

# ПРОГРАММИРОВАНИЕ РОБОТ ТЕХНИЧЕСКИХ СРЕДСТВ



**ПЕРМСКИЙ КРАЙ ЛЫСЬВА  
ПАВЛЕНКО ВАЛЕРИЙ ВЛАДИМИРОВИЧ**

- Основные понятия
  - [Помощь и поддержка](#)
  - [Калибровка датчика](#)
  - [Шины данных](#)
  - [Шины данных - Повреждение](#)
  - [Мультизагрузка NXT](#)
  - [Файлы и память устройства NXT](#)
  - [Профили](#)
  - [Цепочка программы](#)
  - [Простой текст](#)
  - [Горячие клавиши NXT](#)
  - [Начальная точка](#)
  - [Обновление операционной системы NXT](#)
  - [Создание Проекта](#)
- Основная группа Блоков
  - [Блок «Экран»](#)
  - [Блок «Цикл»](#)
  - [Блок «Движение»](#)
  - [Блок «Писать/Играть»](#)
  - [Блок «Звук»](#)
  - [Блок «Переключатель»](#)
  - [Блок «Ожидание»](#)
- Блоки группы Действия
  - [Блок «Экран»](#)
  - [Блок «Лампа»\\*](#)
  - [Блок «Мотор»](#)
  - [Блок «Отправить сообщение»](#)
  - [Блок «Звук»](#)
- Блоки группы Датчики
  - [Блок «Датчик освещённости»](#)
  - [Блок «Кнопки NXT»](#)
  - [Блок «Приём сообщений»](#)
  - [Блок «Датчик оборотов»](#)
  - [Блок «Датчик звука»](#)
  - [Блок «Датчик температуры»](#)
  - [Блок «Таймер»](#)
  - [Блок «Датчик касания»](#)
  - [Блок «Датчик расстояния»](#)
- Блоки группы Операторы
  - [Блок «Цикл»](#)
  - [Блок «Стоп»](#)
  - [Блок «Переключатель»](#)
  - [Блок «Ожидание»](#)
- Блоки группы Данные
  - [Блок «Сравнение»](#)
  - [Блок «Логика»](#)
  - [Блок «Математика»](#)
  - [Блок «Случайное число»](#)
  - [Блок «Интервал»](#)
  - [Блок «Переменная»](#)
  - [Блок «Константа»](#)
- Блоки группы Дополнения
  - [Блок «Калибровка»](#)
  - [Блок «Доступ к файлам»](#)
  - [Блок «Не засыпать»](#)

- [Блок «Число в текст»](#)
- [Блок «Обнулить мотор»](#)
- [Блок «Текст»](#)
- [Блок «Bluetooth соединение»](#)
- [Блок «Начать регистрацию данных»](#)
- [Блок «Остановить регистрацию данных»](#)
- Пользовательские Блоки
  - [«Мой Блок»](#)
- Регистрация данных
  - [Анализ данных](#)
  - [Панель параметров эксперимента](#)
  - [Пульт регистрации данных](#)
  - [Таблица данных](#)
  - [График](#)
  - [Управление файлами экспериментов](#)
  - [Создание прогноза](#)
  - [Удаление данных](#)

## Помощь и поддержка

- Выберите тему слева, чтобы быстро получить ответ на свой вопрос.
- Самоучитель даёт возможность получить консультации по конструированию и программированию, а также ознакомиться с новыми идеями
- Ищете и не можете найти? Загляните в раздел «Помощь и Поддержка» на [Mindstormseducation.com](http://Mindstormseducation.com)



## Начинаем?

- Видеоматериал "Начинаем" поможет вам освоить программирование, научиться загружать и запускать свои первые программы NXT
- Видеоматериал "Обзор программного обеспечения" позволит ознакомиться с основными элементами интерфейса.

## Калибровка датчиков

Окружающие условия могут оказывать влияние на работу датчиков освещённости и датчиков звука. Поэтому калибровка этих датчиков для конкретных окружающих условий имеет большое значение, так как позволяет обеспечить их оптимальную работу.

Например, в очень светлой комнате неоткалиброванный датчик освещённости сможет регистрировать только высокие значения, тем самым не проявляя всех своих

потенциальных возможностей. Если программа создана в условиях низкой освещённости, робот, попав в условия высокой освещённости, может реагировать непредсказуемо. Калибровка датчика освещённости позволит ему регистрировать изменения в пределах всего допустимого диапазона при различных уровнях освещённости.

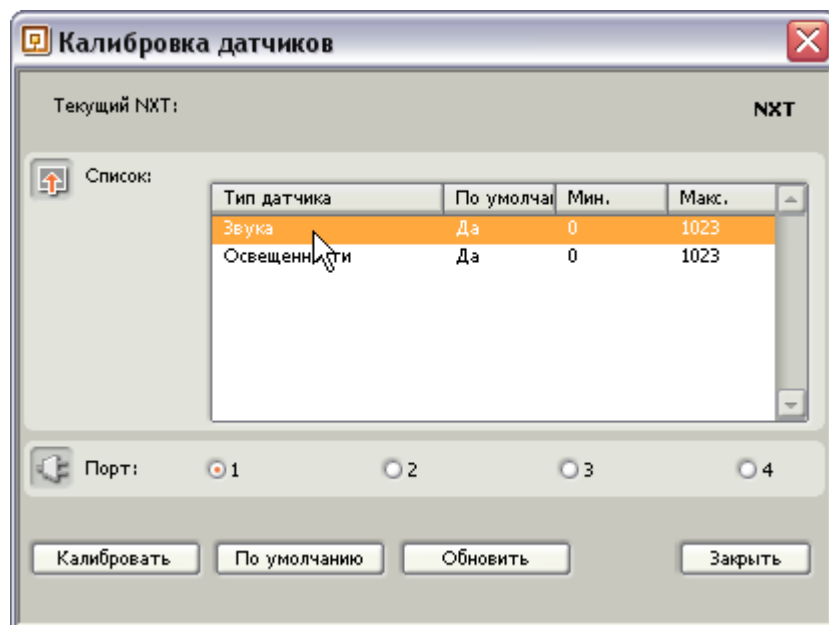
Существует два способа калибровки датчиков освещённости и звука в среде программирования MINDSTORMS NXT. Первый способ предполагает использование команды "Калибровка датчиков" в меню "Инструменты". Эта функция позволяет откалибровать датчик освещённости или звука лишь один раз для определенных условий (а не всякий раз при запуске программы). Например, если вы участвуете в Чемпионате Лиги First LEGO с ярко раскрашенными конкурсными столами, вы можете откалибровать датчик освещённости на столе один раз и больше не беспокоиться о новой калибровке при каждом запуске программы.

При втором варианте калибровки потребуются применение нескольких Блоков «Калибровка» в программе. Всякий раз при запуске программы эти блоки будут выполнять калибровку датчиков для текущих условий. (См. справку по разделу "Блок «Калибровка»".)

## Использование функции "Калибровка датчиков"

Перед использованием этой функции убедитесь в том, что NXT подключен к компьютеру, и что он включен. Также удостоверьтесь, что датчики, которые нужно откалибровать, подключены к NXT. Проверьте, чтобы они были подключены к соответствующим портам.

Убедившись в готовности, выберите в меню "Инструменты" команду "Калибровка датчиков". Откроется окно "Калибровка датчиков". Если NXT не подключен к компьютеру, открывшееся окно будет серым.



## Калибровка датчика освещённости

Выберите датчик освещённости и порт, к которому он подключен. Затем щёлкните на кнопке "Калибровать". После этого в NXT загрузится небольшая программа и автоматически запустится.

На дисплее NXT появится текст - "Min Value: (Минимальное значение:)". Направьте датчик освещённости на объект, который датчик должен регистрировать как темный. Нажмите оранжевую кнопку "Ввод" на NXT.

Затем появится текст - "Max Value: (Макс. значение:)". Направьте датчик освещённости на объекты, которые могут оказаться самыми яркими во время выполнения программы. Ещё раз нажмите оранжевую кнопку "Ввод" на NXT. Калибровка завершена.

## Калибровка датчика звука


Выберите датчик звука и порт, к которому он подключен. Затем щёлкните по кнопке "Калибровать". После этого в NXT загрузится небольшая программа и автоматически запустится.


На дисплее NXT появится текст - "Min Value: (Минимальное значение:)". Поместите датчик в условия, при которых он будет производить измерения во время выполнения программы, и где есть возможность измерить фоновые шумы. Нажмите оранжевую кнопку "Выбор" на NXT.

Затем появится текст - "Select Max: (выбрать макс. значение:)". Измерьте или произведите максимально громкий звук, который вы предполагаете регистрировать, и снова нажмите кнопку "Выбор". Калибровка завершена.

## Дополнительные сведения

Более подробная информация о возможностях датчиков и измерениях приведена в разделе "Технология" в Руководстве пользователя, которое поставляется вместе с MINDSTORMS NXT.

 Примечание: калибровочные значения, установленные при помощи функции "Калибровка датчиков", могут быть изменены при помощи Блока "Калибровка" непосредственно в программе, и наоборот. Превалирующим будет результат более поздней калибровки. Параметры калибровки будут сохраняться до тех пор, пока вы сами не измените эти значения или/и не сбросите их (см. ниже).

 Примечание: после завершения процедуры калибровки установленные значения будут действовать для отдельного датчика, независимо от того, к какому порту он будет подключен. Если при калибровке датчик освещённости подключен к порту 3, параметры калибровки сохранятся при его подключении и к порту 4.

## Восстановление калибровочных параметров, установленных по умолчанию

Если вы хотите начать заново и восстановить заводские параметры калибровки, следуйте инструкциям, приведенным ниже.

Прежде всего, включите NXT и подключите его к компьютеру.

Выберите в меню "Инструменты" команду "Калибровка датчиков". Откроется окно "Калибровка датчиков". Если NXT не подключен, это окно будет серым.

Выберите датчик, для которого вы хотите восстановить заводские настройки по умолчанию, и нажмите кнопку "По умолчанию".

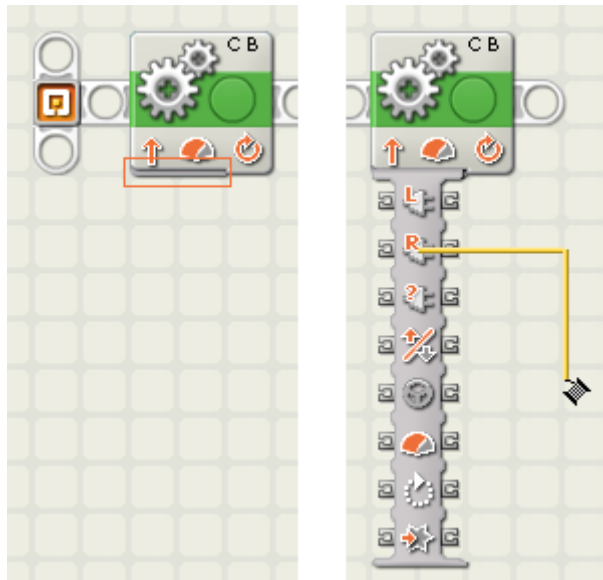
# Шина данных


Шины данных обеспечивают обмен информацией между программными блоками. Для того, чтобы многие блоки вообще работали, требуется подключение шин данных. Например, выходной сигнал из Программного блока случайных событий может быть отправлен только через шину данных.

## Открытие концентраторов данных и создание шин данных

Вы создаете шину данных путем “извлечения” ее из концентратора данных программного блока. Практически все программные блоки имеют концентраторы данных и могут поддерживать шины данных.

Поместив блок в рабочую область, откройте концентратор данных блока, щелкнув по кнопке в нижней левой части блока.



 **Примечание:** Иногда концентраторы данных не открываются полностью, и часть разъемов оказывается скрытыми. Откройте все разъемы концентраторов данных, повторно щелкнув в поле кнопки (как показано на левом изображении выше). Вы также можете закрыть концентратор данных (или скрыть любые неиспользуемые разъемы, не подключенные к шинам данных), щелкнув в поле кнопки. При этом рабочая область будет очищена.

## Извлечение шины данных

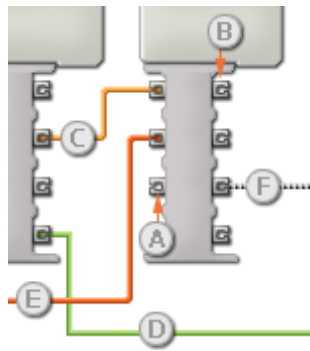
Курсор изменит свою форму, оказавшись над разъемом данных или возле него. Если вы затем нажмете на кнопку мыши и перетащите вправо, шина данных “развернется” так, что ее можно будет подключить к разъему на концентраторе данных другого блока.

## Удаление шины данных

Чтобы удалить шину данных, которая размещена между левым и правым разъемами данных, щелкните по правому разъему.

## Вход и выход

Шины данных, передающие информацию на блок (т.е, выполняющие ввод), подключены к разъемам, находящимся на левой стороне его концентратора данных. Шины данных, передающие информацию от блока (т.е, выполняющие вывод), подключены к разъемам, находящимся на правой стороне.



- [A] Входной разъем
- [B] Выходной разъем
- [C] Числовая шина данных (желтая)
- [D] Логическая шина данных (зеленая)
- [E] Текстовая шина данных (оранжевая)
- [F] Поврежденная шина данных (серая)

## Шины данных осуществляют передачу специфических типов данных

Каждая шина данных осуществляет обмен специфическими типами данных между блоками. Например, если шина данных перетянута с логического разъема на концентратор данных блока, ее можно подключить только к логическому разъему на концентраторе данных другого блока.

## Окраски шин данных

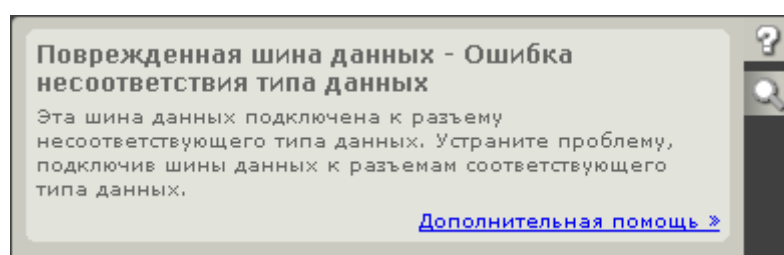
Шины данных различаются особыми окрасками:

- шины, передающие числовые данные, имеют желтую окраску
- шины, передающие логические (истина/ложь) данные, имеют зеленую окраску
- шины, передающие текстовые данные, имеют оранжевую окраску.

## Поврежденные шины данных

Если вы пытаетесь подключить шину данных к разъему несоответствующего типа данных, шина будет "поврежденной" и будет окрашена в серый цвет. Вы не сможете загрузить вашу программу, если она содержит поврежденные шины данных.

Если вы щелкнете по поврежденной шине, то в небольшом окне контекстной справки в правом нижнем углу рабочей области вы сможете прочесть информацию о причине повреждения.



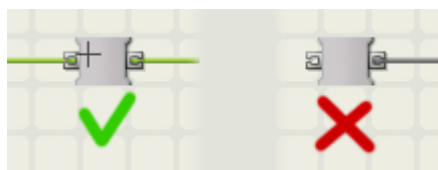


## Данные должны соответствовать возможному диапазону, в котором работает разъем

Если шина данных входа передает значение, не попадающее в допустимый диапазон разъема, к которому она подключена, блок либо проигнорирует такие данные, либо трансформирует данные, попадающие в этот диапазон. Для разъемов, допускающих небольшое количество значений (например, только 0, 1 или 2), разъем проигнорирует входящий сигнал, если передаваемое значение не будет вписываться в диапазон разъема.

Для разъемов, допускающих более широкий диапазон значений (например, 0 – 100), разъем усилит любой входящий сигнал, если он не соответствует диапазону. Например, если разъем питания программного блока перемещения получает входящий сигнал со значением 150, блок преобразует входящее значение в 100 (т.е., в числовое значение, лежащее в пределах диапазона разъема мощности).

## Передача данных от входного разъема на выходной разъем



Если для входного разъема есть соответствующий выходной разъем, входящие данные будут проходить от входного разъема на выходной разъем без изменений. В этом случае вы сможете использовать выходной разъем, если входной разъем подключен к шине данных входа; подключение шины данных выхода к такому выходному разъему без подключенной шины данных входа приведет к "поврежденной" шине данных выхода, и она будет окрашена в серый цвет.

## Поврежденные шины данных

Хотя возможность неправильной последовательности блоков в программе исключается, существует четыре возможных варианта неправильного подключения их к шинам данных. Такие ошибки подключения приводят к появлению "поврежденных" шин данных, которые имеют серый цвет. Программы, содержащие поврежденные шины данных, не могут быть загружены в устройство NXT.

Четыре возможных ошибки в части шины данных:

- [Несоответствие типа данных](#)
- [Цикл](#)
- [Отсутствие входного сигнала](#)
- [Слишком много входных сигналов](#)

### Несоответствие типа данных

Каждая шина данных осуществляет обмен специфическими типами данных между программными блоками. Например, если шина данных перетянута с логического разъема на концентратор данных блока, она будет осуществлять передачу данных для логических операций. Эта шина данных может быть подключена только к логическому входному разъему на концентраторе данных другого блока.

Аналогично, числовые выходные разъемы должны подключаться к числовым входным разъемам, а текстовые выходные разъемы - к текстовым входным разъемам.



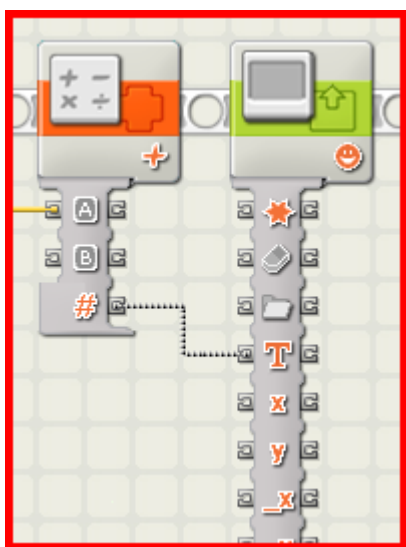
Ошибка несоответствия типа данных происходит в том случае, если вы подключаете шину данных между двумя разъемами, обрабатывающими различные типы данных, например, при подключении логического выходного разъема к текстовому входному разъему.

Избегайте подобной ошибки, обращая особое внимание на соответствие типов данных тех разъемов, к которым вы подключаете шины данных.

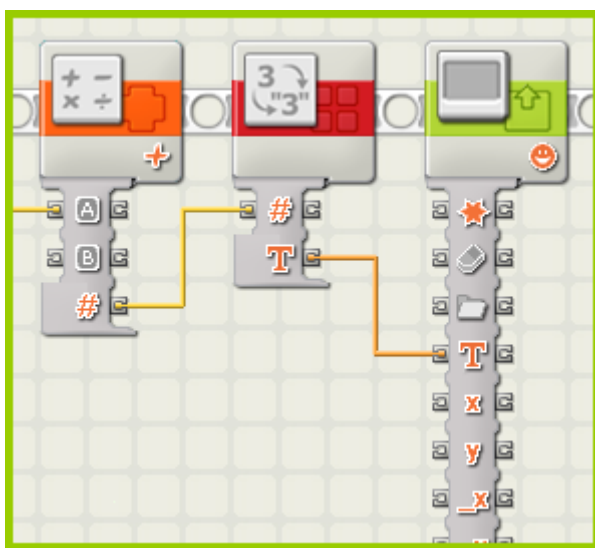
Пример:

Предположим, вы хотите вывести числовое значение на дисплей блока NXT.

В первой программе, показанной выше, шина данных пытается передать числовые данные на текстовый порт ввода. Шина данных "повреждена" и окрашена в серый цвет. Эта программа содержит ошибку и не может быть загружена.



В исправленном варианте программы для преобразования числовых данных в текстовые в начало был добавлен Блок форматирования текста (взятый из комплекта Расширенных блоков). Теперь текстовый выход с Блока форматирования текста соединен с текстовым входом Блока отображения. Эта программа не содержит ошибок и может быть загружена.



## Цикл

При наличии любого набора программных блоков и шин, если вы обнаруживаете канал шины данных, который дважды проходит через один и тот же блок, вы имеете дело с

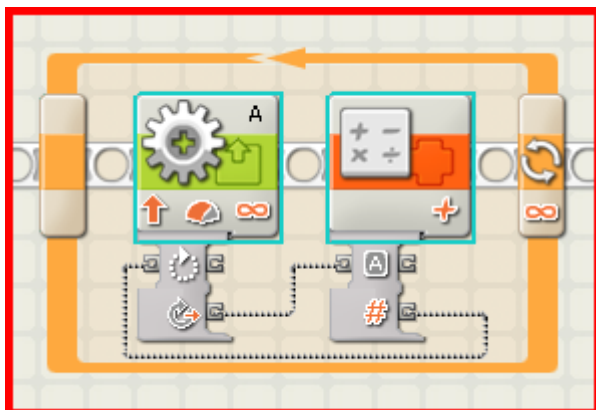
циклом. Это является причиной повреждения шины данных и появления ошибки в программе.

Решение проблемы - удалить одну или несколько шин из цикла.

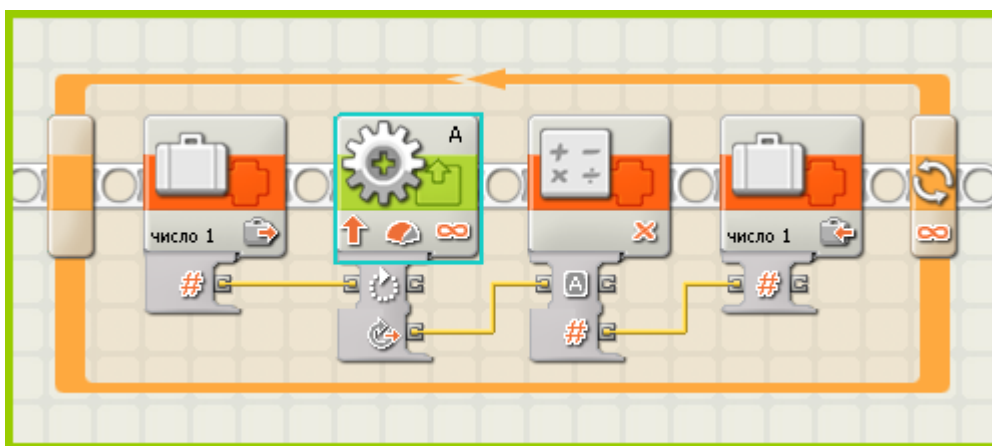
Пример:

Предположим, вы хотите создать программу, которая поддерживает удвоенное значение продолжительности для электродвигателя, считывая показания текущей продолжительности, умножая ее на два и отсылая это удвоенное значение на входной разъем электродвигателя.

Как показано ниже, программа не будет работать, так как шины данных образуют цикл. Вы видите, что канал шины данных дважды проходит через каждый из блоков.



В программе, показанной ниже, эта проблема решена следующим образом: переменное числовое значение используется для обратной отправки в цикл удвоенного значения продолжительности.



