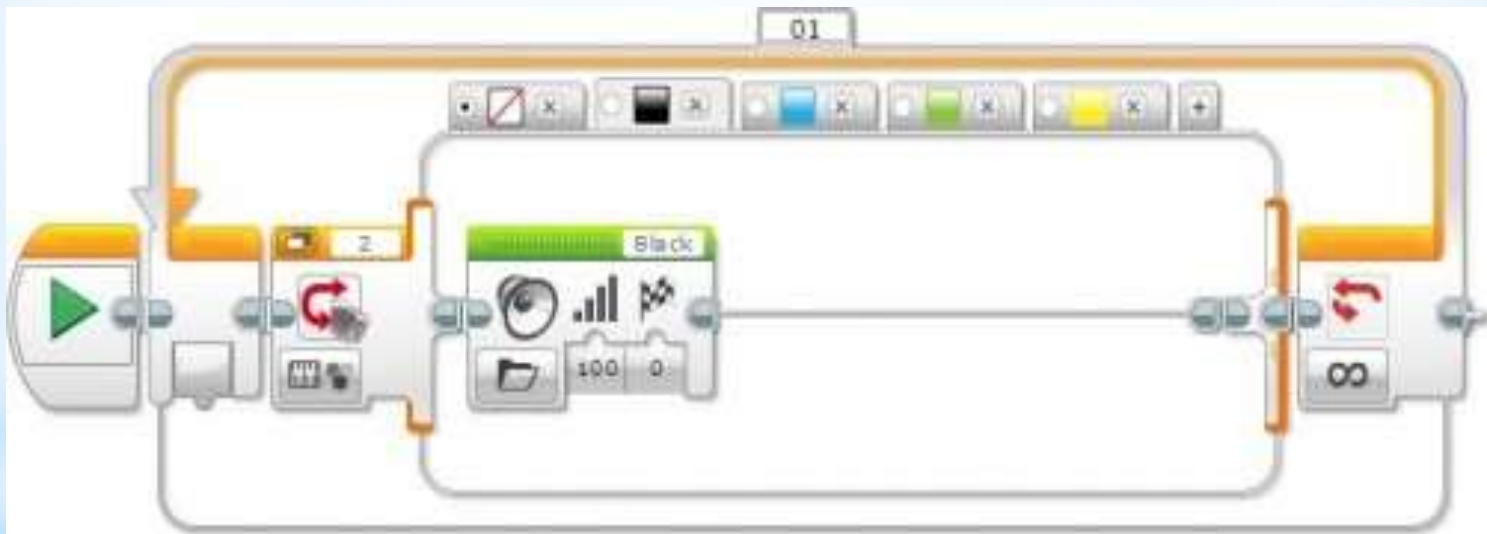


Рис. 5

Оранжевая палитра, программный блок "Прерывание цикла"

Добавим в нашу программу движение. Сделаем следующее поле для выполнения задания:

- * Возьмем белый лист бумаги формата А4 или А3;
- * Нанесем на него последовательно, на равном расстоянии несколько цветных полос. Полосы можно наклеить из цветной бумаги, цветной изоленты или нарисовать и закрасить;
- * Последнюю полосу сделаем черного цвета (Рис. 6).

