**Урок №6 - Продолжаем изучать датчик цвета**

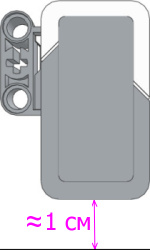
**Содержание урока**

**Введение:**

На этом уроке мы продолжим изучать использование датчика цвета. Материал, изложенный ниже, очень важен для дальнейшего изучения курса робототехники. После того, как мы научимся использовать все датчики конструктора Lego mindstorms EV3, при решении множества практических задач, будем опираться на знания, полученные на этом занятии.

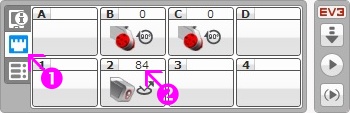
**6.1. Датчик цвета – режим "Яркость отраженного света"**

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. 1)**.



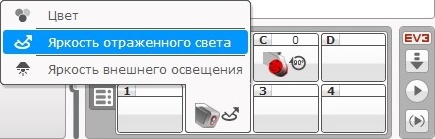
**Рис. 1**

Перейдем к практическим занятиям: датчик цвета уже установлен на нашем роботе и направлен вниз к поверхности покрытия, по которому будет передвигаться наш робот. Расстояние между датчиком и полом соответствует рекомендуемому. Датчик цвета уже подключен к порту **"2"** модуля EV3. Давайте загрузим среду программирования, подключим робота к среде и для проведения замеров воспользуемся полем с цветными полосами, изготовленным нами для выполнения заданий [Раздела 5.4 Урока №5](https://robot-help.ru/lessons/lesson-5.html#a54). Установим робота, таким образом, чтобы датчик цвета расположился над белой поверхностью. **"Страницу аппаратных средств"** среды программирования переключим в режим **"Просмотр портов" (Рис. 2 поз. 1)**. В этом режиме мы можем наблюдать все выполненные нами подключения. На **Рис. 2** отображено подключение к портам **"B"** и **"C"** двух больших моторов, а к порту **"2"** - датчика цвета.



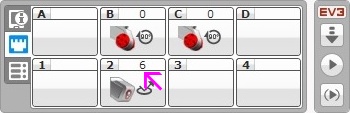
**Рис. 2**

Для выбора варианта отображения показаний датчиков необходимо нажать на изображение датчика и выбрать нужный режим **(Рис. 3)**



**Рис. 3**

На **Рис. 2 поз. 2** мы видим, что значение показания датчика цвета над белой поверхностью равно **84**. В вашем случае может получиться другое значение, ведь оно зависит от материала поверхности и освещения внутри помещения: часть освещения, отражаясь от поверхности, попадает на датчик и влияет на его показания. Установив робота таким образом, чтобы датчик цвета расположился над черной полосой, зафиксируем его показания **(Рис. 4)**. Попробуйте измерить самостоятельно значения отраженного света над оставшимися цветными полосами. Какие значения у вас получились? Напишите ответ в комментарии к этому уроку.



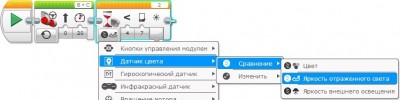
**Рис. 4**

Давайте теперь порешаем практические задачи.

**Задача №11:** необходимо написать программу движения робота, останавливающегося при достижении черной линии.

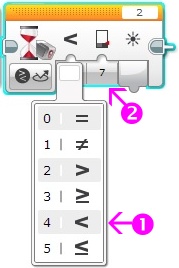
**Решение:**

Проведенный эксперимент показал нам, что при пересечении черной линии, значение датчика цвета в режиме **"Яркость отраженного света"** равняется **6**. Значит, для выполнения **Задачи №11** наш робот должен двигаться прямолинейно, пока искомое значение датчика цвета не станет меньше **7**. Воспользуемся уже знакомым нам программным блоком **"Ожидание"** Оранжевой палитры. Выберем требуемый условию задачи режим работы программного блока **"Ожидание" (Рис. 5).**

[](https://robot-help.ru/images/lego-mindstorms-ev3/lessons/lesson-6/005.jpg)

**Рис. 5**

Необходимо также настроить параметры программного блока **"Ожидание"**. Параметр **"Тип сравнения" (Рис. 6 поз. 1)** может принимать следующие значения: **"Равно"**=0, **"Не равно"**=1, **"Больше"**=2, **"Больше или равно"**=3, **"Меньше"**=4, **"Меньше или равно"**=5. В нашем случае установим **"Тип сравнения"** в значение **"Меньше"**. Параметр **"Пороговое значение"** установим равным **7 (Рис.6 поз. 2)**.



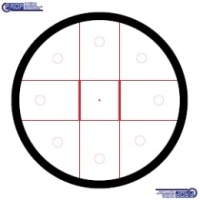
**Рис. 6**

Как только установится значение датчика цвета меньше **7**, что случится, когда датчик цвета окажется расположенным над черной линией, нам необходимо будет выключить моторы, остановив робота. Задача решена **(Рис. 7)**.



**Рис. 7**

Для продолжения занятий нам понадобится изготовить новое поле, представляющее собой черную окружность диаметром примерно 1 метр, нанесенную на белое поле. Толщина линии окружности равняется 2 - 2,5 см. Для основы поля можно взять один лист бумаги размером A0 (841x1189 мм), склеить вместе два листа бумаги размером A1 (594x841 мм). На этом поле разметить линию окружности и закрасить её черной тушью. Можете также скачать макет поля, выполненный в формате Adobe Illustrator, а затем заказать его печать на баннерной ткани в типографии. Размер макета равен 1250x1250 мм. (Просмотреть скачанный ниже макет можно, открыв его в программе Adobe Acrobat Reader)

[](https://robot-help.ru/images/lego-mindstorms-ev3/fields/circle-skittle.ai)

Данное поле пригодится нам для решения нескольких классических задач курса робототехники.