## Отсутствие входного сигнала

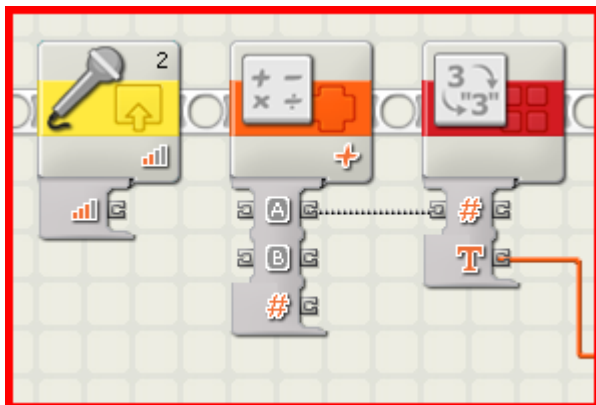
Шина данных не может работать без источника данных; если отсутствуют данные для передачи, шина будет помечена как "поврежденная".

Например, на концентраторе данных с входными и выходными разъемами для определенного атрибута выходной разъем будет работать только в том случае, если другая шина данных также подключена к входному разъему, выполняя для выходного разъема функцию источника данных.

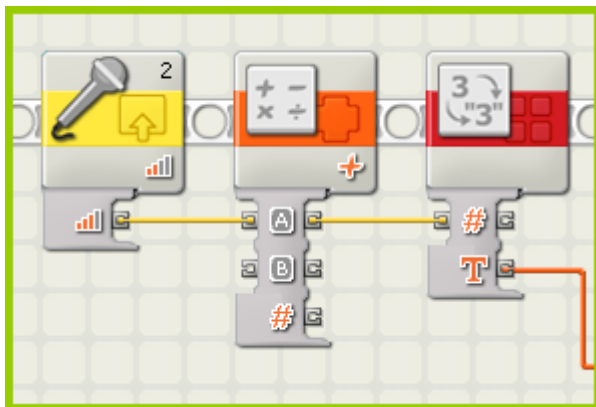
Пример:

Предположим, вы хотите передать данные через концентратор данных другого блока.

В программе, показанной ниже, поврежденная шина данных (серого цвета) не получает данные для передачи, потому что от входного разъема через нее не поступают никакие данные. Эта программа содержит ошибку и не может быть загружена.



В исправленном варианте программы, показанном ниже, шина данных входа подключена к входному разъему. Числовые данные от Программного блока датчика звука теперь могут передаваться через Математический программный блок к Блоку преобразования текста.

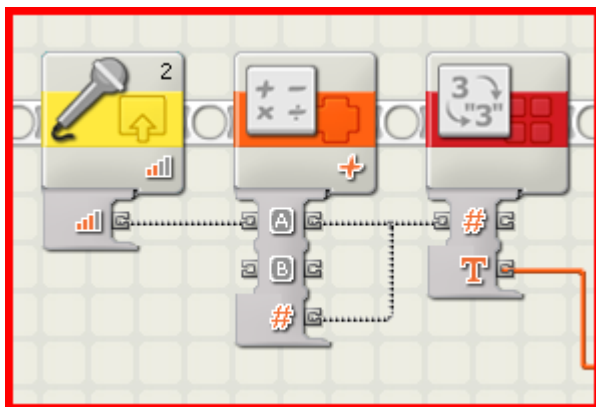


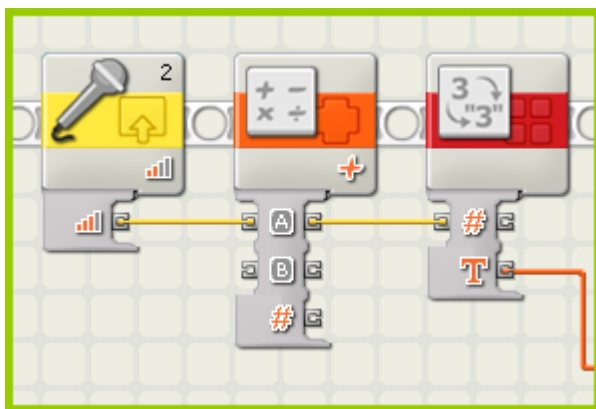
### Слишком много входных сигналов

Если к одному и тому же входному разъему подключено более одной шины данных, появятся поврежденные шины данных и программы с ошибками.

Решение проблемы - удалить или переместить поврежденные шины данных таким образом, чтобы каждый входной разъем был подключен только к одному выходному разъему.

На рисунках внизу показаны программа с ошибкой и поврежденной шиной данных, а также исправленная программа.





## Удаление шины данных

Чтобы удалить шину данных, которая размещена между левым и правым разъемами данных, щелкните по правому разъему.

# Мультизагрузка NXT

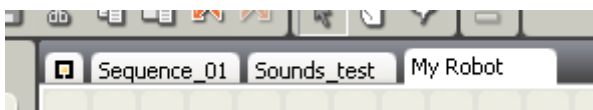
Эта функция позволяет загружать одну или несколько программ в несколько NXT. Она особенно полезна, если требуется загрузить в несколько NXT одинаковые программы, например, для участия в соревнованиях или для занятий в классе.

Функция мультизагрузки доступна только в версии среды программирования LEGO MINDSTORMS NXT, предназначенной для обучения.

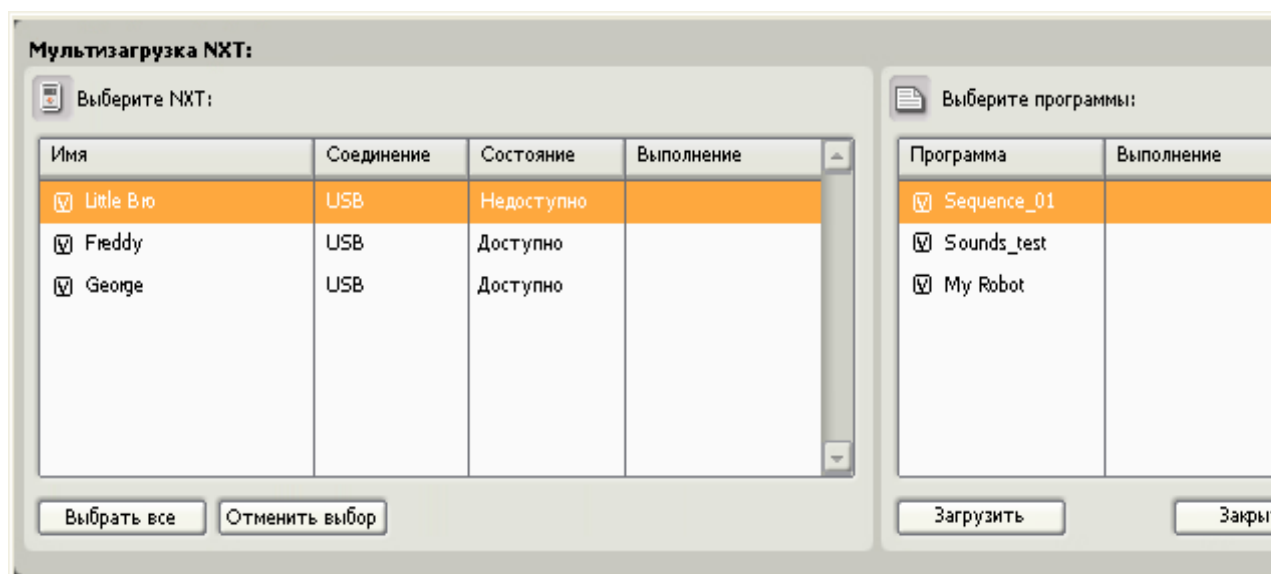
В NXT можно загрузить столько программ, насколько позволяет объём его памяти. Программы занимают различный объём памяти, в среднем он составляет 11 КБ. Когда память заполнится, некоторые программы можно удалить для освобождения места. Дополнительную информацию см. в файле справки «[Управление файлами и памятью NXT](#)».



Примечание: перед использованием данной функции необходимо открыть в среде программирования все программы, которые требуется загрузить в NXT. При одновременном открытии нескольких программ, их имена отображаются в виде вкладок, расположенных в верхней части рабочей области (как показано на рисунке ниже). Также необходимо удостовериться, что все NXT «доступны», для этого нажмите Сканировать (Scan). Иначе эти программы не появятся в окне «Мультизагрузка NXT».



Когда программы, которые требуется загрузить, открыты, в меню «Инструменты» выберите команду «Мультизагрузка NXT...». Откроется окно «Мультизагрузка NXT», показанное ниже.



1. Выберите списки всех доступных в настоящий момент устройств NXT.
2. «Выберите списки всех программ», открытых в настоящий момент программ.



Примечание: если NXT отсутствует в списке, закройте окно «Мультизагрузка NXT» и откройте окно NXT. Удостоверьтесь, что все NXT зарегистрированы.

## Мультизагрузка NXT

Сначала отметьте (флажками) те NXT, в которые нужно загрузить файлы, щёлкнув на окошках, расположенных слева от имен NXT. Можно выбрать все доступные NXT сразу, нажав кнопку «Выбрать всё», расположенную под списком. Если нажать кнопку «Отменить выбор всего», выбор всех NXT будет отменён.

Затем отметьте программы, которые требуется загрузить, щёлкнув на окошках, расположенных слева от имен программы.

Произведя выбор, нажмите кнопку «Загрузить», расположенную под списком программ. Начнется загрузка. Ход загрузки отображается на индикаторе выполнения.

Во время компиляции в списке «Программ» могут отображаться сообщения о четырёх возможных состояниях процесса:

- Пропущено — если эта программа не была отмечена
- Выполняется компиляция — если файл в настоящий момент компилируется
- Готово — когда данная программа скомпилирована
- Ошибка — если при компиляции программы произошла ошибка

После завершения компиляции всех выбранных программ начинается их загрузка. Во время загрузки в списке NXT могут отображаться сообщения о четырёх возможных состояниях:

- Пропущено — данный NXT не отмечен
- Выполняется загрузка — в настоящий момент выполняется загрузка на NXT
- Готово — загрузка всех программ на данный NXT завершена
- Ошибка — при загрузке произошла ошибка

После завершения загрузки программ в несколько NXT, нажмите кнопку «Закрыть».



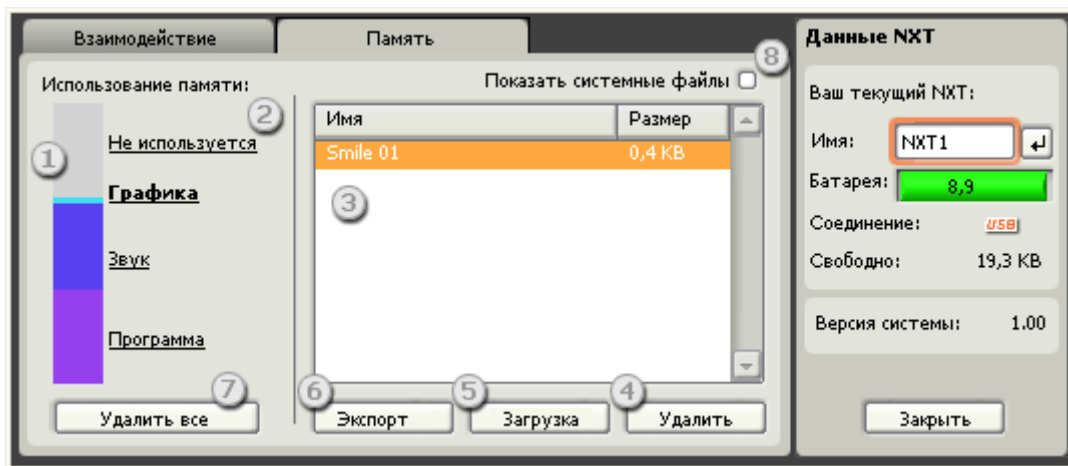
Примечание: чтобы совместно использовать программы на нескольких компьютерах (с возможностью их редактирования на них), необходимо обмениваться файлом программы (с расширением .gbt) по электронной почте или посредством USB накопителя.

## Управление файлами и памятью устройства NXT

Иногда возникает необходимость в управлении файлами и памятью устройства NXT. Выполнение этой операции возможно в том случае, если зайти во вкладку "Память" в окне NXT, которое открывается верхней левой кнопкой контроллера.



Вкладка памяти в окне NXT выводит графический обзор использования памяти на вашем устройстве NXT. Отсюда вы также можете управлять отправкой программ, звуковых, графических и прочих файлов с компьютера на устройство NXT, а также копировать и удалять файлы, уже находящиеся в устройстве NXT.



1. Графическое отображение используемой и свободной памяти.
2. Различные категории файлов (программные, звуковые, графические, прочие...), использующие ресурсы памяти устройства NXT. Щелкнув по наименованию категории, можно отобразить список этих типов файлов.
3. Список файлов в выбранной категории.
4. Удалить. При помощи этой кнопки выбранные файлы можно удалить из устройства NXT.
5. Выгрузить. При помощи этой кнопки файлы можно загрузить из вашего компьютера в устройство NXT.
6. Загрузить. При помощи этой кнопки выбранные файлы можно скопировать с устройства NXT на ваш компьютер.
7. Удалить все пользовательские файлы. При помощи этой кнопки удаляются все пользовательские файлы и очищается память устройства NXT.
8. Показать системные файлы. Выбрав это окно, можно увидеть все файлы, установленные в устройство NXT на заводе-производителе.

## Память устройства NXT

Каждое устройство NXT имеет доступную память около 130,7 килобайт. Часть этого объема памяти используется для хранения примеров программ, графических и звуковых файлов, которые уже загружены в устройство NXT до того, как вы его приобрели. Для файлов, которые вы создаете сами или загружаете, остается около 56 килобайт.

## Недостаточный объем памяти

Если объем памяти недостаточен, вы можете удалить некоторые или все файлы с устройства NXT. Для этого слева (1) изучите графическое представление памяти и определите, какие файлы занимают большую часть памяти. Щелкните по наименованию этой категории (2) и отобразится список ее файлов. Выберите в списке файлы, которые вам не нужны, и щелкните по кнопке "Удалить".

## Удаление всех файлов

Щелкнув по кнопке "Удалить все пользовательские файлы", вы можете удалить все файлы, созданные вами, и очистить память устройства NXT. В ходе этой операции будут удалены все файлы, когда-либо скопированные на устройство NXT, а освободившаяся память будет реорганизована таким образом, чтобы освободилось максимум места для новых файлов. В ходе этой операции не будут удалены системные файлы, загруженные в устройство NXT на заводе-производителе.



Примечание: В редких случаях вы можете получить сообщение об ошибке (недостатке памяти) даже тогда, когда графическое представление состояния памяти (1) свидетельствует о наличии в ней свободного места. Это связано с "фрагментацией" памяти. Устраните эту проблему, воспользовавшись кнопкой "Удалить все пользовательские файлы", но перед этим не забудьте создать резервные копии наиболее важных файлов.

Вы также можете удалить файлы, воспользовавшись пиктограммами на устройстве NXT. Чтобы получить подробную информацию, обратитесь к разделу "Технология NXT" в печатном Руководстве пользователя.

## Максимальное высвобождение памяти

Если удаление пользовательских файлов не влечет за собой заметного высвобождения памяти, вы можете удалить демонстрационные файлы и файлы с примерами, которые были загружены в устройство NXT на заводе-производителе. Эти файлы именуют системными файлами, и на их долю приходится около 62 килобайт. К системным файлам относятся исполняемые и звуковые файлы, а также демо-файлы и Тестовые программы.

Вы всегда имеете возможность восстановить состояние памяти устройства NXT до состояния по умолчанию, в котором она была организована на заводе-производителе (со всеми системными файлами), восстановив микропрограмму. См. ниже "Восстановление памяти устройства NXT...".

Чтобы удалить системные файлы, выберите окно "Отобразить системные файлы" (8). Затем выберите категорию файлов (2) и отметьте системные файлы, которые вы хотите удалить (3). Щелкните по "Удалить", удаляя из памяти каждый файл в отдельности.



Примечание: Удалив из памяти системные файлы, вы освободите память, но одновременно удалите функциональные свойства устройства NXT. Тем не менее, вы не "сломаете" устройство NXT, удалив системные файлы.



## Восстановление памяти устройства NXT до состояния по умолчанию (заводские настройки)

Вы можете восстановить память устройства NXT до заводских настроек по умолчанию, переустановив микропрограмму. Подробную информацию об этой процедуре см. в разделе справочника “Обновление микропрограммы NXT”. Необходимо помнить, что при переустановке микропрограммы NXT из памяти будут удалены все файлы -- перед началом не забудьте создать резервные копии личных файлов, которые вы хотите сохранить.

## Копирование и резервное копирование файлов, хранящихся в устройстве NXT

Чтобы сделать резервные копии ваших файлов или поделиться с друзьями теми файлами, которые хранятся в устройстве NXT, вам следует скопировать файлы с устройства NXT на ПК.

Для этого, во-первых, выберите категорию (2) для файлов, резервные копии которых вы намерены создать или которыми вы хотите поделиться с друзьями. Выберите необходимые файлы в списке (3) и щелкните по "Отправить" (7). Наконец, выберите папку на ПК, в которой будут храниться файлы, и нажмите ОК.

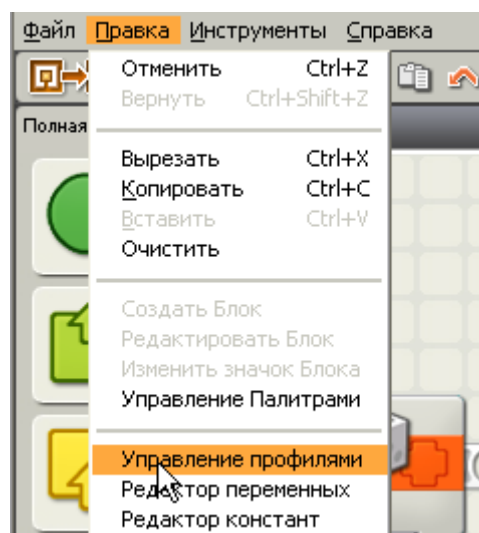
## Возврат файлов в устройство NXT

Чтобы вернуть файлы в устройство NXT, щелкните по кнопке "Выгрузить" (5). Выберите файл, который вы хотите скопировать с вашего ПК, и щелкните по ОК.

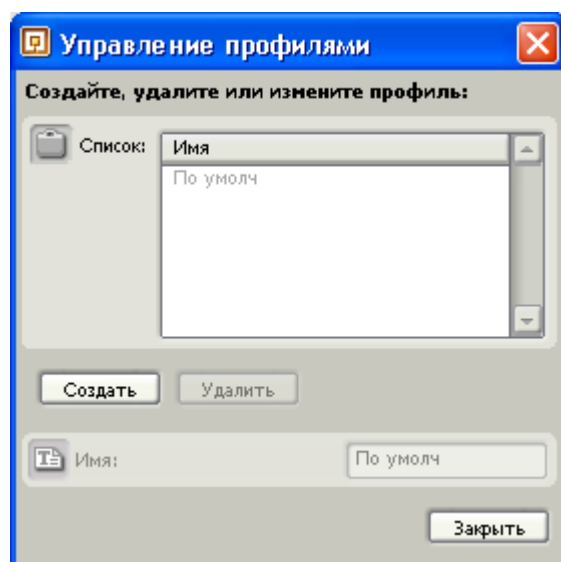
## Профили

Вы можете создать персональный профиль, в котором в одной папке будут сохранены все ваши программы и Мои Блоки. Это особенно удобно, если используется компьютер, на котором работают несколько пользователей. При использовании персонального профиля ваши программы и блоки не смешиваются с программами и блоками остальных пользователей.

## Создание профиля

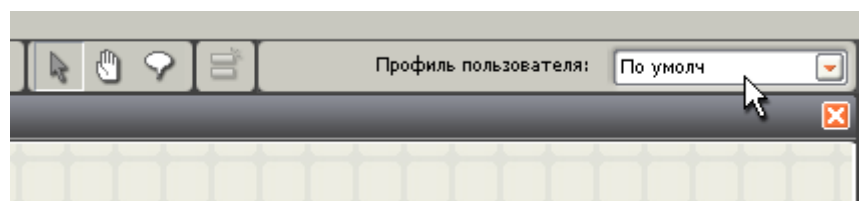


Чтобы создать профиль, в меню Правка выберите Управление профилями. Откроется окно Управление профилями.



Нажмите кнопку Создать и введите свое имя (или название профиля, под которым Вы хотите сохранить свои файлы). Нажмите кнопку Закрыть.

Теперь, каждый раз приступая к работе в среде программирования MINDSTORMS NXT, выберите свой профиль из открывающегося списка «Профиль пользователя» на Панели инструментов. При открытии или сохранении программы, выбранная папка профиля будет представлена как папка по умолчанию.



## Размещение файлов профиля и программных файлов

По умолчанию профили находятся в папке ..\My Documents\LEGO Creations\MINDSTORMS Projects\profiles\[название профиля].

В каждой папке профиля находятся программные файлы, сохраненные при использовании этого профиля. В каждой папке профиля имеется вложенная папка «Blocks» с пользовательскими блоками (включая Мои Блоки) для данного профиля.

В папке профиля также находится файл под названием Settings.ini. В этот файл записаны предпочтения для данного профиля. Он включает в себя подробную информацию об устройствах NXT, связанных с профилем, а также об использованных под данным профилем программах. В этой папке по умолчанию открываются/сохраняются файлы RBT. По умолчанию это папка профиля, но при необходимости её можно перенести. "Known bricks" - это список NXT, которые вы просматривали или с которыми устанавливали соединение (через Bluetooth).

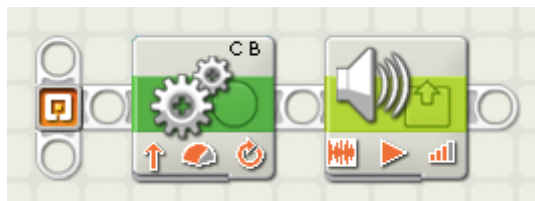
## Переименование или удаление профиля

Чтобы переименовать или удалить профиль, в меню Правка выберите пункт Управление профилями, затем выберите профиль, который нужно переименовать или удалить.

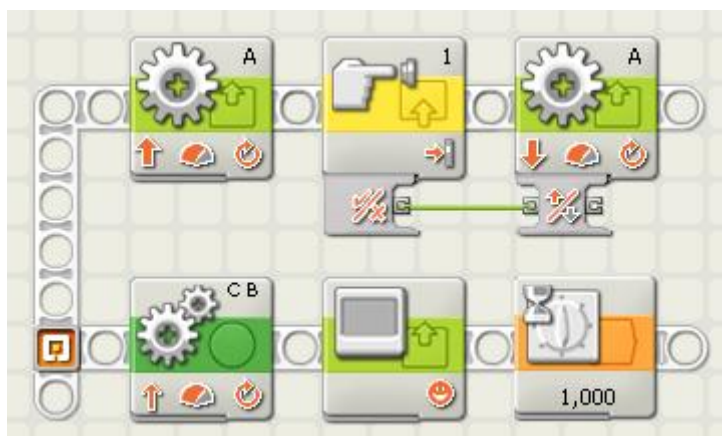
Чтобы удалить профиль, нажмите кнопку Удалить. Чтобы переименовать, введите новое имя в текстовое поле внизу окна.