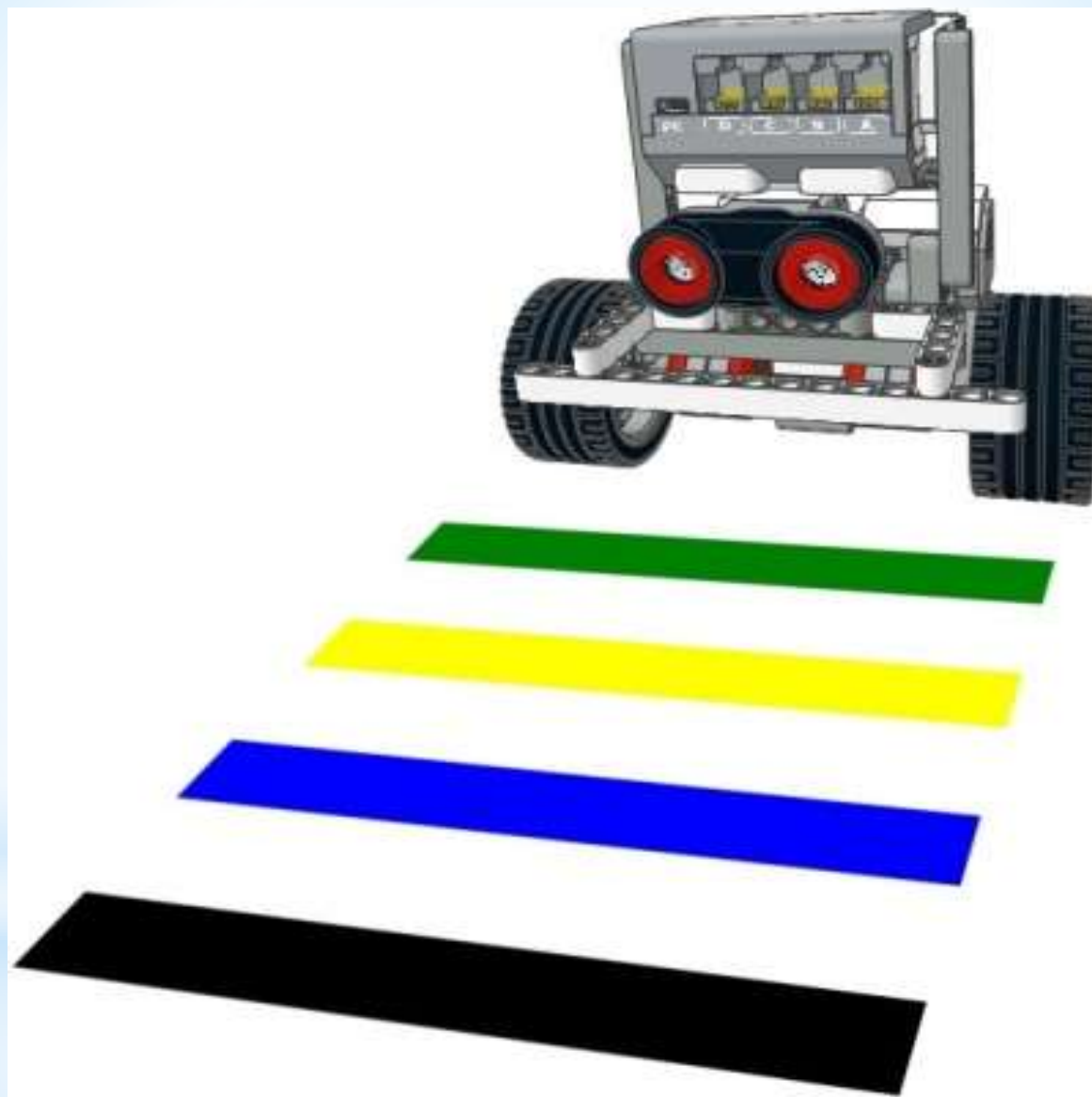


Рис. 6

- * **Задача №2:** необходимо написать программу прямолинейного движения робота, называющего цвета полос, над которыми он проезжает. При достижении черной полосы робот проговаривает "Stop" и останавливается.
- * За основу решения данной задачи возьмем программу, решающую **Задачу №1**. При решении **Задачи №2** нам потребуется прервать выполнение цикла. Этой цели служит программный блок "Прерывание цикла" Оранжевой палитры. С помощью данного блока можно организовать выход из цикла, заданного параметром "Имя прерывания" (Рис. 7 поз. 1).