**Задача №12:** необходимо написать программу для робота, передвигающегося внутри круга, окантованного черной окружностью по следующему правилу:

* робот движется вперед прямолинейно;
* достигнув черной линии, робот останавливается;
* робот отъезжает назад на два оборота моторов;
* робот поворачивает вправо на 90 градусов;
* движение робота повторяется.

Знания, полученные на предыдущих уроках, помогут вам самостоятельно создать программу, решающую Задачу №12.

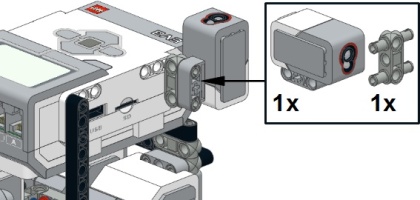
**Решение задачи №12**

К работе датчика цвета в режиме **"Яркость отраженного света"** мы еще неоднократно вернемся, когда будем рассматривать алгоритмы движения вдоль черной линии. А пока разберем третий режим работы датчика цвета.

**6.2. Датчик цвета – режим "Яркость внешнего освещения"**

Режим работы датчика цвета **"Яркость внешнего освещения"** очень похож на режим **"Яркость отраженного света"**, только в этом случае датчик не излучает освещение, а измеряет естественное световое освещение окружающей среды. Визуально данный режим работы датчика можно определить по слабо светящемуся синему светодиоду. Показания датчика изменяются от **0** (отсутствие света) до **100**(самый яркий свет). При решении практических задач, требующих измерения внешнего освещения, рекомендуется располагать датчик, так, чтобы датчик оставался максимально открытым и не загораживался другими деталями и конструкциями.

Давайте закрепим датчик цвета на нашем роботе так же, как мы крепили датчик касания в Уроке №4 **(Рис. 9)**. Подключим датчик цвета кабелем к порту **"2"** модуля EV3. Перейдем к решению практических задач.



**Рис. 9**

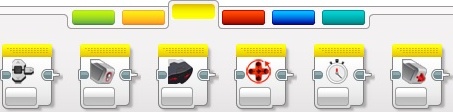
**Задача №13:** необходимо написать программу, изменяющую скорость движения нашего робота в зависимости от интенсивности внешнего освещения.

Чтобы решить эту задачу, нам надо узнать, как получать текущее значение датчика. А поможет нам в этом Желтая палитра программных блоков, которая называется **"Датчики"**.

**6.3. Желтая палитра - "Датчики"**

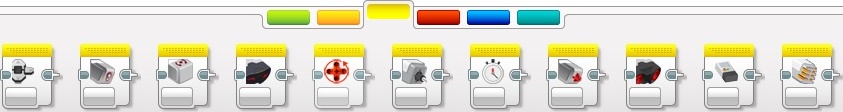
Желтая палитра среды программирования Lego mindstorms EV3 содержит программные блоки, позволяющие получать текущие показания датчиков для дальнейшей обработки в программе. В отличие, например, от программного блока **"Ожидание"** Оранжевой палитры, программные блоки Желтой палитры сразу же передают управление к следующим за ними программным блокам.

Количество программных блоков Желтой палитры отличается в домашней и образовательной версии среды программирования. В домашней версии среды программирования отсутствуют программные блоки для датчиков, не входящих в домашнюю версию конструктора. Но, при необходимости, их можно самостоятельно [подключить](https://robot-help.ru/tips/how-to-add-sensors-ev3-software.html).



**Желтая палитра. Домашняя версия среды программирования.**

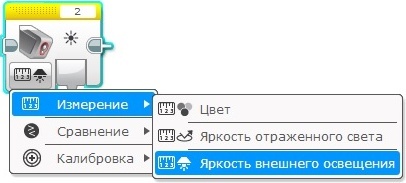
Образовательная версия среды программирования содержит программные блоки для всех датчиков, которые можно использовать с конструктором Lego mindstorms EV3.

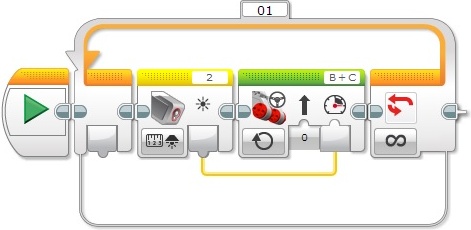


**Желтая палитра. Образовательная версия среды программирования.**

Вернемся же к решению **Задачи №13** и посмотрим, как можно получать и обрабатывать показания датчика цвета. Как мы уже знаем: диапазон значений датчика цвета в режиме **"Яркость внешнего освещения"** находится в пределах от **0** до **100**. Такой же диапазон у параметра, регулирующего мощность моторов. Попробуем показанием датчика цвета регулировать мощность моторов в программном блоке **"Рулевое управление"**.

**Решение:**

1. Считать текущее показание датчика цвета в режиме **"Яркость внешнего освещения";**
2. Подать полученное с датчика цвета значение на вход параметра **"Мощность"** программного блока **"Рулевое управление";**
3. Повторять команды **1 - 2** в бесконечном цикле.



**Рис. 10**

Давайте загрузим получившуюся программу в робота и запустим её на выполнение. Робот поехал медленно? Включим светодиодный фонарик и попробуем подносить его к датчику цвета на разном расстоянии. Что происходит с роботом? Закроем датчик цвета ладонью - что случилось в этом случае? Напишите ответы на эти вопросы в комментарии к уроку.