## Цепочка программы

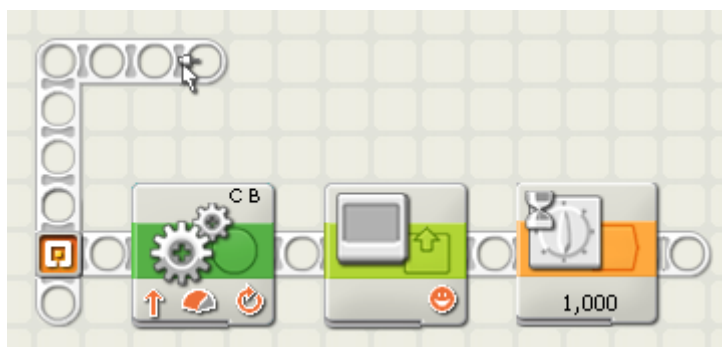
Программа выполняется в последовательности, заданной цепочкой программы. Блоки, которые встроились в цепочку, можно загрузить в NXT. Остальные блоки в рабочей области, которые не встроены в цепочку программы, загружаться не будут.



Из начальной точки можно выводить дополнительные цепочки программы, что позволит программе выполнять несколько задач одновременно. Например, в программе, управляющей движением робота вперед, можно создать основную цепочку и дополнительную, которая будет управлять рукой робота.



Создайте параллельную цепочку программы, поместив указатель мыши на начальную точку, нажмите и удерживайте кнопку мыши, перемещая указатель вверх или вниз. При загрузке и запуске программы, блоки в обеих цепочках будут работать параллельно. Блоки из обеих цепочек можно связать друг с другом шинами данных.

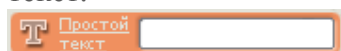


Также можно создать новую цепочку программы, не исходящую из начальной точки. Для этого установите указатель мыши на нужном участке основной цепочки и, нажав клавишу Shift переместите его вверх или вниз.



## Простые текстовые поля

Если текстовые поля выделены, как показано ниже, они поддерживают только простой текст.



Если текстовое поле не выделено, оно поддерживает полный текст, включая акцентированные символы и символы азиатских алфавитов.

Вам следует ограничить свой текст символами, перечисленными ниже, во избежание появления их в виде пробелов на дисплее устройства NXT.

### Что такое простой текст?

Дисплей блока NXT поддерживает отображение заглавных и прописных букв от А до Z и чисел от 0 до 9, а также знаков пунктуации и символов, показанных ниже.

```
!"#$%&'()*+,-./0123456789:;<=>?  
@ABCDEFGHIJKLMNPOQRSTUVWXYZ[\]^_  
*abcdefghijklmnopqrstuvwxyz{|}~`
```

Простой текст используется для названий блока, Моих Блоков, названий файлов (включая программы и файлы, созданные при помощи блока доступа к файлам и блока записи/воспроизведения) и отображает текст на устройстве NXT при помощи программного блока отображения. Если поля простого текста содержат другие символы, отличные от простых, они будут отображаться в виде пробелов. Устройство NXT сохраняет символы, и они будут правильно отображаться на подключенном компьютере, однако во избежание большого количества пробелов, возможно, Вы захотите использовать простые символы для выделенных текстовых полей.

Во всех таких случаях требуется использование "Простого текста". В таких случаях невозможно ввести другие символы. Исключениями являются названия программных файлов и текст в программном блоке отображения.

## Горячие клавиши

При работе в среде программирования NXT для регистрации данных можно использовать следующие горячие клавиши:

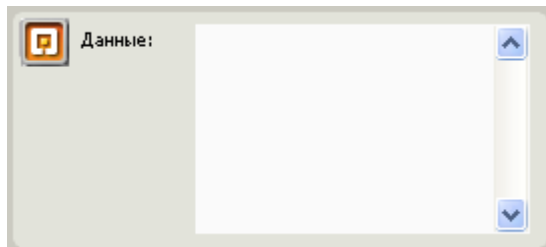
<b>Windows</b>	<b>Mac</b>	<b>Действие</b>
CTRL+A	Command-A	Новая аннотация
CTRL+B	Command-B	Остановить NXT
CTRL+C	Command-C	Копировать
CTRL+D	Command-D	Загрузить в NXT
CTRL+E	Command-E	Параметры эксперимента
CTRL+F	Command-F	Скриншот
CTRL+I	Command-I	Окно NXT
CTRL+N	Command-N	Новый эксперимент
CTRL+O	Command-O	Открыть...
CTRL+P	Command-P	Печать
CTRL+Q	Command-Q	Выход
CTRL+R	Command-R	Загрузить и запустить
CTRL+S	Command-S	Сохранить
CTRL+T	Command-T	Прогноз
CTRL+U	Command-U	Экспорт из NXT
CTRL+V	Command-V	Вставить
CTRL+W	Command-W	Заккрыть
CTRL+X	Command-X	Вырезать
CTRL+Z	Command+Z	Отменить
CTRL+Shift+P	Command-Shift-P	Анализ точки
CTRL+Shift+S	Command-Shift-S	Анализ участка
CTRL+Shift+Z	Command-Shift-Z	Вернуть
F1	F1	Справка
1	1	Полная палитра
2	2	Полная палитра
3	3	Моя палитра

## Начальная точка

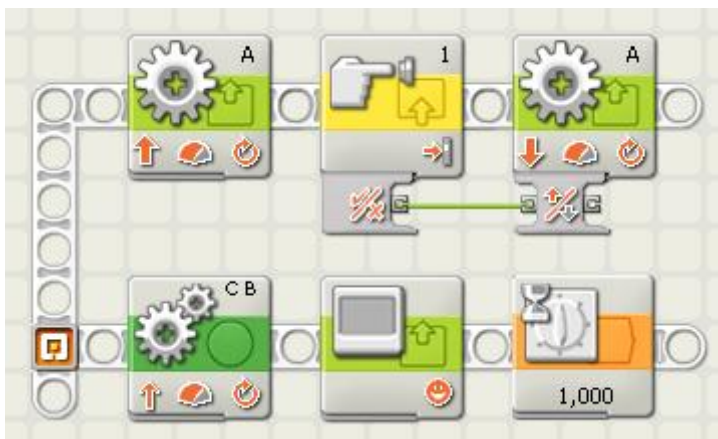
Это начальная точка для всех программ. Блоки, подключенные к начальной точке, будут загружены в NXT, если кнопками на пульте управления выбрать команды "Загрузить", "Загрузить и запустить" или "Загрузить и запустить выбранное".



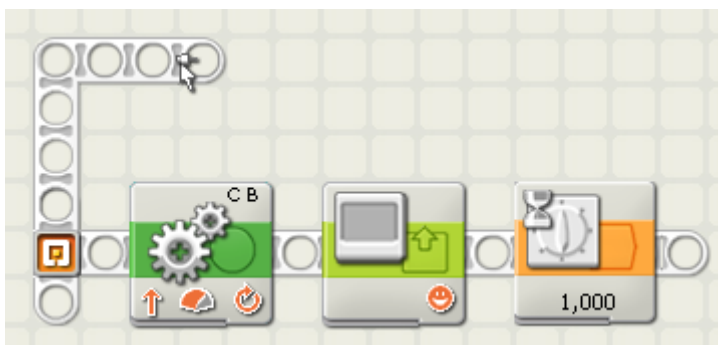
Если щёлкнуть на начальной точке, то в поле на панели настройки можно ввести примечание, которое будет сохранено вместе с программой.



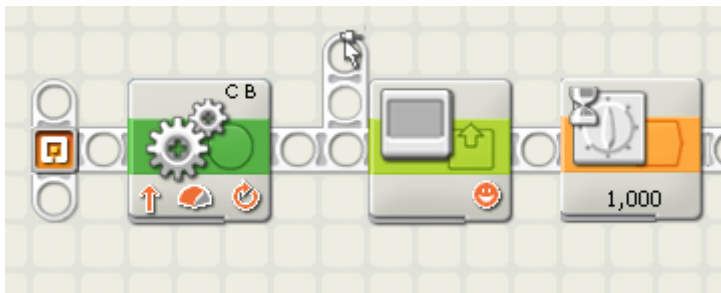
Из начальной точки можно выводить дополнительные цепочки программы, которые позволят программе выполнять параллельные задачи. Например, в программе, управляющей движением робота вперед, можно создать основную цепочку и дополнительную, которая будет управлять рукой робота.



Создайте параллельную цепочку программы, поместив указатель мыши на начальную точку, нажмите и удерживайте кнопку мыши, перемещая указатель вверх или вниз. При загрузке и запуске программы, блоки в обеих цепочках будут работать параллельно. Блоки из обеих цепочек можно связать друг с другом шинами данных.



Также можно создать новую цепочку программы, не исходящую из начальной точки. Для этого установите указатель мыши на нужном участке основной цепочки и, нажав клавишу "Shift", переместите его вверх или вниз.

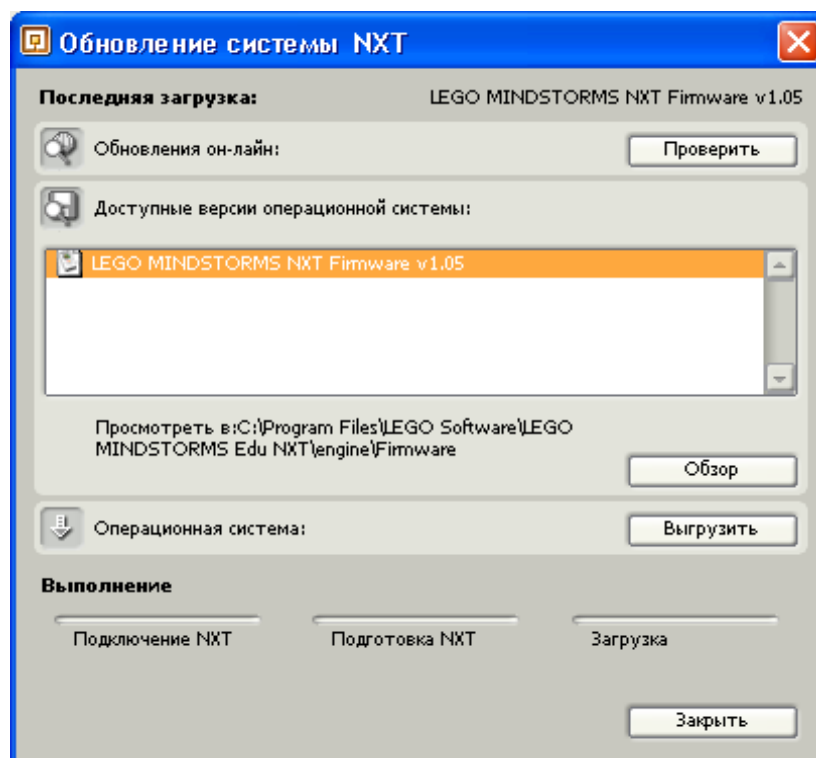


## Обновление операционной системы NXT

Система NXT – это операционная система микрокомпьютера NXT. Без неё NXT не сможет управлять моторами, воспринимать сигналы, поступающие от датчиков, да и вообще не сможет работать.

Компания LEGO Group постоянно выпускает обновлённые версии своих микропрограмм, расширяя их функциональные возможности, и устраняет обнаруженные ошибки в программном обеспечении. Обновляйте среду программирования своего NXT, чтобы получить доступ к новейшим технологиям и повысить производительность и надежность своего микрокомпьютера NXT.

Обновление системы не вызывает затруднений. В меню "Инструменты" выберите команду "Обновление системы NXT" ... . Откроется окно "Обновление системы NXT", показанное ниже.



Удостоверьтесь, что NXT включен и подсоединен к компьютеру.

Если компьютер находится в сети, можно проверить наличие обновлений, нажав кнопку "Проверить наличие обновлений". После этого будет произведен поиск наиболее поздней версии системы на сайте MINDSTORMS.com.



Выберите последнюю версию системы и укажите где сохранить обновление.  
Рекомендуются следующие адреса для сохранения:

Для Macintosh:

Macintosh HD:Applications:LEGO MINDSTORMS NXT:engine:Firmware

Для Windows:

C:\Program Files\LEGO Software\LEGO MINDSTORMS EDU NXT\engine\Firmware

После завершения поиска (он занимает 1 - 3 минуты, в зависимости от скорости соединения с Интернетом), обновленная версия ОС будет готова к установке. Если ОС сохранена по указанному выше адресу, можно просто выбрать файл из списка и нажать "Загрузить". Индикатор, расположенный в нижней части поля "Обновление ОС" будет отображать ход загрузки ОС в NXT.

Если файл системы сохранён в другом месте, щёлкните на кнопке "Обзор", чтобы найти папку, в которой файл сохранен. После этого можно загружать сохранённую ОС в NXT.

## Переустановка ОС, поставляемой с NXT

Если по какой-либо причине ваш NXT перестал функционировать, переустановка системы, поставляемой вместе с ним, может решить проблему.

Предварительно необходимо перезагрузить NXT. Дополнительную информацию по перезагрузке NXT можно найти в разделе “Устранение неполадок” печатного Руководства пользователя.

Перезагрузив NXT, следуйте данным ранее инструкциям, чтобы переустановить встроенную ОС NXT.



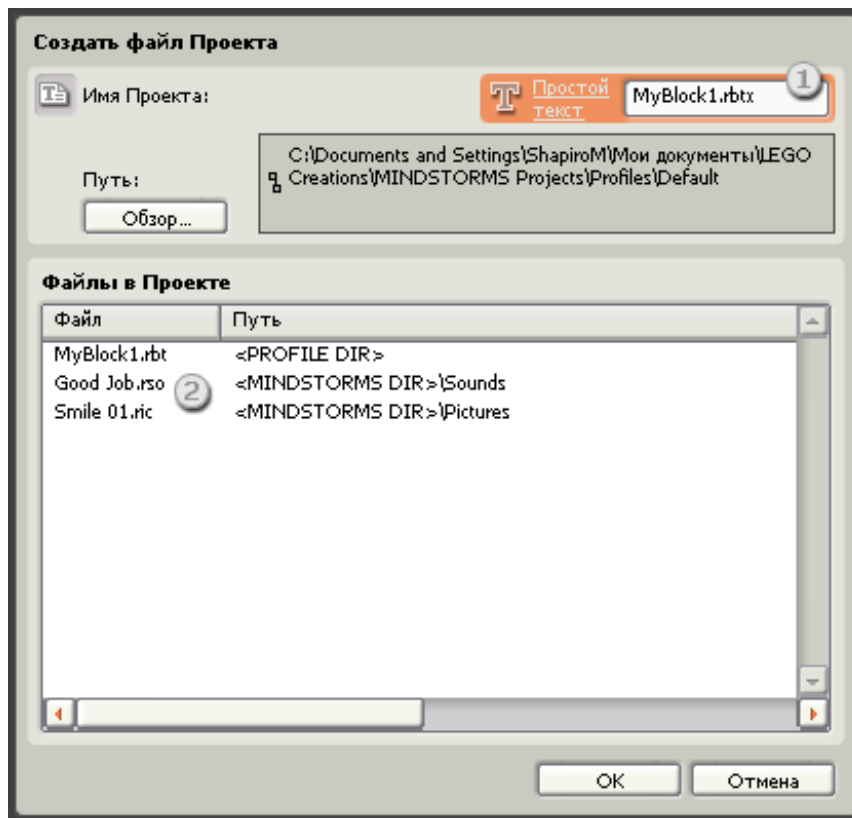
Примечание: перед переустановкой любых ОС проверьте аккумулятор NXT. Изношенный или разряженный аккумулятор может стать причиной неполадок.

## Создание Проекта

Проекты предназначены в первую очередь для передачи другим пользователям. Они включают созданные вами Блоки, звуки и картинки.

Чтобы создать Проект, выберите команду Создать файл Проекта в меню Инструменты.

## Параметры Проекта



1. Присвойте новому Проекту имя и укажите, в какой папке он будет сохранён на компьютере.
2. Список файлов, которые будут добавлены в файл Проекта
3. Щёлкните ОК, чтобы сохранить файл Проекта

#### Основная группа Блоков

- [Блок «Экран»](#)
- [Блок «Цикл»](#)
- [Блок «Движение»](#)
- [Блок «Писать/Играть»](#)
- [Блок «Звук»](#)
- [Блок «Переключатель»](#)
- [Блок «Ожидание»](#)

## Блок «Экран»

Этот блок используется для вывода изображений, набора текстового фрагмента или рисования на дисплее NXT.

Последовательно устанавливая несколько Блоков «Экран», можно создавать сложные изображения, дополняя их картинками, текстом и формами, усложняя их при помощи каждого дополнительного Блока «Экран».

Если перед началом работы вы хотите очистить экран, установите флажок "Очистить".

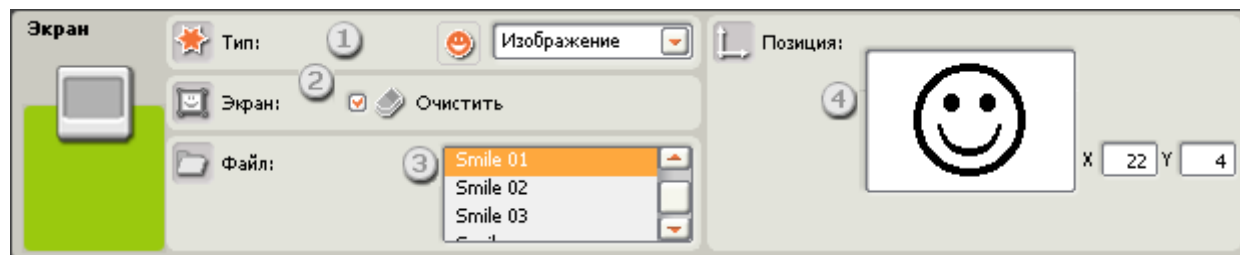
Чтобы восстановить пиктограмму LEGO MINDSTORMS, установленную по умолчанию, выберите в списке "Сброс".

Размер экрана - 100 пикселей в ширину и 64 пикселей в высоту.



1. Эта пиктограмма показывает назначение блока: настроен ли он на изображение, текст или рисунок, или просто возвращает на дисплей пиктограмму, установленную по умолчанию.
2. Значения можно изменять динамически, подключая шины данных к концентратору данных этого блока.

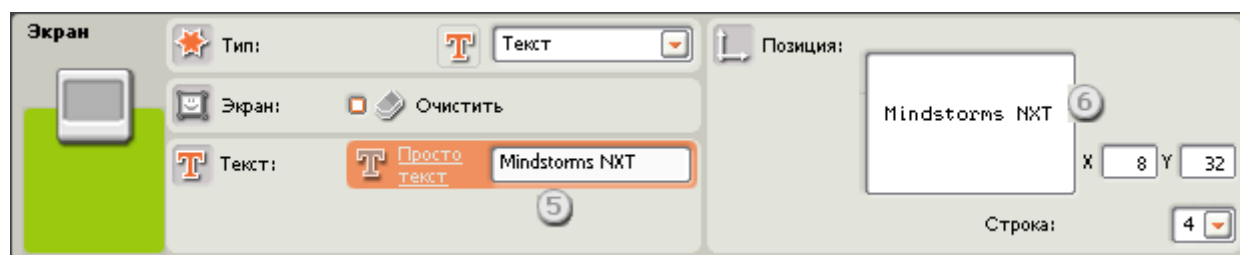
## Настройка Блока «Экран»:



1. В открывающемся списке можно выбрать: отображать картинку, текст или чертеж, или же просто сбросить настройки дисплея.
2. Чтобы очистить дисплей NXT, установите флажок "Очистить".

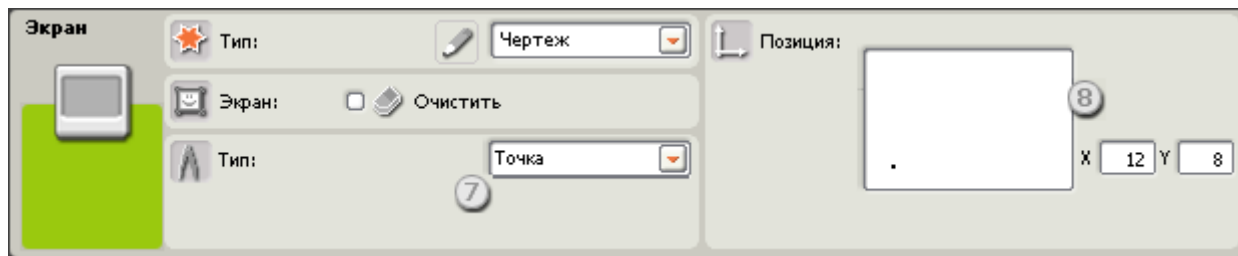
Если выбрано отображение картинки (как показано на панели настройки), можно задать следующие дополнительные функции:

3. Функция Файл позволяет выбрать картинку для отображения на дисплее. Щёлкните на названии файла, чтобы просмотреть хранящуюся в нем картинку.
4. При помощи мыши переместите картинку, текст или рисунок по экрану предварительного просмотра, чтобы определить, где они будут расположены. Поля ввода координат X и Y позволяют очень точно позиционировать изображение. Если в оба поля ввести значение "0", изображение будет помещено в левую нижнюю часть дисплея.



Если выбрано отображение текста (как показано на панели настройки), можно задать следующие дополнительные функции:

5. Функция Текст позволяет вводить текст для отображения на дисплее.
6. В поле "Строка" выберите номер строки, на которой следует разместить начало текста или используйте поля ввода координат X и Y, чтобы назначить точное место размещения начала текста. Если в оба поля ввести значение "0", текст будет помещен в левую нижнюю часть дисплея.



Если выбран Чертеж, то можно задать следующие дополнительные функции:

7. Выбрать инструмент черчения - точку, прямую или окружность.
8. При выборе точки, величины, введенные в поля координат X и Y определяют положение точки на экране. Если в оба поля ввести значение "0", точка будет помещена в левую нижнюю часть дисплея.

Если выбрать линию, появятся дополнительные поля ввода для задания крайних точек отрезка прямой. Начальная точка задаётся в верхних полях ввода координат x и y, а конечная точка - в нижних полях ввода координат x и y.

При выборе окружности, появится дополнительное поле для ввода радиуса окружности. Введите координаты центра окружности в первые два поля координат, а величину радиуса - в поле Радиус.

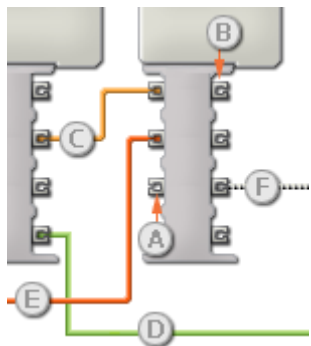
## Настройка концентратора данных Блока «Экран»

Блоком «Экран» можно управлять динамически, подключая шины данных (от концентраторов данных других блоков) к концентратору данных Блока «Экран».

Перетащив блок в рабочую область, откройте его концентратор данных, щёлкнув на кнопке в нижней левой части блока.



Шины данных, передающие входящие данные на блок, подключены к разъёмам, расположенным на левой стороне их концентраторов данных. Шины данных, передающие исходящие данные, подключены к разъёмам, расположенным справа.



- [A] Входной разъём
- [B] Выходной разъём
- [C] Числовая шина данных (желтая)
- [D] Логическая шина данных (зеленая)
- [E] Текстовая шина данных (оранжевая)
- [F] Повреждённая шина данных (серая)

### Передача данных от входного разъёма на выходной разъём

Если для входного разъёма имеется соответствующий выходной разъём (см. выше А), входящие данные будут проходить от входного разъёма на выходной разъём без изменений. В этом случае можно использовать выходной разъём, если входной разъём подключен к шине данных входа; подключение шины данных выхода к такому выходному разъёму без подключенной шины данных входа приведет к «повреждению» шины данных выхода (она станет серой).