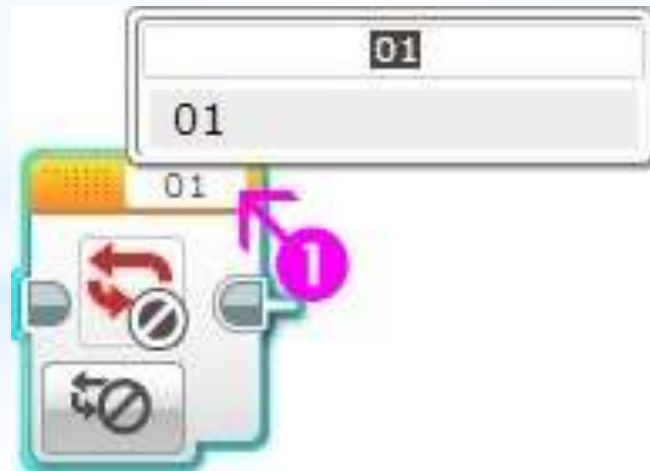


Рис. 7

Решение Задачи №2

- * Внутри цикла перед программным блоком "Переключатель" добавим программный блок "Рулевое управление", тем самым заставим нашего робота двигаться (Рис. 8 поз. 1). Во время движения робот будет проверять текущее состояние датчика цвета и произносить название цвета. Если полосы будут широкими, а робот будет двигаться медленно, то, возможно, он станет произносить название цвета более одного раза, так как проверка цвета будет происходить неоднократно. Если такое положение дел вас не устроит - увеличьте скорость робота, чтобы он быстрее проезжал цветные полосы.
- * В соответствии с условием задачи нам надо изменить поведение контейнера программного блока "Переключатель" для черного цвета.
- * В программном блоке "Звук" изменим звуковой файл "Black" на "Stop" (Рис. 8 поз. 2).
- * Добавим в контейнер программный блок, выключающий моторы (Рис. 8 поз. 3).
- * Нам требуется прервать выполнение программного блока "Цикл", чтобы завершить выполнение программы. Для этого поместим в контейнер программный блок "Превышение цикла" Оранжевой палитры (Рис. 8 поз. 4). У данного программного блока существует только одна настройка - название прерываемого цикла. В сложной программе со множеством циклов важно правильно устанавливать эту настройку, чтобы остановить выполнение нужного цикла (Рис. 8 поз. 5). В нашей программе за программным блоком "Цикл" отсутствуют другие программные блоки, поэтому программа завершится.

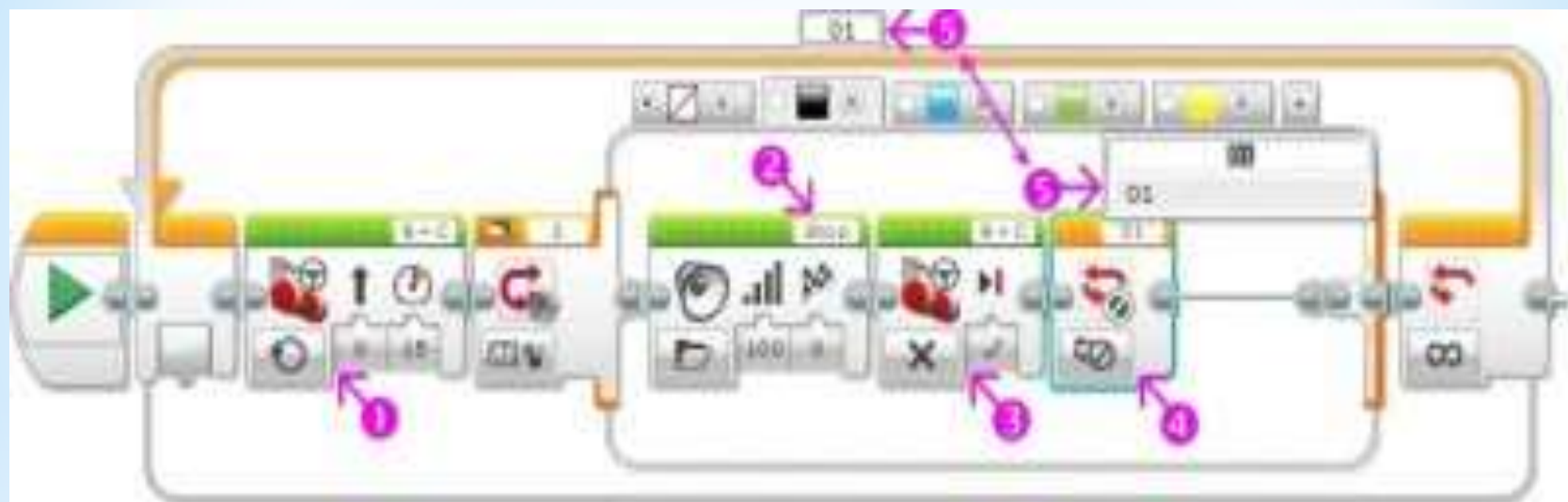


Рис. 8

Датчик цвета - режим "Яркость отраженного света"