## **По шинам данных передаются данные различных типов**

Через каждую шину данных осуществляется обмен данными различных типов между блоками. Например, если шина данных протянута с логического разъёма на концентратор данных блока, её можно подключить только к логическому разъёму на концентраторе данных другого блока. На приведенной ниже схеме показано, данные какого типа может принимать или передавать каждый из разъёмов.

## **Цвет шин данных**

Шины данных окрашены в различные цвета: шины для передачи числовых данных - желтые; шины, для передачи логических данных - зеленые; шины для передачи текстовых данных - оранжевые.

## **«Повреждённые» шины данных**

При попытке подключить шину данных к разъёму данных несоответствующего типа, шина будет повреждена (ее цвет изменится на серый). Вы не сможете загрузить программу, если шина данных повреждена.

Если щёлкнуть на повреждённой шине, то в небольшом окне контекстной справки в правом нижнем углу рабочей области появится информация о причине повреждения.

## **Данные должны соответствовать возможному диапазону, в котором работает разъём**

Если шина данных входа передает значение, не попадающее в допустимый диапазон разъёма, к которому она подключена, блок либо проигнорирует такие данные, либо приведёт их к значению, попадающему в этот диапазон. Если разъём допускает ограниченное количество значений (например, только 0, 1 или 2), он проигнорирует входящий сигнал, если передаваемое значение не будет вписываться в допустимый диапазон.

Если разъём допускает более широкий диапазон значений (например, 0 – 100), любой входящий сигнал, не попадающий в допустимый диапазон, будет приведен к значению, соответствующее ему. Например, если разъём мощности блока «Движение» получает входящий сигнал со значением 150, блок преобразует входящее значение в 100 (т.е., в числовое значение, лежащее в пределах диапазона разъёма мощности ).

## **Блок «Цикл»**

Этот блок используется для повтора последовательностей команд. Назначьте условие, необходимое для завершения цикла: истекшее время, количество повторов, логическая команда или сигнал от датчика. Также можно задать бесконечное повторение цикла.

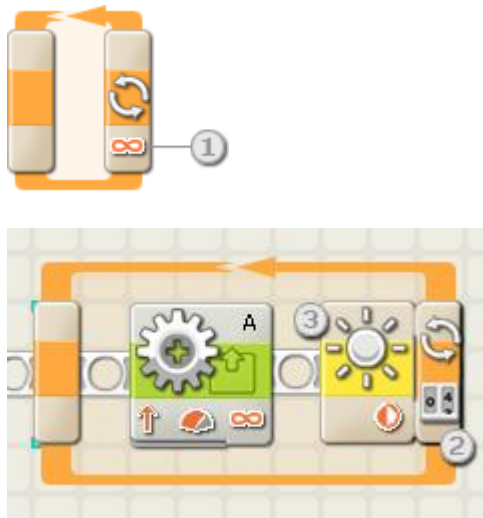
[Отображение параметров настройки блока](#)

[Добавление программных блоков в Блок «Цикл»](#)

[Перемещение Блока «Цикл»](#)

[Настройка Блока «Цикл»](#)

## Отображение параметров настройки блока



1. Если в секции Управления Блоком «Цикл» выбрано условие “Постоянно”, справа внизу блока отображается знак бесконечности ( $\infty$ ).
2. Если на панели настроек установлен флажок “Показать счётчик”, появится разъём, который позволит зацикливать программу на любом её участке на заданное количество завершённых циклов (если подключить разъём к концентратору данных другого блока через шину данных). Количество завершённых на данный момент циклов можно использовать для управления самим циклом. (См. ниже "Подсчёт" в разделе "Настройка Блока «Цикл»").
3. Если для управления циклом назначается датчик, Блок «Цикл» растягивается вправо, открывая пиктограмму выбранного датчика. Например, если для управления циклом назначен Датчик освещённости, в расширившейся части программного блока появится его пиктограмма. А в нижней части программного блока будет отображена соответствующая информация о выбранном способе управления.

## Добавление программных блоков в Блок «Цикл»

Если Блок «Цикл» вставлен в цепочку программы, её небольшой участок окажется внутри Блока «Цикл»; с ним будут связаны программные блоки, перетаскиваемые в верхнюю часть этой области цепочки. С каждым новым блоком, перетаскиваемым внутрь цикла, который уже содержит программные блоки, область цикла будет расширяться. За счёт этого создаётся пространство для новых блоков, которые будут встраиваться в цепочку программы.

## Перемещение Блока «Цикл»

Чтобы переместить Блок Цикл со всеми входящими в него Блоками, щёлкните именно на Блоке Цикл, чтобы его выделить, а затем перетащите на новое место.

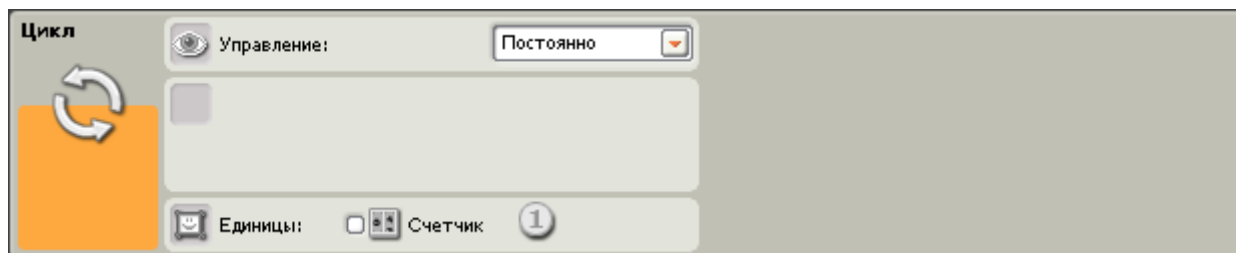
## Настройка Блока «Цикл»

В списке управления можно условие работы цикла:

- [Постоянно \[по умолчанию\]](#)
- [Время](#)
- Датчик (появляется новое поле со списком доступных датчиков)

- [Касания](#)
- [Освещённости](#)
- [Звука](#)
- [Расстояния](#)
- [Кнопка NXT](#)
- [Оборотов](#)
- [Таймер](#)
- [Принять сообщение](#)
- [Температуры](#)
- [Подсчёт](#)
- [Логика](#)

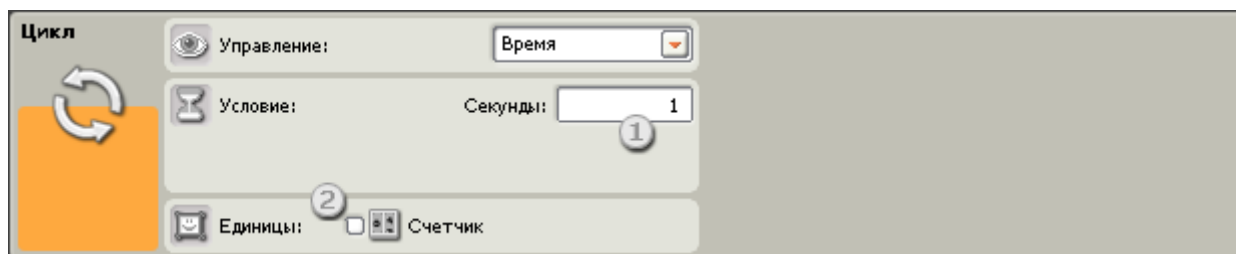
## Постоянно



Если выбрать “Постоянно,” действия всех программных блоков внутри цикла будут бесконечно и безостановочно повторяться.

1. Установите флажок “Счётчик”, чтобы использовать количество завершённых циклов в качестве входа другого блока. (Пример: увеличение мощности мотора пропорционально количеству циклов.)

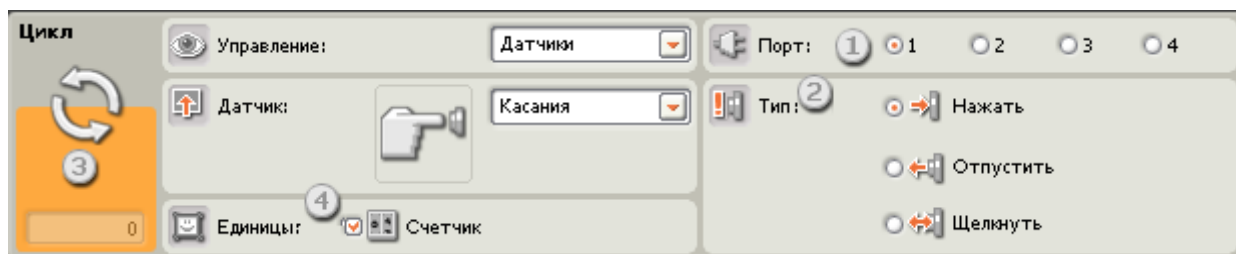
## Время



Если выбрать “Время,” программные блоки внутри цикла будут повторять свои действия до тех пор, пока не пройдет определенное количество секунд. По истечении заданного периода времени цикл завершится.

1. Введите в это окно то количество секунд, в течение которых цикл должен будет работать. Например, если выбрать 5 секунд, цикл завершится по истечении 5 секунд.
2. Если на панели настроек установлен флажок “Счётчик”, появится разъём, который позволит заикливать программу на любом её участке на заданное количество завершённых циклов (если подключить разъём к концентратору данных другого блока через шину данных).

## Датчик касания

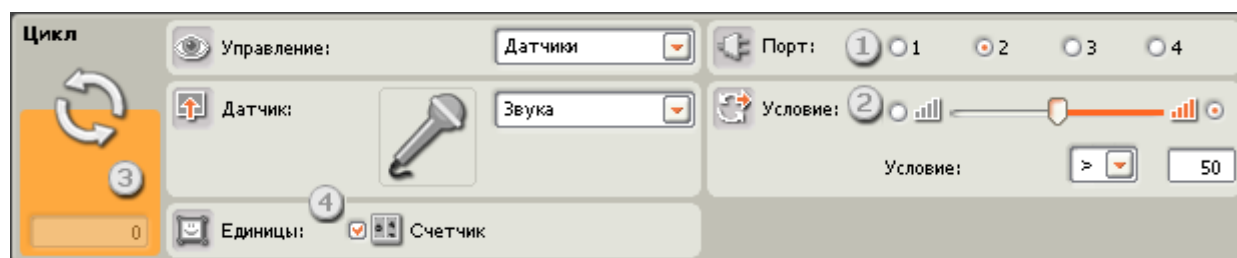




Если выбрать “Датчик касания,” программные блоки внутри цикла будут повторять свои действия до тех пор, пока датчик касания не зарегистрирует щелчок, нажатие или освобождение. После этого цикл завершит свою работу, а выполнение программы продолжится.

1. Выберите порт, к которому подключен датчик касания.
2. Кнопками выбора задайте щелчок, нажатие или освобождение датчика касания, в результате чего выполнение цикла должно прекратиться. Выберите "Щелчок", чтобы блок срабатывал при щелчке по кнопке датчика касания (длительность воздействия менее 0,5 с). Выберите "Нажать", чтобы блок срабатывал при нажатии на кнопку датчика касания. Выберите "Отпустить", чтобы блок срабатывал при освобождении кнопки датчика касания.
3. В поле обратной связи можно проверить состояние датчика касания. Если датчик робота активирован, в нём появится цифра “1”.
4. Если на панели настроек установлен флажок “Счётчик”, появится разъём, который позволит заикливать программу на любом её участке на заданное количество завершённых циклов (если подключить разъём к концентратору данных другого блока через шину данных).

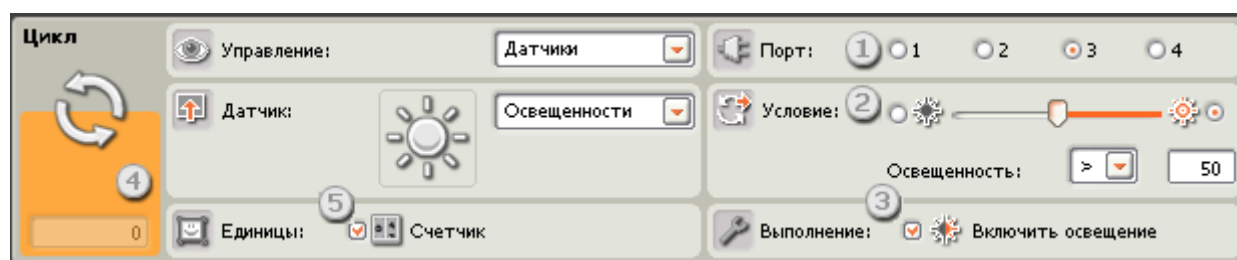
## Датчик звука



Если выбрать “Датчик звука,” программные блоки внутри цикла будут повторять свои действия до тех пор, пока датчик звука не зарегистрирует звук определенной громкости.

1. Выберите порт, к которому подключен датчик звука. По умолчанию для датчика звука блоку будет назначен порт 2.
2. Пороговое значение можно задать движком или ввести нужное значение в поле. Активируйте кнопку выбора справа от движка, чтобы, выполнение цикла прекращалось, когда громкость звука превысит пороговое значение; активируйте левую кнопку выбора, чтобы цикл завершался при громкости звука ниже порогового значения. Также можно воспользоваться открывающимся списком, чтобы установить движком предел, при котором выполнение цикла будет прекращаться.
3. В поле обратной связи отображаются текущие показания датчика звука (0-100%). Этим можно воспользоваться для опробования различных пороговых значений.
4. Если на панели настроек установлен флажок “Счётчик”, появится разъём, который позволит заикливать программу на любом её участке на заданное количество завершённых циклов (если подключить разъём к концентратору данных другого блока через шину данных).

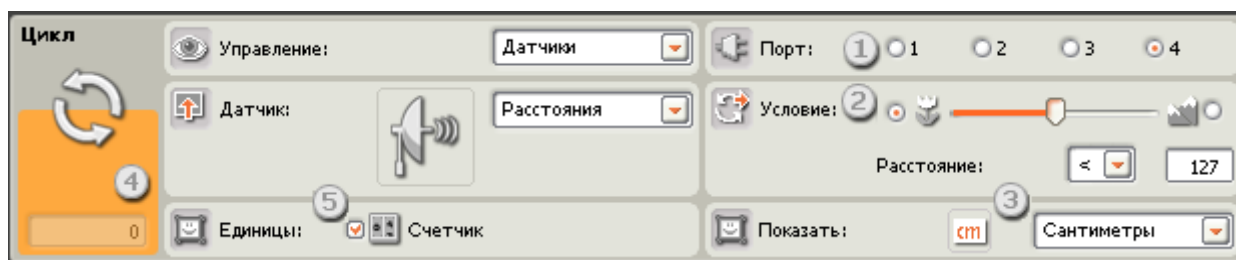
## Датчик освещённости



Если выбран “Датчик освещённости,” программные блоки внутри цикла будут повторять свои действия, пока датчик освещённости не зарегистрирует определенную яркость света. После этого выполнение цикла прекратится, а выполнение программы продолжится.

1. Выберите порт, к которому подключен датчик освещённости. По умолчанию него блоку будет назначен порт 3.
2. Пороговое значение можно задать движком или ввести нужное значение в поле. Активируйте кнопку выбора справа от движка, чтобы выполнение цикла прекращалось, когда уровень освещённости превышает пороговое значение; активируйте левую кнопку выбора, чтобы выполнение цикла прекращалось при уровнях освещённости более низких, чем пороговое значение. Также можно воспользоваться открывающимся списком, чтобы установить движком предел, при котором выполнение цикла будет прекращаться.
3. Если установлен флажок “Включить освещение”, датчик включит свой собственный слабый источник света и зарегистрирует отражённый свет.
4. В поле обратной связи будут отображаться текущие показания датчика.
5. Если на панели настроек установлен флажок “Счётчик”, появится разъём, который позволит заикливать программу на любом её участке на заданное количество завершённых циклов (если подключить разъём к концентратору данных другого блока через шину данных).

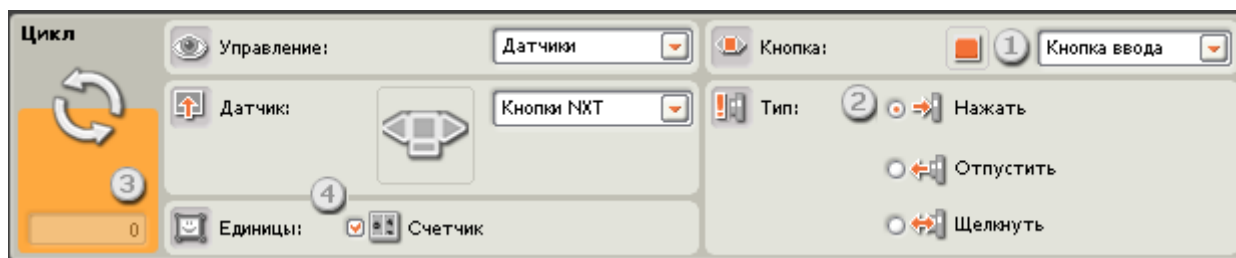
## Датчик расстояния



Если выбрать “Датчик расстояния,” программные блоки внутри цикла будут повторять свои действия до тех пор, пока этот датчик не обнаружит объект на заданном расстоянии.

1. Выберите порт, к которому подключен датчик расстояния. По умолчанию для него блоку будет назначен порт 4.
2. Пороговое значение можно задать движком или ввести нужное значение в поле. Активируйте кнопку выбора слева от движка, чтобы выполнение цикла прекращалось, когда датчик расстояния обнаруживает объект на расстоянии, меньшем порогового значения ; активируйте правую кнопку выбора, чтобы выполнение цикла прекращалось, когда датчик расстояния обнаруживает объект на расстоянии, превышающем пороговое значение. Также можно воспользоваться открывающимся списком, чтобы установить движком предел, при котором выполнение цикла будет прекращаться.
3. Выберите единицу измерения для регистрируемых значений - сантиметры или дюймы.
4. В поле обратной связи будут отображаться текущие показания датчика расстояния.
5. Если на панели настроек установлен флажок “Счётчик”, появится разъём, который позволит заикливать программу на любом её участке на заданное количество завершённых циклов (если подключить разъём к концентратору данных другого блока через шину данных).

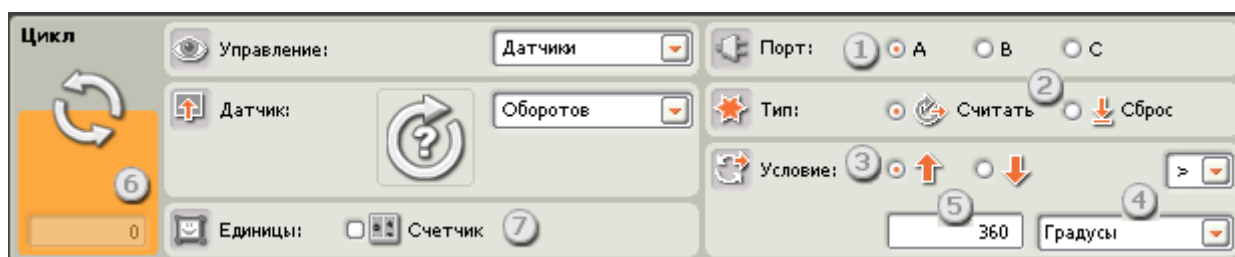
## Кнопки NXT



Если выбрать “Кнопки NXT,” работа программных блоков внутри цикла будет повторяться, пока с выбранными кнопками NXT не будет произведено какое-либо действие (щелчок, нажатие или освобождение).

1. Выберите кнопку NXT, воздействие на которую вызовет команду "истина" (и прекратит выполнение цикла).
2. Выберите "Щелкнуть" чтобы Блок срабатывал после щелчка по выбранной кнопке. Выберите "Нажать", чтобы Блок срабатывал срабатывал при нажатии кнопки. Выберите "Отпустить", чтобы Блок срабатывал, после отпускания кнопки.
3. В поле обратной связи появится "1", если с выбранной кнопкой NXT было произведено какое-либо действие.
4. Если на панели настроек установлен флажок "Счётчик", появится разъём, который позволит заикливать программу на любом её участке на заданное количество завершённых циклов (если подключить разъём к концентратору данных другого блока через шину данных).

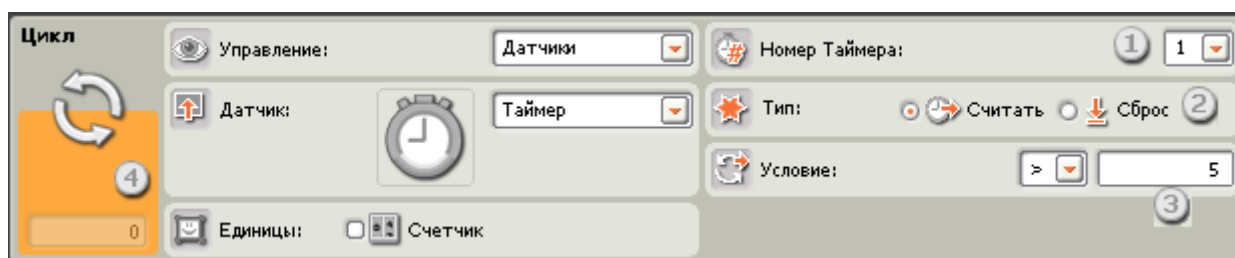
## Датчик оборотов



Если выбрать вариант "Считать", программные блоки внутри цикла будут повторять свои действия до тех пор, пока датчик оборотов не зафиксирует достижение определенного значения; затем выполнение цикла прекратится, а программа продолжит свою работу. Если выбрать вариант "Сброс", датчик будет сбрасываться на "ноль" после каждого цикла; блок будет продолжать выполнение цикла до тех пор, пока датчик оборотов не зафиксирует достижение порогового значения во время одного из циклов.

1. Выберите порт, к которому подключен мотор.
2. Выберите настройку для датчика оборотов - "Считать" или "Сброс".
3. Кнопками выбора задайте направление вращения: вперед или назад.
4. В открывающемся списке выберите "подсчёт оборотов" или "подсчёт градусов".
5. Введите количество оборотов или градусов, которые вы хотите наработать, прежде, чем прекратиться выполнение цикла (позволив программе продолжить работу).
6. Поле обратной связи будет отображать текущее количество совершённых оборотов или градусов поворота вала мотора.
7. Если на панели настроек установлен флажок "Счётчик", появится разъём, который позволит заикливать программу на любом её участке на заданное количество завершённых циклов (если подключить разъём к концентратору данных другого блока через шину данных).

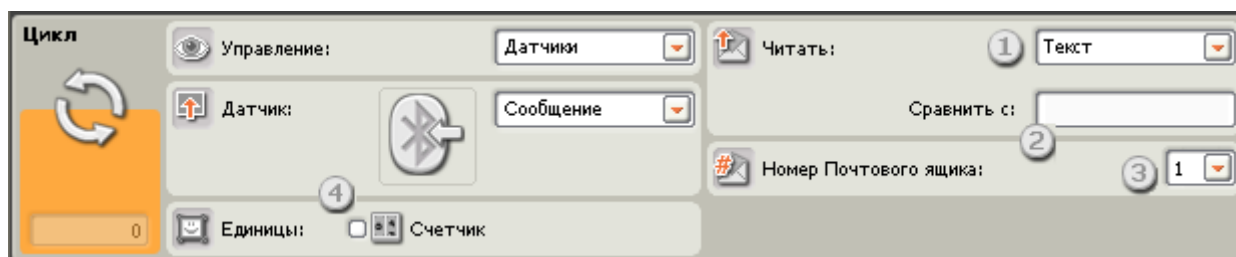
## Таймер



Если выбрать вариант "Считать", программные блоки внутри цикла будут повторять свои действия до тех пор, пока один из встроенных таймеров NXT не зарегистрирует окончание заданного промежутка времени; затем выполнение цикла прекратится, а выполнение программы будет продолжено. Если выбрать "Сброс", таймер будет обнуляться после завершения каждого цикла; блок будет продолжать выполнение цикла до тех пор, пока в каком-либо из циклов отсчёт таймера не достигнет своего порогового значения.

1. Выберите таймер NXT, за которым вы хотите наблюдать.
2. Выберите тип работы с таймером - "Считать" или "Сброс".
3. Задайте условие существования цикла: знак и время (в секундах).
4. Если на панели настроек установлен флажок “Счётчик”, появится разъём, который позволит заикливать программу на любом её участке на заданное количество завершённых циклов (если подключить разъём к концентратору данных другого блока через шину данных).

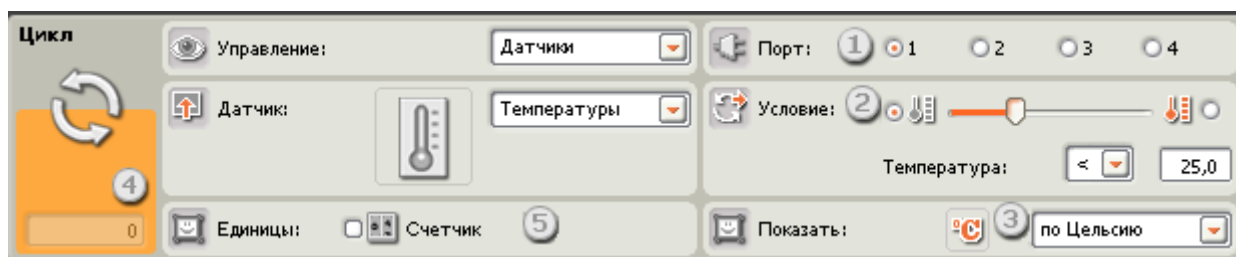
## Приём сообщений



Если выбрать “Приём сообщений”, программные блоки внутри цикла будут повторять свои действия до тех пор, пока NXT не получит определенное сообщение через Bluetooth. После этого выполнение цикла прекратится, а выполнение программы продолжится.

1. В открывающемся списке “Читает” выберите тип ожидаемого сообщения (текстовое, числовое или логическое).
2. Для сравнения входящего сообщения с тестовым, наберите контрольный текст или число (в зависимости от того, какой формат был выбран: текстовый или числовой), или кнопками выбора задайте тестовую логическую команду ("истина" или "ложь").
3. Выберите номер почтового ящика для приёма сообщения.
4. Если на панели настроек установлен флажок “Счётчик”, появится разъём, который позволит заикливать программу на любом её участке на заданное количество завершённых циклов (если подключить разъём к концентратору данных другого блока через шину данных).

## Датчик температуры

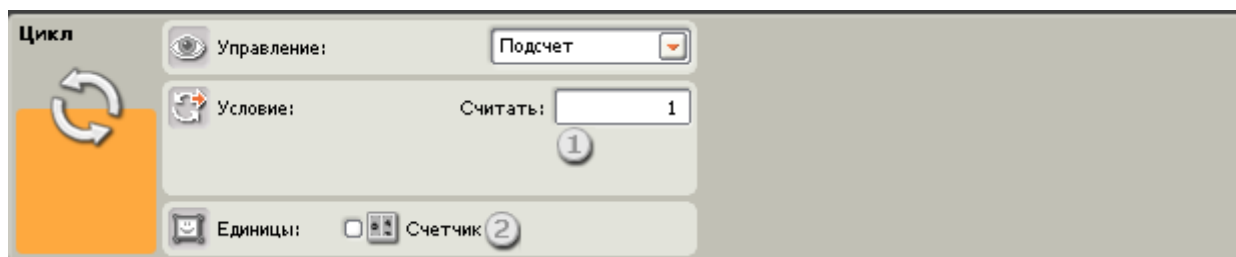


Если выбрать “Датчик температуры”, программные блоки внутри цикла будут повторять свои действия до тех пор, пока датчик температуры не зарегистрирует определенную температуру. После этого выполнение цикла прекратится, а выполнение программы продолжится.

1. Выберите порт, к которому подключен датчик температуры. По умолчанию для датчика температуры будет назначен порт 4.
2. Пороговое значение можно задать движком или ввести нужное значение в поле Температура. При помощи кнопок справа и слева от движка выберите условие остановки цикла (подачи команды "истина") – выше или ниже порогового значения. То же самое можно сделать используя список Температура.
3. Выберите шкалу измерения регистрируемых значений - Цельсия или Фаренгейта.
4. В поле обратной связи будут отображаться текущие показания датчика температуры.

5. Если на панели настроек установлен флажок “Счётчик”, появится разъём, который позволит заикливать программу на любом её участке на заданное количество завершённых циклов (если подключить разъём к концентратору данных другого блока через шину данных).

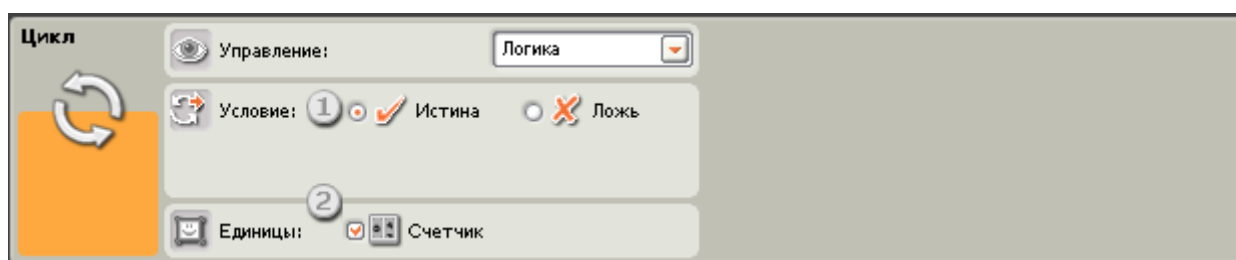
## Подсчёт



Если выбрать “Подсчёт,” программные блоки внутри цикла будут повторять свои действия, пока не будет выполнено заданное количество повторений.

1. Введите в это поле количество повторений, после выполнения которых цикл завершится. Например, если ввести “2”, цикл завершится после того, как программные блоки сработают дважды.
2. Установите флажок “Счётчик”, это даёт возможность соединить правую сторону цикла с левой (где появился разъём “счётчик”). После того как соединение установлено, программа сможет подсчитывать количество завершённых циклов и после выполнения заданного количества циклов остановит их.

## Логика



Если выбрать условие “Логика” и подключить шину входа данных к правой стороне Блока «Цикл», цикл будет повторяться, пока Блок «Цикл» не получит через шину данных логическую команду “истина” или “ложь”. После получения такой логической команды, выполнение цикла будет прекращено.

1. Выберите логическую команду, по которой выполнение цикл будет прекращено - “истина” или “ложь”.
2. Если на панели настроек установлен флажок “Счётчик”, появится разъём, который позволит заикливать программу на любом её участке на заданное количество завершённых циклов (если подключить разъём к концентратору данных другого блока через шину данных).

## Блок «Движение»

Этот блок предназначен для задания роботу прямолинейного движения вперед или назад, или для выполнения поворотов при движении по криволинейной траектории. Задайте дальность перемещения робота, используя параметр “Время”.



1. Буквы в правом верхнем углу блока показывают, какие порты NXT будут контролироваться.
2. Эта пиктограмма показывает направление движения робота.
3. Эта пиктограмма показывает уровень мощности. На скорость робота также влияют различные условия, например, состояние поверхности, по которой он движется, подъём или спуск.
4. Эта пиктограмма показывает, как задан параметр «Отработать»: Без ограничения, в градусах, оборотах или секундах.



1. Эта пиктограмма показывает, что в параметрах «Направление» задана команда «Стоп». При этой установке все моторы выключаются.

## Настройка Блока «Движение»



1. Выберите моторы, которыми нужно управлять. При управлении двумя моторами (например, В и С), они синхронизируются и вращаются вперед или назад при одинаковой мощности. Если нужно управлять тремя моторами, то синхронизируются моторы В и С.
2. Выберите направление вращения мотора: вперед или назад, или его остановку. При выборе команды «Стоп» настройки моторов также сбрасываются.
3. Если для управления роботом используются два мотора (по одному с каждой стороны), появляется движок управления, буквы на каждой из его сторон соответствуют выбранным портам. Для задания движения робота по кривой переместите движок. Если его сдвинуть в одно из крайних положений, робот будет поворачиваться на месте.
4. Этим движком можно задать уровень мощности [0-100 %], или ввести значение в поле ввода.
5. В списке «Время» можно выбрать работу мотора в течение неограниченного времени, определённого количества секунд, оборотов [настройка по умолчанию] или градусов поворота вала. Указывая значение параметра «Секунды», «Обороты» или «Градусы», вы однозначно определяете длину перемещения робота. (См. совет ниже.)
6. Задайте действие моторов после выключения: «Тормозить» или «Катиться». Если робот должен остановиться точно в заданном месте, выберите «Тормозить». Это также предотвратит скатывание робота по наклонной поверхности, однако при такой настройке увеличивается расход заряда аккумулятора NXT, поскольку моторы работают, удерживая робота на месте.
7. В поле обратной связи отображается количество градусов поворота вала мотора или количество полных оборотов, совершённых им. Нажмите кнопку сброса чтобы обнулить показания. (Эту информацию можно получать, если мотор подключен к выбранному порту и установлена связь с NXT.)



Совет: как задать перемещение робота на определённое расстояние?

Воспользуйтесь для этого полями обратной связи (7).

### Для колёсных или гусеничных моделей

Сначала нажмите кнопку сброса показаний в поле обратной связи. Подключите кабель USB или активируйте соединение Bluetooth, установите робота в стартовую позицию и



подтолкните его рукой вперед. Можно заметить, что в поле обратной связи отображается число оборотов или градусов поворота вала мотора. (Предварительно нужно задать параметр «Время» в «Оборотах» или «Градусах».) Остановите робота в нужном месте и запишите число совершённых оборотов или количество градусов. Если теперь ввести это значение в поле параметра «Время», загрузить и запустить программу, то робот переместится точно на заданное расстояние.

### Для роботов с ногами

В этом случае придётся подбирать нужное число оборотов или градусов методом проб и ошибок, каждый раз проверяя, насколько робот приблизился к заданной точке. Не забудьте предварительно сбросить показания.

### Где ещё можно использовать параметр «Время»

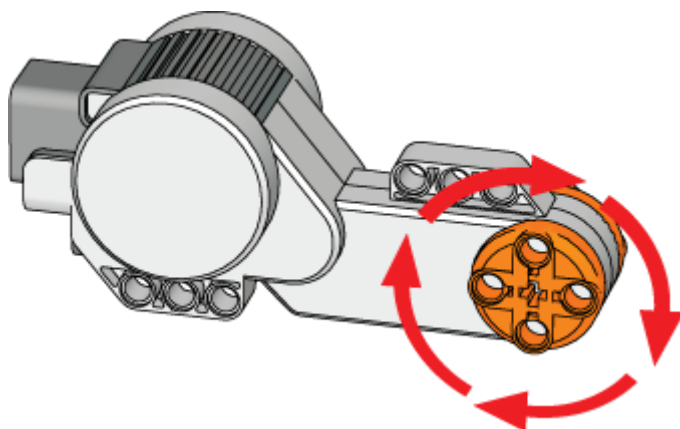
При программировании действий робота, оснащённого каким-либо приспособлением, например захватом, рекомендуется задавать «Время» в секундах. Тогда не потребуется измерять точное количество градусов, необходимое для управления приспособлением, и программа будет выполнена.



Примечание: если задать для продолжительности работы моторов любое значение, кроме «Ограничения нет», дальнейшее выполнение программы продолжится только по прошествии заданного времени.

## Вращение вала мотора в сторону, противоположную направлению движения робота

При установке направления Блока «Движение» или Блока «Мотор», задается направление вращения мотора. При этом оно может отличаться от фактического направления движения робота. В зависимости от конструкции робота, он может двигаться назад при вращении мотора вперед. На рисунке показано вращение мотора вперед (по умолчанию).



## Настройка концентратора данных Блока «Движение»

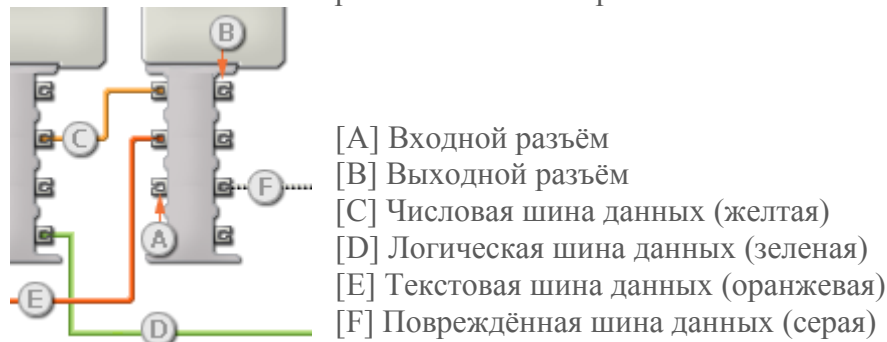
Блоком «Движение» можно управлять динамически, подключая к его концентратору данных шины данных (от концентраторов данных других блоков).

Перетащив блок в рабочую область, откройте его концентратор данных, щёлкнув на кнопке в нижней левой части блока.





Шины данных, передающие входящие данные на блок, подключены к разъёмам, расположенным на левой стороне их концентраторов данных. Шины данных, передающие исходящие данные, подключены к разъёмам, расположенным справа.



### Передача данных от входного разъёма на выходной разъём

Если для входного разъёма имеется соответствующий выходной разъём (см. выше A), входящие данные будут проходить от входного разъёма на выходной разъём без изменений. В этом случае можно использовать выходной разъём, если входной разъём подключен к шине данных входа; подключение шины данных выхода к такому выходному разъёму без подключенной шины данных входа приведет к «повреждению» шины данных выхода (она станет серой).

### По шинам данных передаются данные различных типов

Через каждую шину данных осуществляется обмен данными различных типов между блоками. Например, если шина данных протянута с логического разъёма на концентратор данных блока, её можно подключить только к логическому разъёму на концентраторе данных другого блока. На приведенной ниже схеме показано, данные какого типа может принимать или передавать каждый из разъёмов.

### Цвет шин данных

Шины данных окрашены в различные цвета: шины для передачи числовых данных - желтые; шины, для передачи логических данных - зеленые; шины для передачи текстовых данных - оранжевые.

### «Повреждённые» шины данных

При попытке подключить шину данных к разъёму данных несоответствующего типа, шина будет повреждена (ее цвет изменится на серый). Вы не сможете загрузить программу, если шина данных повреждена.

Если щёлкнуть на повреждённой шине, то в небольшом окне контекстной справки в правом нижнем углу рабочей области появится информация о причине повреждения.

### Данные должны соответствовать возможному диапазону, в котором работает разъём

Если шина данных входа передает значение, не попадающее в допустимый диапазон разъёма, к которому она подключена, блок либо проигнорирует такие данные, либо

приведёт их к значению, попадающему в этот диапазон. Если разбег допускает ограниченное количество значений (например, только 0, 1 или 2), он проигнорирует входящий сигнал, если передаваемое значение не будет вписываться в допустимый диапазон.

Если разбег допускает более широкий диапазон значений (например, 0 – 100), любой входящий сигнал, не попадающий в допустимый диапазон, будет приведен к значению, соответствующее ему. Например, если разбег мощности блока «Движение» получает входящий сигнал со значением 150, блок преобразует входящее значение в 100 (т.е., в числовое значение, лежащее в пределах диапазона разбёга мощности ).

## Блок «Писать/Играть»

Этот блок позволяет записывать параметры действия, выполненного роботом. Блок «Писать/Играть» «запоминает» количество оборотов, сделанных моторами робота, и затем воспроизводит это количество оборотов при выборе команды “Пуск” на Блоке «Писать/Играть» и запуске программы.

Например, Блок «Писать/Играть» может производить запись, когда вы катите колёсного робота вперед, а затем резко поворачиваете его влево. Также можно собрать роботизированную руку и записать движения при рукопожатии. Теперь при помощи Блока «Писать/Играть» можно воспроизвести эти записанные движения: робот будет катиться вперед и повернет в нужном месте, а роботизированная рука пожмет вам руку, и все это произойдет без вашей помощи.

Для записи какого-либо действия сначала нужно дать ему название и приблизительно оценить время до его завершения. В первом примере с катящимся роботом файл можно было бы назвать “вперед и поворот” и установить продолжительность действия 10 секунд.

После этого нужно загрузить блок. Когда всё будет сделано, запустите программу и произведите действие, которое нужно записать. По истечении заданного времени (в нашем примере это 10 секунд), действие, которое вы производили, будет сохранено.

И, наконец, измените режим действия блока с “запись” на “воспроизведение”, введите название, которое было присвоено записанной последовательности движений, еще раз загрузите Блок «Писать/Играть» и запустите программу. Она самостоятельно повторит (максимально точно, насколько это возможно) это действие.

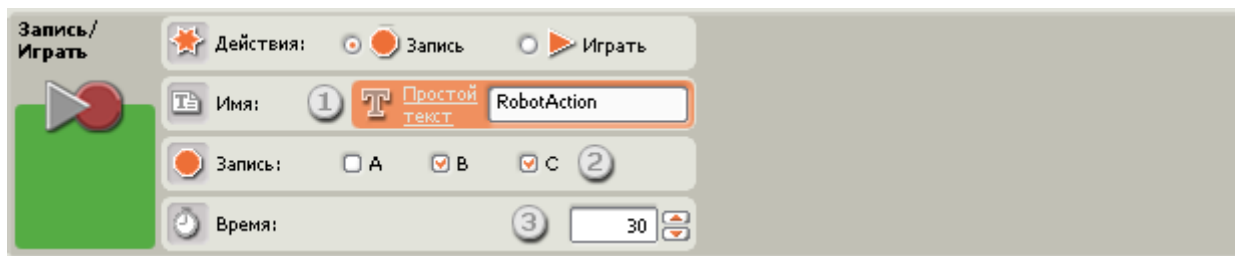


Примечание: воспроизведенное действие может немного отличаться от оригинального.



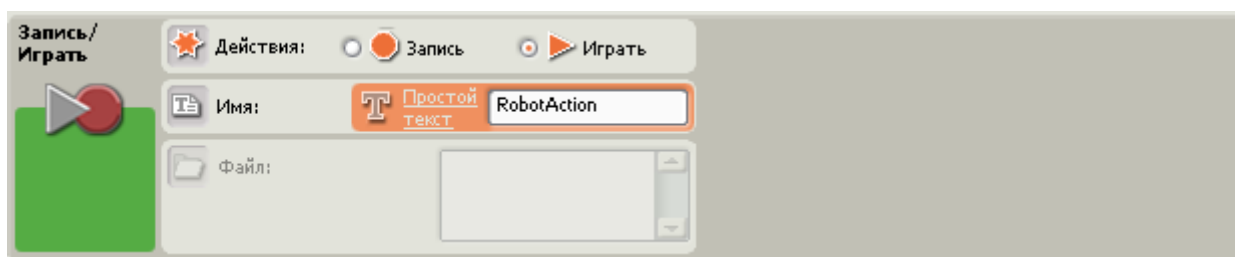
1. Эта пиктограмма показывает текущий режим работы Блока «Писать/Играть».

## Настройка Блока «Писать/Играть»:



При записи файла Писать/Играть, панель настройки будет выглядеть, как показано выше.

1. Ведите в текстовое поле название действия, которое вы собираетесь выполнить.
2. Если моторы подсоединены к портам В и С, отметьте поля В и С.
3. Установите продолжительность записи в секундах.



При воспроизведении уже записанного файла Писать/Играть, панель настройки будет выглядеть, как показано выше.

1. Введите название файла записанного действия, которое нужно воспроизвести. Уже сохранённые в NXT файлы будут представлены списком в алфавитном порядке. Выберите файл, который требуется воспроизвести.



Примечание: при записи и воспроизведении действия в одной и той же программе, необходимо ввести одно и то же имя файла в обоих блоках – название этого файла не появится в списке предварительно сохранённых действий.

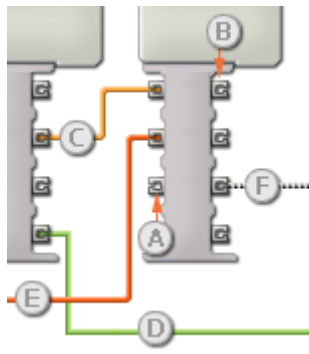
## Настройка концентратора данных Блока «Писать/Играть»

Блоком «Писать/Играть» можно управлять динамически, подключая к его концентратору данных шины данных (от концентраторов данных других блоков).

Перетаскив блок в рабочую область, откройте его концентратор данных, щёлкнув на кнопке в нижней левой части блока.



Шины данных, передающие входящие данные на блок, подключены к разъёмам, расположенным на левой стороне их концентраторов данных. Шины данных, передающие исходящие данные, подключены к разъёмам, расположенным справа.



- [A] Входной разъём
- [B] Выходной разъём
- [C] Числовая шина данных (желтая)
- [D] Логическая шина данных (зеленая)
- [E] Текстовая шина данных (оранжевая)
- [F] Повреждённая шина данных (серая)

## Передача данных от входного разъёма на выходной разъём

Если для входного разъёма имеется соответствующий выходной разъём (см. выше А), входящие данные будут проходить от входного разъёма на выходной разъём без изменений. В этом случае можно использовать выходной разъём, если входной разъём подключен к шине данных входа; подключение шины данных выхода к такому выходному разъёму без подключенной шины данных входа приведет к «повреждению» шины данных выхода (она станет серой).

## По шинам данных передаются данные различных типов

Через каждую шину данных осуществляется обмен данными различных типов между блоками. Например, если шина данных протянута с логического разъёма на концентратор данных блока, её можно подключить только к логическому разъёму на концентраторе данных другого блока. На приведенной ниже схеме показано, данные какого типа может принимать или передавать каждый из разъёмов.

## Цвет шин данных

Шины данных окрашены в различные цвета: шины для передачи числовых данных - желтые; шины, для передачи логических данных - зеленые; шины для передачи текстовых данных - оранжевые

При попытке подключить шину данных к разъёму данных несоответствующего типа, шина будет повреждена (ее цвет изменится на серый). Вы не сможете загрузить программу, если шина данных повреждена.

Если щёлкнуть на повреждённой шине, то в небольшом окне контекстной справки в правом нижнем углу рабочей области появится информация о причине повреждения.

## Данные должны соответствовать возможному диапазону, в котором работает разъём

Если шина данных входа передает значение, не попадающее в допустимый диапазон разъёма, к которому она подключена, блок либо проигнорирует такие данные, либо приведёт их к значению, попадающему в этот диапазон. Если разъём допускает ограниченное количество значений (например, только 0, 1 или 2), он проигнорирует входящий сигнал, если передаваемое значение не будет вписываться в допустимый диапазон.

Если разъем допускает более широкий диапазон значений (например, 0 – 100), любой входящий сигнал, не попадающий в допустимый диапазон, будет приведен к значению, соответствующее ему. Например, если разъем мощности блока «Движение» получает входящий сигнал со значением 150, блок преобразует входящее значение в 100 (т.е., в числовое значение, лежащее в пределах диапазона разьема мощности ).