- * Итак, мы приступаем к изучению следующего режима работы датчика цвета, который называется "Яркость отраженного света". В этом режиме датчик цвета направляет поток красного света на близкорасположенный предмет или поверхность и измеряет количество отраженного света. Более темные предметы будут поглощать световой поток, поэтому датчик будет показывать меньшее значение, по сравнению с более светлыми поверхностями. Диапазон значений датчика измеряется от 0 (очень темный) до 100 (очень яркий). Данный режим работы датчика цвета используется во множестве задач по робототехнике, например, для организации движения робота по заданному маршруту вдоль черной линии, нанесенной на белое покрытие. При использовании этого режима рекомендуется располагать датчик таким образом, чтобы расстояние от него до исследуемой поверхности составляло примерно 1 см (Рис. 9).

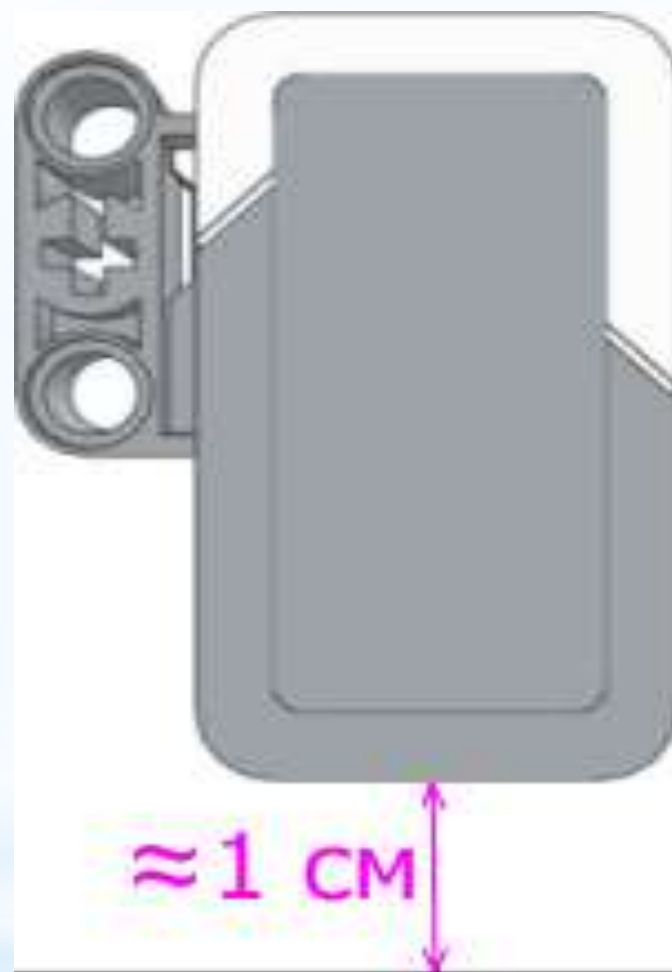


Рис. 9

- * Датчик цвета уже подключен к порту "2" модуля EV3.
- * Установим робота, таким образом, чтобы датчик цвета расположился над белой поверхностью. "Страницу аппаратных средств" среды программирования переключим в режим "Просмотр портов" (Рис. 10 поз. 1). В этом режиме мы можем наблюдать все выполненные нами подключения. На Рис. 10 отображено подключение к портам "B" и "C" двух больших моторов, а к порту "2" - датчика цвета.