## Блок «Звук»

Этот Блок можно использовать для воспроизведения звукового файла или какого-либо одиночного звука. Чтобы составить мелодию из отдельных звуков, выстройте в ряд несколько Блоков "Звук", каждый из которых настроен для воспроизведения различных звуков (нот).

Если на панели настройки установить флажок "До завершения", переход к следующему Блоку программы совершится только после того, как закончится воспроизведение звукового файла или отдельного звука. Если этот флажок не установлен, то воспроизведение звукового файла или отдельного звука может продолжаться и во время исполнения следующего Блока программы.

Если установить флажок "Повторять" звук будет воспроизводиться непрерывно.



1. Эта пиктограмма показывает настройку Блока: воспроизведение звукового файла или отдельного звука.
2. Эта пиктограмма показывает настройку Блока: включение или прекращение воспроизведения звука.
3. Эта пиктограмма показывает уровень громкости. Четыре оранжевые полосы на пиктограмме означают максимальную громкость.
4. Если протянуть шины данных от других Блоков на концентратор данных Блока «Звук», это повлияет на его свойства.

## Настройка Блока «Звук»



1. Укажите типа Блока, то есть, что должно воспроизводиться - звуковой файл или отдельный звук.
2. В секции Управление выберите действие Блока: начать или остановить воспроизведение звука.

3. Установите уровень громкости при помощи движка, или вписав значение в поле ввода.

Специальные параметры Блока "Файл":

4. Установите флажок "Повторять", чтобы звуковой файл проигрывался безостановочно. При этом поле "До завершения" станет недоступным.
5. В списке "Файл" выберите файл по названию мелодии. Чтобы услышать звук файла сделайте щелчок на его имени.
6. Если установлен флажок "Подождать завершения", звуковой файл будет воспроизведен полностью, прежде чем программа перейдет к следующему Блоку. В противном случае воспроизведение звукового файла может продолжаться и при исполнении следующего программного Блока. Если следующий Блок окажется тоже звуковым, воспроизведение первого звукового файла автоматически прекратится и начнется воспроизведение нового.

Специальные параметры Блока "Звук":

4. В секции "Нота" при помощи музыкальной клавиатуры укажите ноту, которую будет воспроизводить Блок «Звук». На клавиатуре умещаются три октавы. В полях над клавиатурой отображаются названия ноты и продолжительность ее звучания в секундах.
5. Если установлен флажок "До завершения", звук будет воспроизводиться до конца, прежде чем программе будет разрешено перейти к исполнению следующего Блока. В противном случае воспроизведение звука может продолжаться и при исполнении следующего Блока.

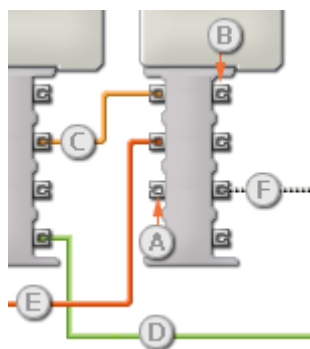
## Настройка концентратора данных Блока «Звук»

Блоком «Звук» можно управлять динамически, подключая к его концентратору данных шины данных (от концентраторов данных других Блоков).

Перетаскив Блок в рабочую область, откройте его концентратор данных, щёлкнув на кнопке в нижней левой части Блока.



Шины данных, передающие входящие данные на Блок, подключены к разъёмам, расположенным на левой стороне их концентраторов данных. Шины данных, передающие исходящие данные, подключены к разъёмам, расположенным справа.



- [A] Входной разъём
- [B] Выходной разъём
- [C] Числовая шина данных (желтая)
- [D] Логическая шина данных (зеленая)
- [E] Текстовая шина данных (оранжевая)
- [F] Повреждённая шина данных (серая)

## **Передача данных от входного разъёма на выходной разъём**

Если для входного разъёма имеется соответствующий выходной разъём (см. выше А), входящие данные будут проходить от входного разъёма на выходной разъём без изменений. В этом случае можно использовать выходной разъём, если входной разъём подключен к шине данных входа; подключение шины данных выхода к такому выходному разъёму без подключенной шины данных входа приведет к «повреждению» шины данных выхода (она станет серой).

## **По шинам данных передаются данные различных типов**

Через каждую шину данных осуществляется обмен данными различных типов между Блоками. Например, если шина данных протянута с логического разъёма на концентратор данных Блока, её можно подключить только к логическому разъёму на концентраторе данных другого Блока. На приведенной ниже схеме показано, данные какого типа может принимать или передавать каждый из разъёмов.

## **Цвет шин данных**

Шины данных окрашены в различные цвета: шины для передачи числовых данных - желтые; шины, для передачи логических данных - зеленые; шины для передачи текстовых данных - оранжевые.

## **«Повреждённые» шины данных**

При попытке подключить шину данных к разъёму данных несоответствующего типа, шина будет повреждена (ее цвет изменится на серый). Вы не сможете загрузить программу, если шина данных повреждена.

Если щёлкнуть на повреждённой шине, то в небольшом окне контекстной справки в правом нижнем углу рабочей области появится информация о причине повреждения.

## **Данные должны соответствовать возможному диапазону, в котором работает разъём**

Если шина данных входа передает значение, не попадающее в допустимый диапазон разъёма, к которому она подключена, Блок либо проигнорирует такие данные, либо приведёт их к значению, попадающему в этот диапазон. Если разъём допускает ограниченное количество значений (например, только 0, 1 или 2), он проигнорирует входящий сигнал, если передаваемое значение не будет вписываться в допустимый диапазон.

Если разъём допускает более широкий диапазон значений (например, 0 – 100), любой входящий сигнал, не попадающий в допустимый диапазон, будет приведен к значению, соответствующее ему. Например, если разъём мощности Блока «Движение» получает входящий сигнал со значением 150, Блок преобразует входящее значение в 100 (т.е., в числовое значение, лежащее в пределах диапазона разъёма мощности ).

## **Блок «Переключатель»**

Этот Блок используют для выбора одной из двух последовательностей команд. Например, если Блок «Переключатель» сконфигурирован под датчик касания, то он должен запустить одну последовательность Блоков в случае, если датчик нажат, и другую последовательность - если датчик не нажат.

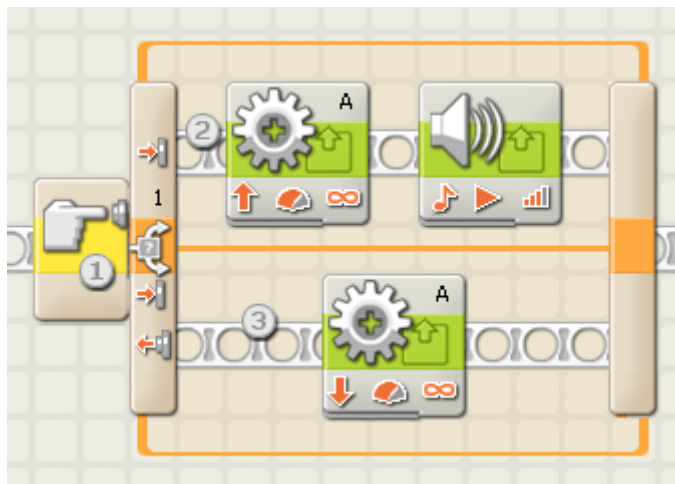
[Отображение параметров настройки Блока](#)

[Добавление Блоков в Блок «Переключатель»](#)

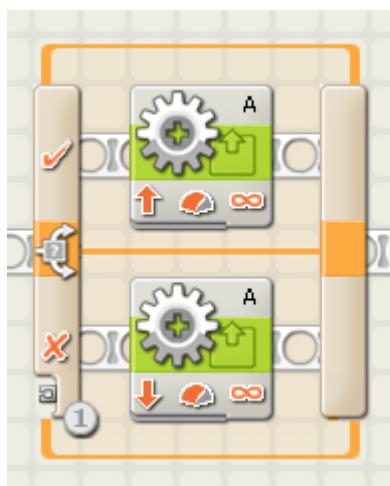
[Перемещение Блока «Переключатель»](#)

[Настройка Блока «Переключатель»](#)

## Отображение параметров настройки Блока :

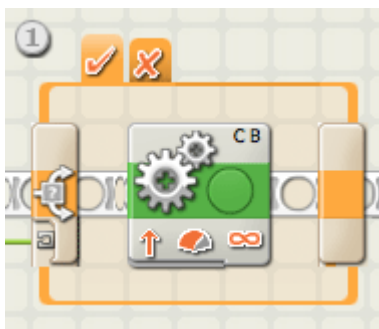


1. Эта пиктограмма показывает датчик или какое-либо условие, при выполнении которого Блок «Переключатель» должен будет запустить выполнение той или иной последовательности программных Блоков. В этом случае программа будет переключаться в зависимости от текущего состояния датчика касания.
2. Если датчик касания нажат, будут запущены Блоки на верхней цепочке программы.
3. Если датчик касания не нажат, будут запущены Блоки на нижней цепочке программы.



1. Если вы решили управлять Блоком «Переключатель» при помощи условия "Значение", в нижней части левого края Блока появится разъём шины данных; чтобы осуществлять переключение, необходимо присоединить к этому разъёму логическую или числовую шину данных, идущую от какого-то другого программного Блока.





1. Если снять флажок “Вид: Без закладок”, то для отображения возможных последовательностей программных Блоков будет предложен интерфейс с набором закладок. Щёлкнув на закладке, можно просмотреть или отредактировать программные Блоки и узнать, при выполнении каких условий эти Блоки были запущены.



Примечание: для подключения внешней шины данных к Блоку, находящемуся внутри Блока «Переключатель», нужно отменить выбор “Вид: Без закладок”, чтобы показать Блок «Переключатель» с вкладками.

## Добавление программных Блоков в Блок «Переключатель»:

Перетащите Блоки на свободное место внутри рамки, ограничивающей Блок «Переключатель». При этом рамка будет раздвигаться, и Блок можно будет встроить в нужном месте. Если внутри рамки уже есть Блоки, перетащите новые Блоки на нужное место цепочки программы, при этом соседние Блоки раздвинутся.

## Перемещение Блока «Переключатель»:

Блок «Переключатель» можно выделить и перетащить в нужное место, только если щёлкнуть на нём самом.

## Настройка Блока «Переключатель»:

В открывающемся списке секции «Контролировать» можно выбрать одно из двух условий: «Значение» или «Датчик»:

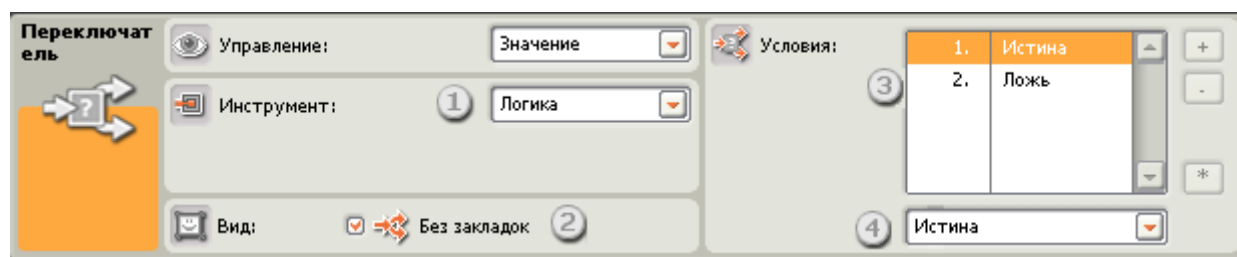
- [Значение](#)
- Датчик (при таком выборе открывается следующий список)
  - [Касания](#)
  - [Освещённости](#)
  - [Звука](#)
  - [Расстояния](#)
  - [Кнопка NXT](#)
  - [Оборотов](#)
  - [Таймер](#)
  - [Приём сообщений](#)
  - [Температуры\\*](#)

**Значение**

Блок «Переключатель» с управлением по "Значению" может принимать как логические, так и числовые сигналы через шину данных, подключенную к его разъёму, расположенному в левой части Блока. Блок определит тип подведённой шины данных (т.е., логическая она или числовая), и предложит различные варианты переключения, соответствующие каждому случаю.

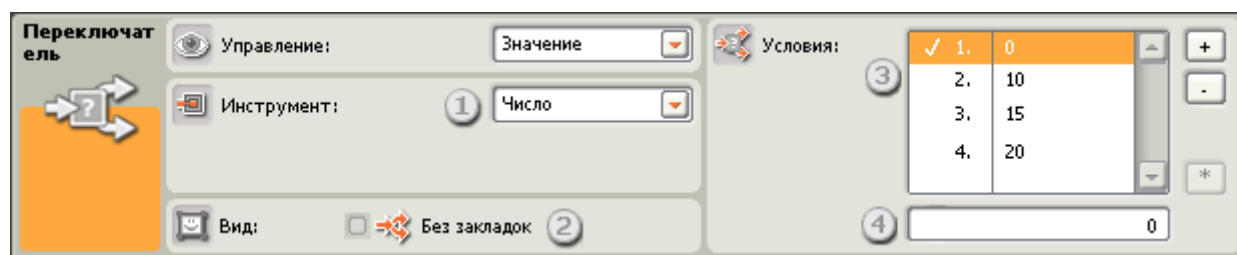
Если подключена логическая ("истина"/"ложь") шина данных, то при получении Блоком «Переключатель» команды "истина", программа запустит выполнение верхней цепочки программы. При получении команды "ложь", будет запущена нижняя цепочка.

При подключении числовой или текстовой шины данных, можно задать точное значение входного сигнала, по которому будет запущена та или иная цепочка (программы). Более того, если отключить отображение «Вид: Без закладок», можно будет запустить более двух цепочек программы.



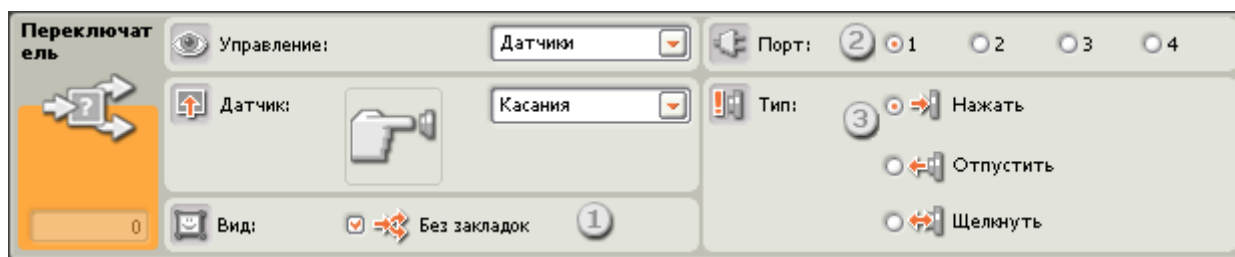
1. В этом поле будет показан тип входного сигнала, определённый Блоком.
2. Если снять флажок в окошке «Вид: Без закладок», то для отображения возможных последовательностей Блоков будет предложен интерфейс с набором закладок. Щёлкнув на закладке, можно просмотреть или отредактировать Блоки и узнать, при выполнении каких условий были запущены эти Блоки.
3. Каждый ряд в таблице представляет собой вариант переключения. Если включено отображение «Вид: Без закладок», самый верхний вариант, номер 1 - это условие, при котором будет запущена верхняя цепочка (программы). По второму будет запущена нижняя цепочка.

Если числовая или текстовая шина данных подключена к Блоку «Переключатель», а отображение «Вид: Без закладок» отменено (при этом включается диалоговый интерфейс с отображением закладок), можно добавить дополнительные строки таблицы, что позволит управлять большим количеством цепочек программы, как показано на рисунке внизу.



4. Это поле выполняет две функции. Если параметр Инструмент имеет значение "Логика", то в этом поле можно назначить условие ("истина") включения одной из двух цепочек. Если это значение "Число" или "Текст", в это поле для каждой цепочки вводят число или текст, которые будут её запускать.
5. При помощи этих кнопок можно добавлять или удалять варианты переключения. Они активны только если снят флажок «Вид: Без закладок», и к входному разъёму Блока «Переключатель» протянута числовая или текстовая шина данных.
6. Кнопка "\*" присваивает выделенной цепочке статус "запуск по умолчанию". Когда Блок «Переключатель» настроен на числовой входной сигнал, и поступающий сигнал не совпадает с заданными числовыми значениями, Блок начнёт работать независимо от того, какой вариант установлен по умолчанию.

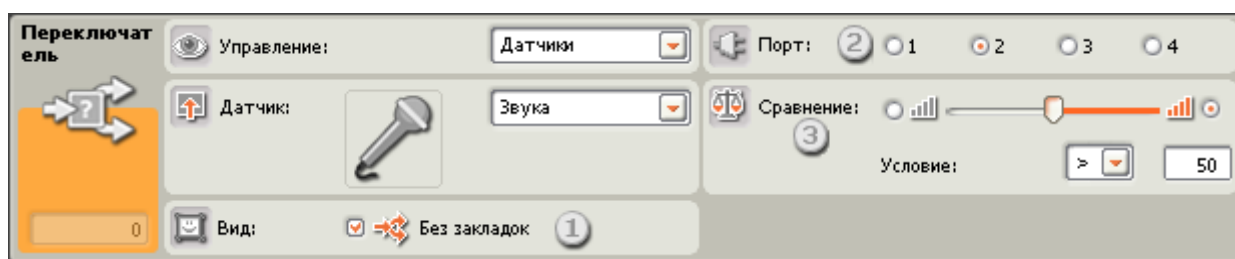
## Датчик касания



Выберите этот пункт, чтобы программа переключалась в зависимости от состояния датчика касания.

1. Если снять флажок “Вид: Без закладок”, то для отображения возможных вариантов программы будет предложен интерфейс с набором закладок. Щёлкнув на закладке, можно просмотреть и отредактировать Блоки каждого варианта и узнать, при выполнении каких условий эти варианты будут выбраны.
2. Выберите порт, к которому подключен датчик касания. По умолчанию для датчика касания будет назначен порт 1. При необходимости эту настройку можно изменить.
3. Кнопками выбора можно задать при каком состоянии датчика касания Блок «Переключатель» запустит Блоки в верхней цепочке; нижняя цепочка включится в том случае, если не было произведено никаких действий. При выборе "Нажать", Блок переключается при нажатии кнопки датчика касания. При выборе "Отпустить", Блок переключается после освобождения кнопки датчика. Если выбран вариант "Щелкнуть", Блок переключается при щелчке кнопкой датчика.

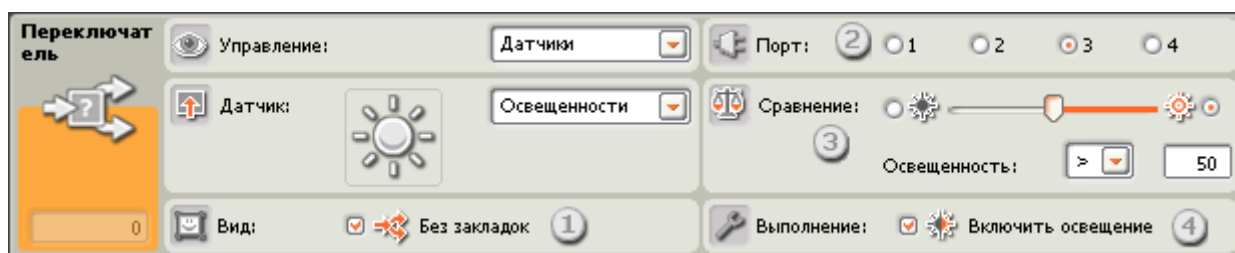
## Датчик звука



Выберите этот пункт, чтобы программа переключалась в зависимости от показаний датчика звука.

1. Если снять флажок “Вид: Без закладок”, то для отображения возможных последовательностей программных Блоков будет предложен интерфейс с набором закладок. Щёлкнув на закладке, можно просмотреть или отредактировать Блоки и узнать, при выполнении каких условий были запущены эти Блоки.
2. Выберите порт, к которому подключен датчик звука. По умолчанию для датчика звука Блоку будет назначен порт 2. При необходимости эту настройку можно изменить.
3. Задайте пороговое значение движком или введите значение непосредственно в поле. При установке по умолчанию (50%), будет запущена верхняя цепочка (программы), если датчик зарегистрирует звук, уровень которого превышает 50%; если уровень громкости будет меньше 50%, включится нижняя цепочка.

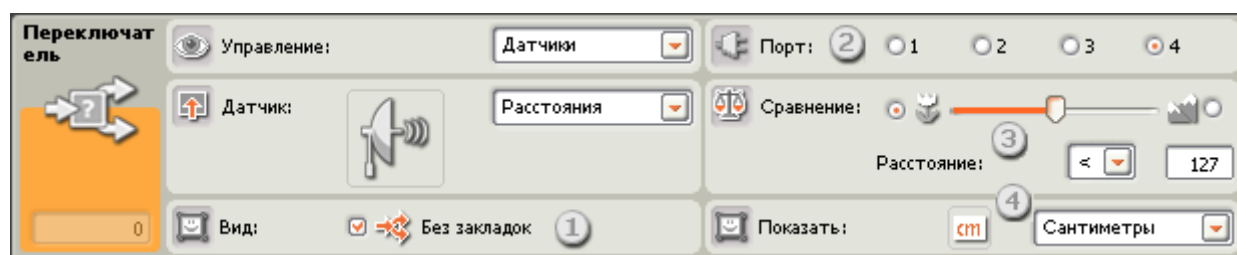
## Датчик освещённости



Выберите этот пункт, чтобы программа переключалась в зависимости от показаний датчика освещённости.

1. Если снять флажок “Вид: Без закладок”, то для отображения возможных последовательностей программных Блоков будет предложен интерфейс с набором закладок. Щёлкнув на закладке, можно просмотреть или отредактировать программные Блоки и узнать, при выполнении каких условий были запущены эти Блоки.
2. Выберите порт, к которому подключен датчик освещённости. По умолчанию для датчика освещённости Блоку будет назначен порт 3. При необходимости эту настройку можно изменить.
3. Задайте пороговое значение движком или введите значение непосредственно в поле. При установке по умолчанию (50%), будет запущена верхняя цепочка (программы), если датчик зарегистрирует уровень освещённости выше 50%; если уровень освещённости будет меньше 50%, включится нижняя цепочка.
4. Если установлен флажок “Включить освещение”, датчик включит свою лампу подсветки и зарегистрирует отражённый свет.
5. В поле обратной связи отображаются текущие показания датчика освещённости (0-100%).

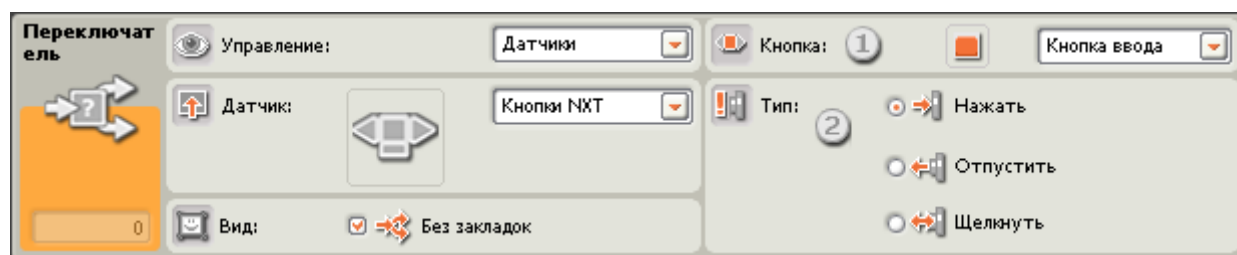
## Датчик расстояния



Выберите этот пункт, чтобы программа переключалась в зависимости от показаний датчика расстояния.

1. Если снять флажок “Вид: Без закладок”, то для отображения возможных последовательностей программных Блоков будет предложен интерфейс с набором закладок. Щёлкнув на закладке, можно просмотреть или отредактировать программные Блоки и узнать, при выполнении каких условий были запущены эти Блоки.
2. Выберите порт, к которому подключен датчик расстояния. По умолчанию для датчика расстояния Блоку будет назначен порт 4. При необходимости эту настройку можно изменить.
3. Задайте пороговое значение движком или введите его непосредственно в поле. При установке по умолчанию 50 (127), будет запущена верхняя цепочка (программы), если датчик обнаружит объект на расстоянии превышающем 50 дюймов (127 см), если объект будет зарегистрирован на расстоянии меньшем, чем 50 дюймов (127 см), включится нижняя цепочка.
4. Выберите единицы измерения расстояния - сантиметры или дюймы.
5. В поле обратной связи отображаются текущие показания датчика расстояния.

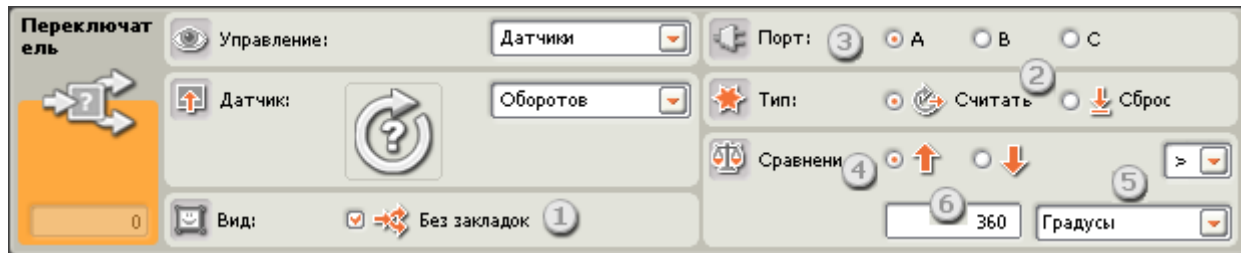
## Кнопки NXT



При выборе датчика “Кнопки NXT”, будет запущена верхняя цепочка (программы), если будет зарегистрировано какое-либо воздействие на выбранную кнопку NXT. Если никаких действий не будет зарегистрировано, включится нижняя цепочка (программы).

1. Выберите кнопку NXT, при нажатии которой будет отправлена команда "истина" (и будет запущена верхняя цепочка программы).
2. Выберите "Щелкнуть", чтобы переключение происходило после щелчка кнопкой NXT. Выберите "Нажать", чтобы переключение происходило при нажатии на неё. Выберите "Отпустить", чтобы переключение происходило при освобождении кнопки.

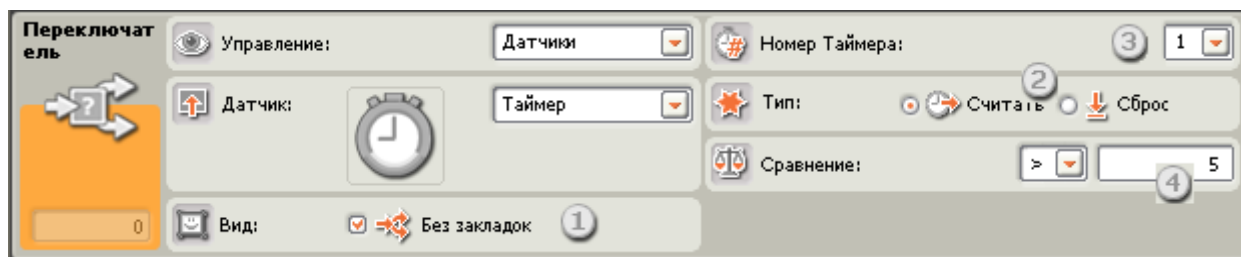
## Датчик оборотов



Выберите этот пункт, чтобы программа переключалась в зависимости от количества оборотов или градусов поворота вала мотора.

1. Если снять флажок “Вид: Без закладок”, то для отображения возможных последовательностей программных Блоков будет предложен интерфейс с набором закладок. Щёлкнув на закладке, можно просмотреть или отредактировать программные Блоки и узнать, при выполнении каких условий были запущены эти Блоки.
2. Если параметр Тип имеет значение "Сброс", Блок «Переключатель» будет считывать показания датчика оборотов и обнулять счётчик. Если выбрать "Считать", показания датчика обнуляться не будут.
3. Укажите порт, к которому подключен мотор, используемый Блоком «Переключатель».
4. Кнопками выбора укажите направление вращения мотора (вперед или назад), которое должно контролироваться Блоком.
5. Выберите из списка единицу измерения "Обороты" или "Градусы".
6. Введите пороговое значение в выбранных единицах измерения. По умолчанию пороговое значение равно 360 градусов. При такой настройке запустится верхняя цепочка (программы), если будет зарегистрирован поворот вала мотора больше, чем 360 градусов. При меньших значениях программа пойдет по нижней цепочке.
7. В поле обратной связи отображается текущее количество оборотов или градусов.

## Таймер

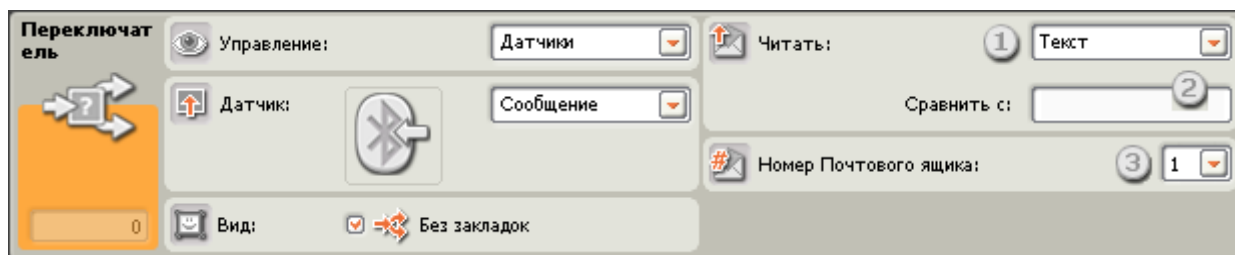


Выберите этот пункт, чтобы программа переключалась в зависимости от текущих показаний таймера.

1. Если снять флажок “Вид: Без закладок”, то для отображения возможных последовательностей программных Блоков будет предложен интерфейс с набором закладок. Щёлкнув на закладке, можно просмотреть или отредактировать программные Блоки и узнать, при выполнении каких условий были запущены эти Блоки.
2. Если выбрать "Сброс", Блок «Переключатель» будет считывать показания таймера и обнулять его. Если выбрать "Считать", показания таймера обнуляться не будут.
3. Выберите один из трех таймеров NXT, который будет использован.

4. Введите пороговое значение. По умолчанию задано 5 секунд. При такой настройке запустится верхняя цепочка (программы), если текущее показание превысит 5 секунд, а если будет зарегистрировано меньше 5 секунд, включится нижняя цепочка.

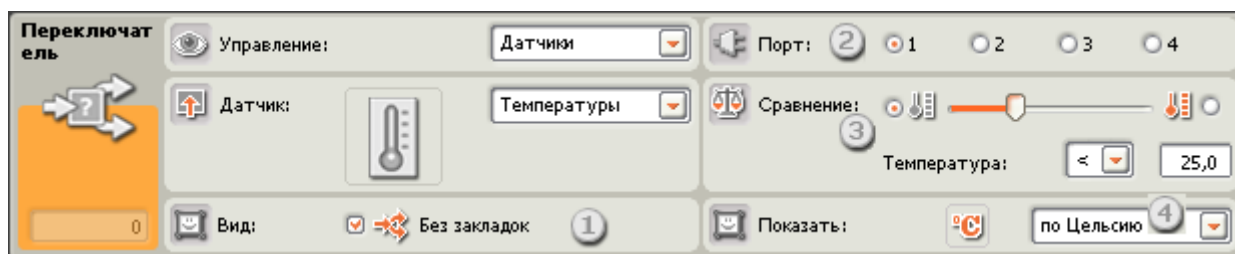
## Сообщение



Если выбрать “Сообщение”, то при поступлении определенного сообщения по Bluetooth будет запущена верхняя цепочка (программы). Если такое сообщение не поступит, включится нижняя цепочка.

1. В открывающемся списке можно выбрать тип ожидаемого сообщения (текстовое, числовое или логическое).
2. Для сравнения входящего сообщения с контрольным, наберите тестовый текст или число (в зависимости от того, какой формат был выбран: текстовый или числовой), или кнопками выбора задайте контрольную логическую команду ("истина" или "ложь"). Если входящее сообщение совпадает с тестовым текстом или значением, будет запущена верхняя цепочка (программы). В противном случае включится нижняя цепочка.
3. Выберите номер почтового ящика, в который будет поступать сообщение.

## Датчик температуры\*



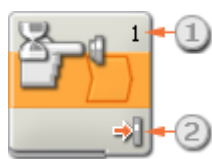
Выберите этот пункт, чтобы программа переключалась в зависимости от показаний Датчика температуры.

1. Если снять флажок “Вид: Без закладок”, то для отображения возможных последовательностей программных Блоков будет предложен интерфейс с набором закладок. Щёлкнув на закладке, можно просмотреть или отредактировать программные Блоки и узнать, при выполнении каких условий были запущены эти Блоки.
2. Выберите порт, к которому подключен датчик температуры. По умолчанию для датчика температуры Блоку будет назначен порт 4. При необходимости эту настройку можно изменить.
3. Пороговое значение можно задать движком или ввести нужное значение в поле. Значение по умолчанию 25° C или 77° F. В открывающемся списке выберите “<”, тогда, если температура окажется ниже, чем 25° C/77° F, будет запущена верхняя цепочка (программы), если температура превысит 25° C/77° F, включится нижняя цепочка. При выборе “>” в открывающемся списке, эти операции будут выполняться наоборот, как и при активации противоположной кнопки выбора.
4. Выберите шкалу измерения температуры - градусы Цельсия или Фаренгейта.
5. В поле обратной связи отображаются текущие значения температуры.

## Блок «Ожидание»



Этот Блок придаёт роботу способность следить за окружающей его обстановкой, ожидая наступления определенных условий, чтобы продолжить свои действия. Воспользуйтесь движком, или введите число, чтобы задать пороговое значение времени или показания датчика.



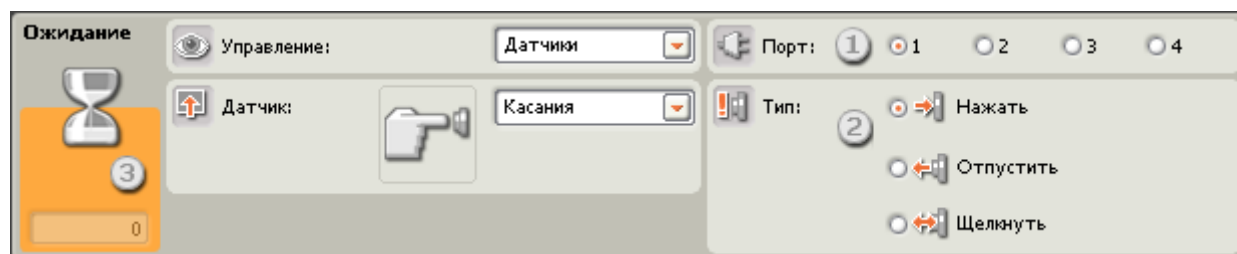
1. Цифра или буква в верхнем левом углу Блока «Ожидание» указывает порт, отслеживаемый блоком. Порты при необходимости можно сменить в панели настройки.
2. При выборе ожидания сигнала датчика освещённости, звука или расстояния, эта пиктограмма показывает установленный уровень порогового значения. Чем больше окрашенных полосок, тем выше пороговое значение. Если выбрано ожидание сигнала датчика касания, пиктограмма отображает, какое воздействие на этот датчик продолжит выполнение программы (Щелчок, Нажатие или Освобождение кнопки датчика).

## Настройка Блока «Ожидание»

В списке "Управление", выбирается условие, управляющее Блоком: время, или показание датчика:

- Датчик (открывается второй список)
  - [Касания](#)
  - [Звука](#)
  - [Освещённости](#)
  - [Расстояния](#)
  - [Кнопки NXT](#)
  - [Оборотов](#)
  - [Таймер](#)
  - [Сообщение](#)
- [Температуры\\*](#)
- [Время](#)

### Датчик Касания



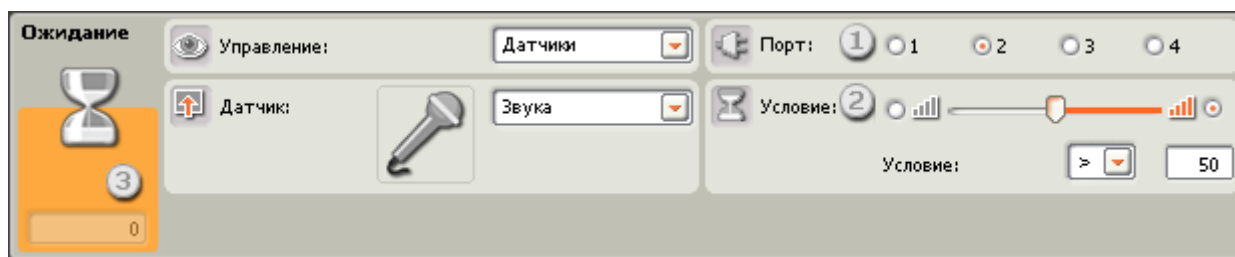
При выборе "Датчик касания" программа будет ожидать заданного состояния датчика касания (Щелчок, Нажатие или Освобождение), прежде чем перейти к следующему блоку.

1. Выберите порт, к которому подключен датчик касания. По умолчанию назначается порт 1.
2. Укажите действие (Щелкнуть, Нажать или Отпустить), которое должна ожидать программа для перехода к следующему Блоку. Если выбрано "Щелкнуть", Блок будет срабатывать при резком надавливании и ослаблении давления на кнопку датчика касания. "Нажать" - Блок будет срабатывать

при нажатии на кнопку датчика касания. "Отпустить" - Блок сработает при ослаблении давления на кнопку датчика касания.

3. Поле обратной связи позволяет контролировать состояние датчика касания. Если датчик работа активирован, в поле появится "1".

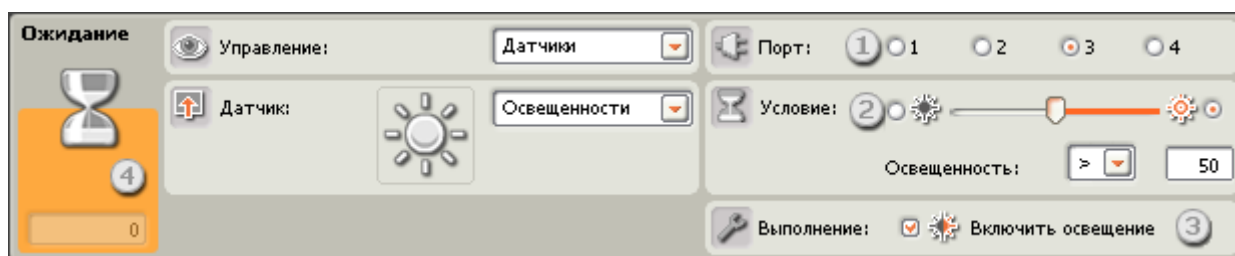
## Датчик Звука



Выберите "Датчик звука", и программа будет ожидать, когда датчик звука зарегистрирует звук заданной громкости.

1. Выберите порт, к которому подключен датчик звука. По умолчанию для датчика звука назначен порт 2.
2. Задайте пороговое значение движком, или напечатайте его в поле ввода. Активируйте правую (относительно движка) кнопку выбора, чтобы Блок срабатывал при уровнях громкости, превышающих пороговое значение. Активируйте левую кнопку выбора, чтобы Блок срабатывал при уровнях громкости меньших, чем пороговое значение. Установить движок в диапазоне подачи команды "истина" можно и при помощи открывающегося списка.
3. В поле обратной связи отображаются текущие показания датчика звука (0-100%). Его можно использовать для опробования различных пороговых значений.

## Датчик освещённости

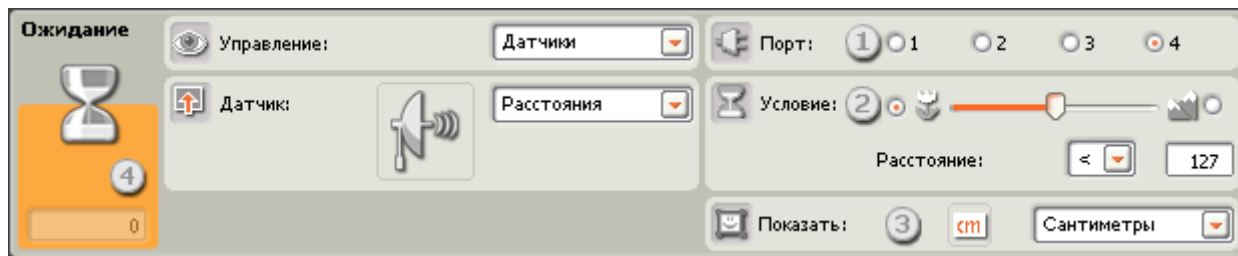


Выберите "Датчик освещённости", и программа будет ожидать, когда датчик освещённости зарегистрирует заданный уровень освещённости.

1. Выберите порт, к которому подключен датчик освещённости. По умолчанию для него блоку будет назначен порт 3.
2. Задайте пороговое значение движком, или введите его в поле. Активируйте правую (относительно движка) кнопку выбора, чтобы Блок срабатывал при уровнях освещённости, превышающих пороговое значение. Активируйте левую кнопку выбора, чтобы Блок срабатывал при уровнях освещённости меньших, чем пороговое значение. Установить движок в диапазоне подачи команды "истина" можно и при помощи открывающегося списка;.
3. Если установлен флажок "Включить освещение", датчик включит собственную лампу подсветки и зарегистрирует отражённый свет.
4. Поле обратной связи отображает текущие показания датчика освещённости (0-100%). Его можно использовать для опробования различных пороговых значений.

## Датчик Расстояния

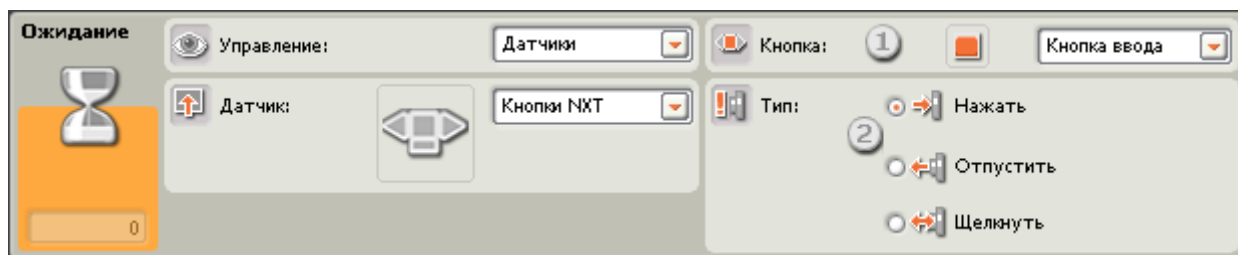




Выберите "Датчик расстояния", и программа будет ожидать, когда датчик расстояния обнаружит объекта на определенном удалении.

1. Выберите порт, к которому подключен датчик расстояния. По умолчанию для него блоку будет назначен порт 4.
2. При активации левой кнопки выбора, Блок запускается при регистрации объекта, находящегося на расстоянии меньшем, чем пороговое значение; если активирована правая кнопка выбора, Блок включится при обнаружении объекта на расстоянии большем, чем пороговое значение. Задайте движком пороговое значение, или введите его непосредственно в поле (0-250, если расстояние измеряется в сантиметрах, или 0-100 - для дюймов). Помните о том, что объекты с хорошо отражающей поверхностью можно обнаружить на большем расстоянии, чем объекты со слабо отражающей поверхностью.
3. Выберите единицу измерения расстояния - сантиметры или дюймы.
4. В поле обратной связи отображаются текущие показания датчика расстояния (0-250 см или 0-100 дюймов). Показание "0" соответствует минимальному расстоянию, на котором датчик может обнаруживать объекты. Максимальное расстояние обнаружения составляет приблизительно 250 см (показание 250) или 100 дюймов (показание 100).

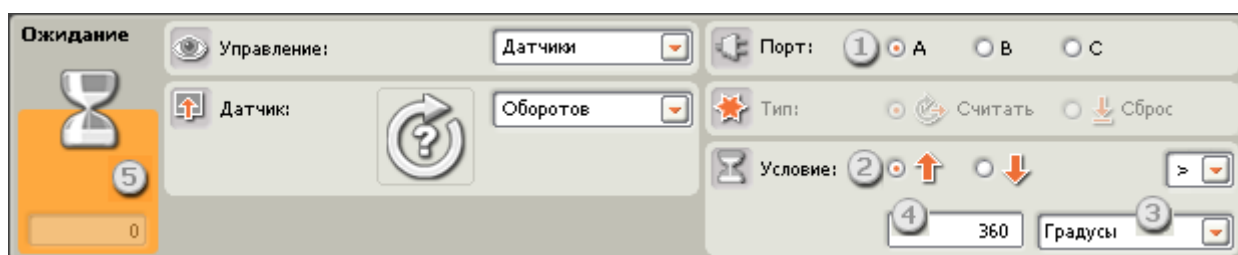
## Кнопки NXT



Если выбрать "Кнопки NXT", программа будет ожидать совершения определённого действия (Щелчок, Нажатие или Освобождение) выбранной кнопкой NXT, прежде чем продолжить выполнение.

1. Выберите, при активации какой кнопки NXT будет выдан сигнал "истина" для продолжения выполнения программы.
2. Щелкнуть - сигнал возникает после быстрого нажатия и освобождения выбранной кнопки. "Нажать" - сигнал возникает после при нажатии выбранной кнопки. "Отпустить" - сигнал возникает после при освобождении выбранной кнопки.

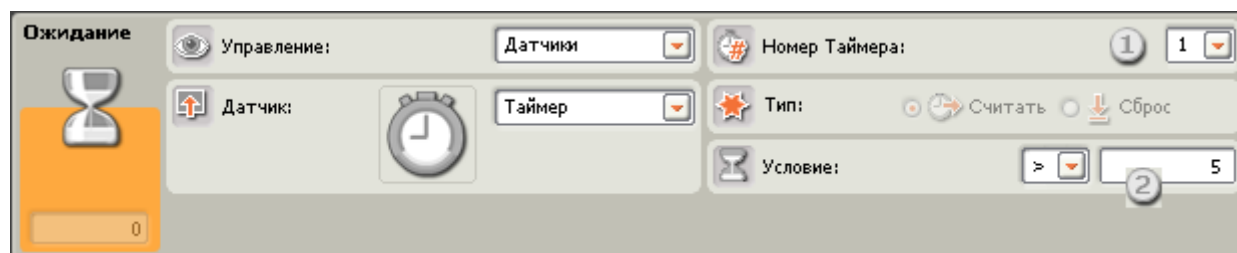
## Датчик оборотов



При выборе "Датчика оборотов" выполнение программы продолжится после того, как вал мотора повернется на заданное количество оборотов или градусов.