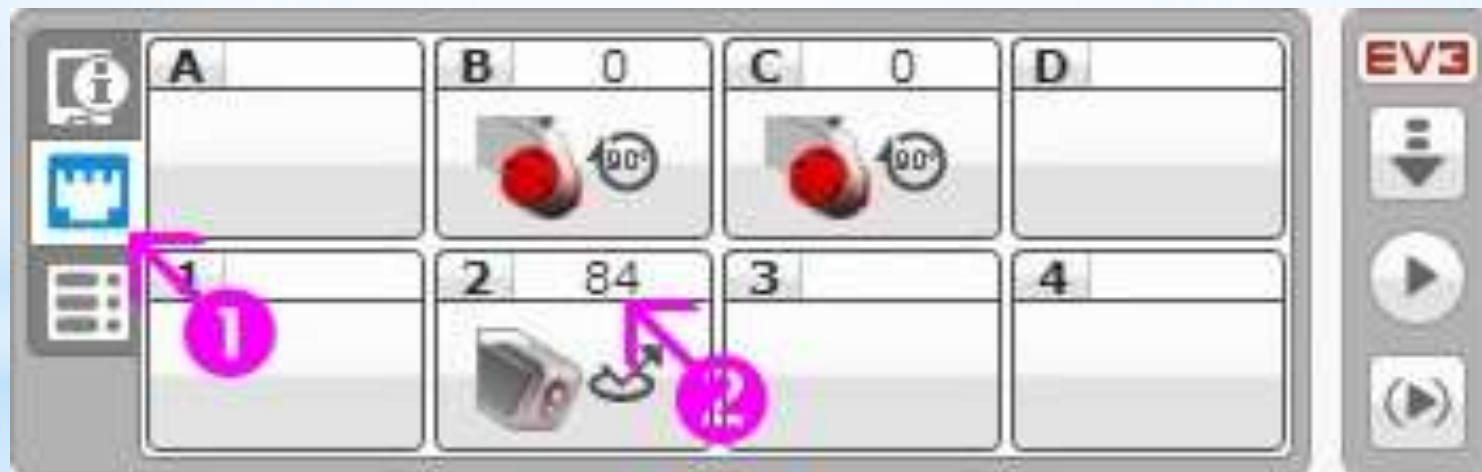


Рис. 10

- * Для выбора варианта отображения показаний датчиков необходимо нажать на изображение датчика и выбрать нужный режим (Рис. 11)

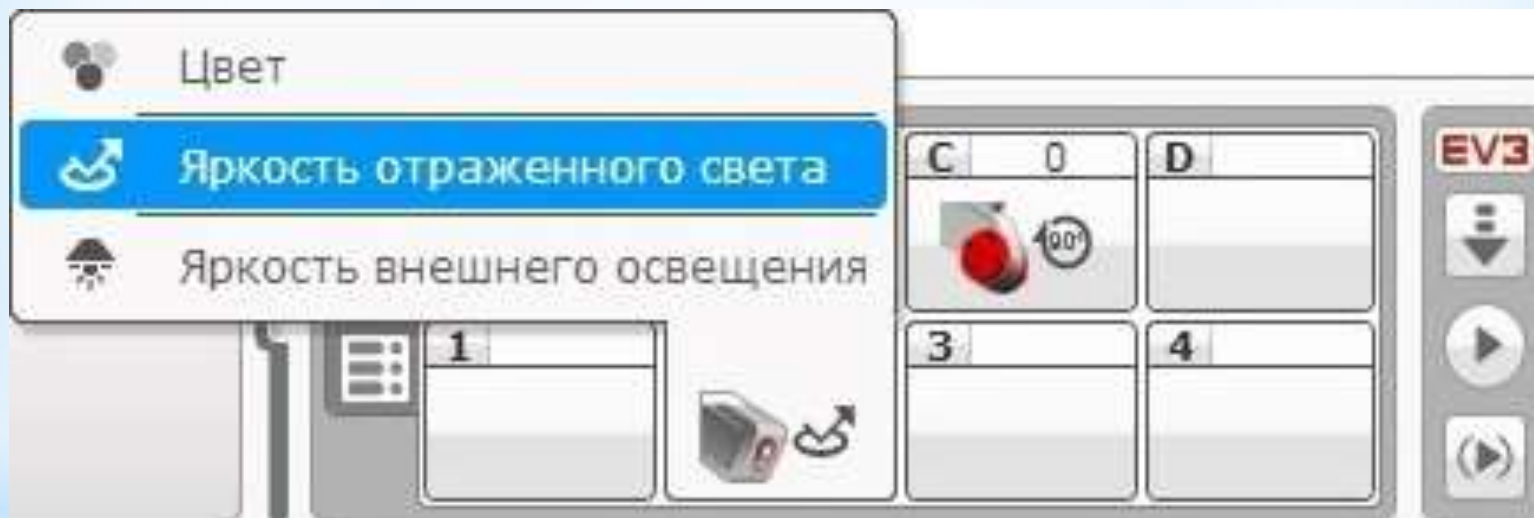


Рис. 11