1. Выберите порт, к которому подключен мотор.
2. Кнопками выбора задайте направление вращения мотора: вперед или назад.
3. В открывающемся списке выберите "Обороты" или "Градусы".
4. Задайте, на какое количество оборотов или градусов должен повернуться вал мотора, после чего выполнение программы сможет продолжиться.
5. В поле обратной связи будет отображаться текущее количество оборотов или градусов. Для очистки поля обратной связи нажмите кнопку Сброс.

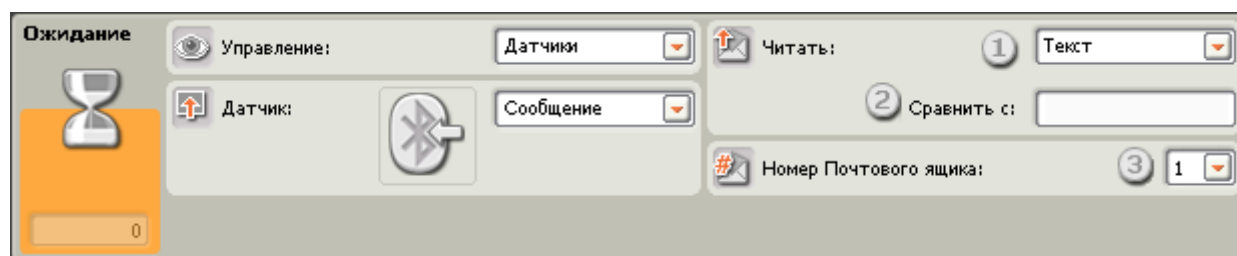
## Таймер



Если выбрать "Таймер", выполнение программы продолжится после того, как один из внутренних таймеров NXT отсчитает заданный промежуток времени.

1. Выберите таймер NXT, который будет отображаться и контролироваться.
2. Введите значение времени (в секундах) в поле. Это значение можно увеличивать или уменьшать при помощи стрелок вверх-вниз.

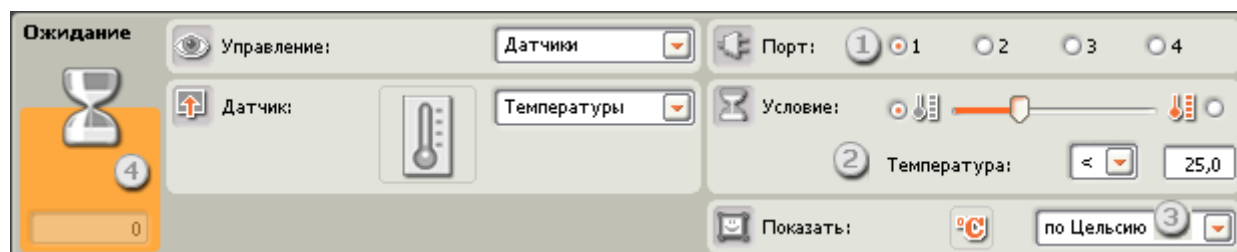
## Приём сообщений



Если выбрать "Сообщение", выполнение программы продолжится после того, как NXT получит определенное сообщение по Bluetooth.

1. В открывающемся списке можно выбрать тип ожидаемого сообщения (текстовое, числовое или логическое).
2. Для сравнения входящего сообщения с контрольным, наберите тестовый текст или число (если для сообщения выбран соответственно текстовый или числовой формат), или кнопками выбора задайте контрольное логическое значение ("истина" или "ложь").
3. Выберите номер почтового ящика, в который будет попадать сообщение.

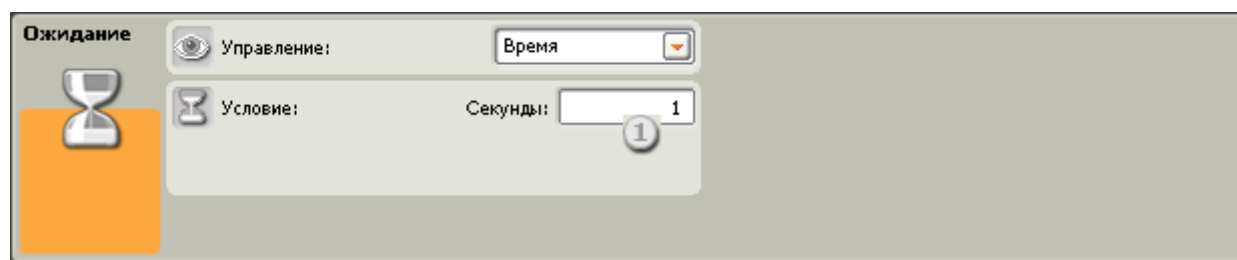
## Датчик температуры\*



Если выбрать "Датчик температуры"\*, выполнение программы продолжится после того, как датчик температуры зарегистрирует её заданное значение.

1. Выберите порт, к которому подключен датчик температуры. По умолчанию для него блоку будет назначен порт 4.
2. Задайте пороговое значение движком, или введите его в поле. Активируйте правую (относительно движка) кнопку выбора, чтобы Блок срабатывал при температуре, превышающей пороговое значение. Активируйте левую кнопку выбора, чтобы Блок срабатывал при температурах меньших, чем пороговое значение. Установить движок в диапазоне подачи команды "истина" можно и при помощи открывающегося списка.
3. Выберите шкалу измерения температуры - градусы Цельсия или Фаренгейта.
4. Поле обратной связи будет отображать текущие значения температуры.

## Время



Выберите "Время", чтобы выполнение программы приостанавливалось на заданный промежуток времени, прежде чем перейти к следующему блоку.

1. Выберите количество секунд, которое нужно ждать, прежде чем продолжить выполнение программы. Время можно задать с точностью до десятых долей секунды (например: 12,3 или 12.3 (в английском языке)).

## Блоки группы Действия

- [Блок «Экран»](#)
- [Блок «Лампа»\\*](#)
- [Блок «Мотор»](#)
- [Блок «Отправить сообщение»](#)
- [Блок «Звук»](#)

## Блок «Экран»

Этот блок используется для вывода изображений, набора текстового фрагмента или рисования на дисплее NXT.

Последовательно устанавливая несколько Блоков «Экран», можно создавать сложные изображения, дополняя их картинками, текстом и формами, усложняя их при помощи каждого дополнительного Блока «Экран».

Если перед началом работы вы хотите очистить экран, установите флажок "Очистить".

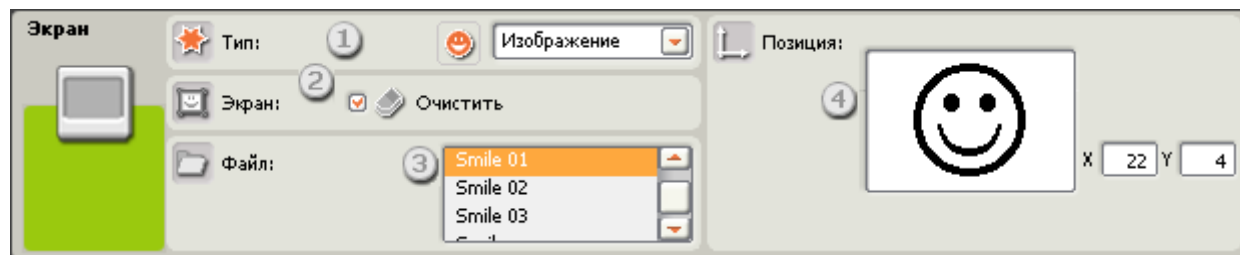
Чтобы восстановить пиктограмму LEGO MINDSTORMS, установленную по умолчанию, выберите в списке "Сброс".

Размер экрана - 100 пикселей в ширину и 64 пикселей в высоту.



1. Эта пиктограмма показывает назначение блока: настроен ли он на изображение, текст или рисунок, или просто возвращает на дисплей пиктограмму, установленную по умолчанию.
2. Значения можно изменять динамически, подключая шины данных к концентратору данных этого блока.

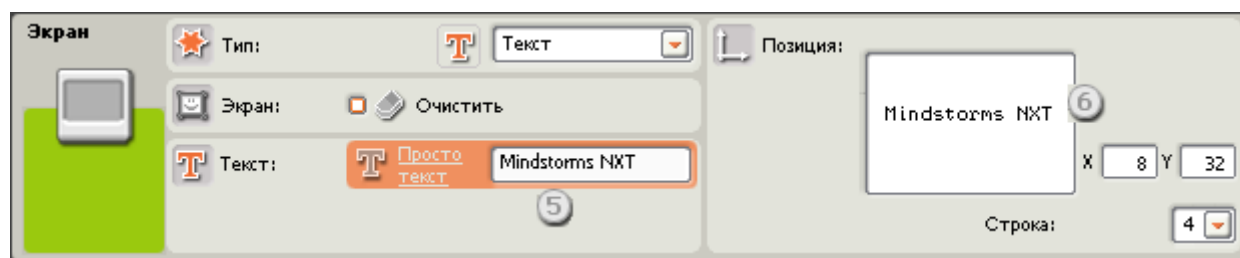
## Настройка Блока «Экран»:



1. В открывающемся списке можно выбрать: отображать картинку, текст или чертеж, или же просто сбросить настройки дисплея.
2. Чтобы очистить дисплей NXT, установите флажок "Очистить".

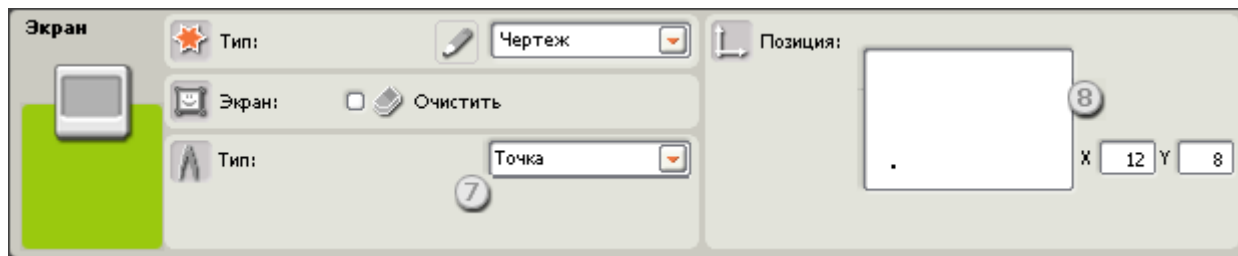
Если выбрано отображение картинки (как показано на панели настройки), можно задать следующие дополнительные функции:

3. Функция Файл позволяет выбрать картинку для отображения на дисплее. Щёлкните на названии файла, чтобы просмотреть хранящуюся в нем картинку.
4. При помощи мыши переместите картинку, текст или рисунок по экрану предварительного просмотра, чтобы определить, где они будут расположены. Поля ввода координат X и Y позволяют очень точно позиционировать изображение. Если в оба поля ввести значение "0", изображение будет помещено в левую нижнюю часть дисплея.



Если выбрано отображение текста (как показано на панели настройки), можно задать следующие дополнительные функции:

5. Функция Текст позволяет вводить текст для отображения на дисплее.
6. В поле "Строка" выберите номер строки, на которой следует разместить начало текста или используйте поля ввода координат X и Y, чтобы назначить точное место размещения начала текста. Если в оба поля ввести значение "0", текст будет помещен в левую нижнюю часть дисплея.



Если выбран Чертеж, то можно задать следующие дополнительные функции:

7. Выбрать инструмент черчения - точку, прямую или окружность.
8. При выборе точки, величины, введенные в поля координат X и Y определяют положение точки на экране. Если в оба поля ввести значение "0", точка будет помещена в левую нижнюю часть дисплея.

Если выбрать линию, появятся дополнительные поля ввода для задания крайних точек отрезка прямой. Начальная точка задаётся в верхних полях ввода координат x и y, а конечная точка - в нижних полях ввода координат x и y.

При выборе окружности, появится дополнительное поле для ввода радиуса окружности. Введите координаты центра окружности в первые два поля координат, а величину радиуса - в поле Радиус.

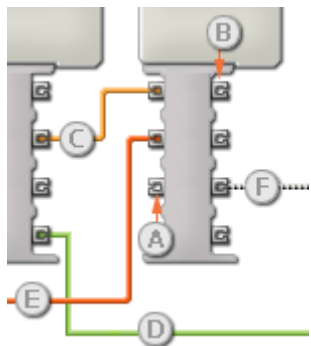
## Настройка концентратора данных Блока «Экран»

Блоком «Экран» можно управлять динамически, подключая шины данных (от концентраторов данных других блоков) к концентратору данных Блока «Экран».

Перетащив блок в рабочую область, откройте его концентратор данных, щёлкнув на кнопке в нижней левой части блока.



Шины данных, передающие входящие данные на блок, подключены к разъёмам, расположенным на левой стороне их концентраторов данных. Шины данных, передающие исходящие данные, подключены к разъёмам, расположенным справа.



- [A] Входной разъём
- [B] Выходной разъём
- [C] Числовая шина данных (желтая)
- [D] Логическая шина данных (зеленая)
- [E] Текстовая шина данных (оранжевая)
- [F] Повреждённая шина данных (серая)

### Передача данных от входного разъёма на выходной разъём

Если для входного разъёма имеется соответствующий выходной разъём (см. выше А), входящие данные будут проходить от входного разъёма на выходной разъём без изменений. В этом случае можно использовать выходной разъём, если входной разъём подключен к шине данных входа; подключение шины данных выхода к такому выходному разъёму без подключенной шины данных входа приведет к «повреждению» шины данных выхода (она станет серой).

## **По шинам данных передаются данные различных типов**

Через каждую шину данных осуществляется обмен данными различных типов между блоками. Например, если шина данных протянута с логического разъёма на концентратор данных блока, её можно подключить только к логическому разъёму на концентраторе данных другого блока. На приведенной ниже схеме показано, данные какого типа может принимать или передавать каждый из разъёмов.

## **Цвет шин данных**

Шины данных окрашены в различные цвета: шины для передачи числовых данных - желтые; шины, для передачи логических данных - зеленые; шины для передачи текстовых данных - оранжевые.

## **«Повреждённые» шины данных**

При попытке подключить шину данных к разъёму данных несоответствующего типа, шина будет повреждена (ее цвет изменится на серый). Вы не сможете загрузить программу, если шина данных повреждена.

Если щёлкнуть на повреждённой шине, то в небольшом окне контекстной справки в правом нижнем углу рабочей области появится информация о причине повреждения.

## **Данные должны соответствовать возможному диапазону, в котором работает разъём**

Если шина данных входа передает значение, не попадающее в допустимый диапазон разъёма, к которому она подключена, блок либо проигнорирует такие данные, либо приведёт их к значению, попадающему в этот диапазон. Если разъём допускает ограниченное количество значений (например, только 0, 1 или 2), он проигнорирует входящий сигнал, если передаваемое значение не будет вписываться в допустимый диапазон.

Если разъём допускает более широкий диапазон значений (например, 0 – 100), любой входящий сигнал, не попадающий в допустимый диапазон, будет приведен к значению, соответствующее ему. Например, если разъём мощности блока «Движение» получает входящий сигнал со значением 150, блок преобразует входящее значение в 100 (т.е., в числовое значение, лежащее в пределах диапазона разъёма мощности ).

## **Блок «Лампа»\***

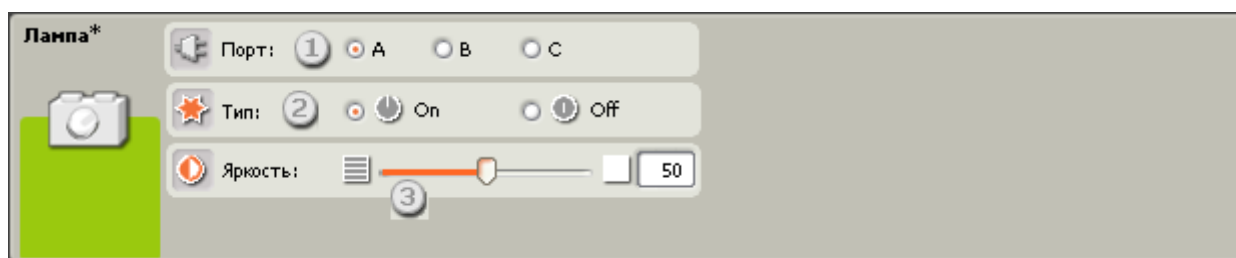
Этот блок предназначен для управления работой лампы. Вам понадобится один такой блок для включения лампы и ещё один блок для её выключения.

\*Для подключения этой лампы к блоку NXT требуется кабель-переходник. Чтобы получить дополнительную информацию обратитесь в LEGO Education.  
(<http://www.lego.com/education/>)



1. Буквы в правом верхнем углу Блока «Лампа»\* указывают на порт NXT, которым будет управлять блок. Если лампа подключена к порту А, одна из букв должна быть буквой А. При необходимости можно назначить другой порт с панели настройки.
2. Эта пиктограмма показывает настройку блока - включение или выключение лампы.
3. Пиктограмма в правом нижнем углу отображает яркость лампы. Четыре белые полосы на пиктограмме свидетельствует о максимальной яркости.
4. Чтобы изменить свойства Блока «Лампа»\*, можно перетащить шины данных с других блоков на концентратор данных этого блока. (Дополнительную информацию см. в разделе "Концентратор данных".)

## Настройка Блока «Лампа»\*



1. Назначьте порты [А, В и/или С], которыми будет управлять блок. Можно назначить одновременно несколько портов (для управления более чем одной лампой). Если выбрать опцию управления более чем одной лампой при помощи одного Блока «Лампа»\*, то для всех ламп будет задана одинаковая яркость. Для того чтобы лампы горели с различной яркостью, потребуется несколько Блоков «Лампа»\*.
2. Кнопками выбора укажите, что должен сделать блок - включить или выключить лампы.
3. Яркость лампы задаётся при помощи движка "Яркость". При перемещении движка вправо яркость лампы увеличивается. Уровень яркости также можно задать непосредственно в открывающемся списке.

## Настройка концентратора данных Блока «Лампа»\*

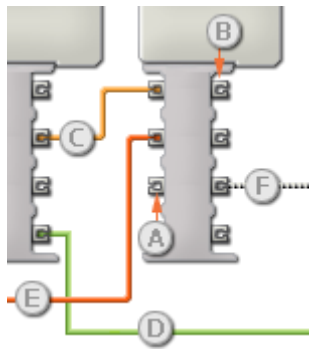
Блоком «Лампа»\* можно управлять динамически, подключая шины данных (от концентраторов данных других блоков) к концентратору данных Блока «Лампа»\*.

Перетащив блок в рабочую область, откройте его концентратор данных, щёлкнув на кнопке в нижней левой части блока.



Шины данных, передающие входящие данные на блок, подключены к разъёмам, расположенным на левой стороне их концентраторов данных. Шины данных, передающие исходящие данные, подключены к разъёмам, расположенным справа.





- [A] Входной разъём
- [B] Выходной разъём
- [C] Числовая шина данных (желтая)
- [D] Логическая шина данных (зеленая)
- [E] Текстовая шина данных (оранжевая)
- [F] Повреждённая шина данных (серая)

## Передача данных от входного разъёма на выходной разъём

Если для входного разъёма имеется соответствующий выходной разъём (см. выше А), входящие данные будут проходить от входного разъёма на выходной разъём без изменений. В этом случае можно использовать выходной разъём, если входной разъём подключен к шине данных входа; подключение шины данных выхода к такому выходному разъёму без подключенной шины данных входа приведет к «повреждению» шины данных выхода (она станет серой).

## По шинам данных передаются данные различных типов

Через каждую шину данных осуществляется обмен данными различных типов между блоками. Например, если шина данных протянута с логического разъёма на концентратор данных блока, её можно подключить только к логическому разъёму на концентраторе данных другого блока. На приведенной ниже схеме показано, данные какого типа может принимать или передавать каждый из разъёмов.

## Цвет шин данных

Шины данных окрашены в различные цвета: шины для передачи числовых данных - желтые; шины, для передачи логических данных - зеленые; шины для передачи текстовых данных - оранжевые.

## «Повреждённые» шины данных

При попытке подключить шину данных к разъёму данных несоответствующего типа, шина будет повреждена (ее цвет изменится на серый). Вы не сможете загрузить программу, если шина данных повреждена.

Если щёлкнуть на повреждённой шине, то в небольшом окне контекстной справки в правом нижнем углу рабочей области появится информация о причине повреждения.

## Данные должны соответствовать возможному диапазону, в котором работает разъём

Если шина данных входа передает значение, не попадающее в допустимый диапазон разъёма, к которому она подключена, блок либо проигнорирует такие данные, либо приведёт их к значению, попадающему в этот диапазон. Если разъём допускает ограниченное количество значений (например, только 0, 1 или 2), он проигнорирует



входящий сигнал, если передаваемое значение не будет вписываться в допустимый диапазон.

Если разъём допускает более широкий диапазон значений (например, 0 – 100), любой входящий сигнал, не попадающий в допустимый диапазон, будет приведен к значению, соответствующее ему. Например, если разъём мощности блока «Движение» получает входящий сигнал со значением 150, блок преобразует входящее значение в 100 (т.е., в числовое значение, лежащее в пределах диапазона разъёма мощности).

## Блок «Мотор»

Этот блок позволяет с высокой точностью управлять оборотами одного мотора. Вы можете повышать обороты до заданного значения, или снижать их вплоть до полной остановки мотора. Если снять флажок "До завершения", программа может перейти к следующему блоку сразу же после того, как Блок «Мотор» включит мотор.

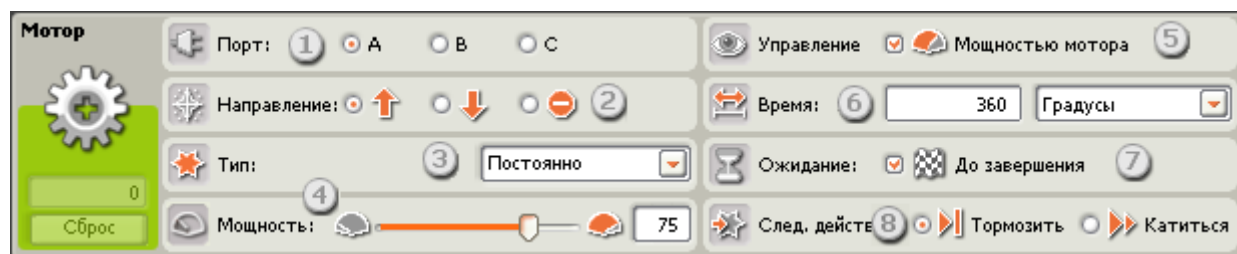


1. Буква в верхнем правом углу блока показывает, через какой порт NXT осуществляется управление мотором.
2. Эта пиктограмма показывает направление вращения мотора.
3. Эта пиктограмма показывает уровень мощности.
4. Эта пиктограмма показывает как задана продолжительность работы мотора: Без ограничения, количеством угловых градусов, оборотов или секунд.
5. Настройки Блока «Мотор» можно изменять динамически, подключая шины данных к его концентратору данных.



1. Этот значок показывает, что для параметра «Направление» установлено значение «Стоп». При этой установке указанный мотор останавливается.

## Настройка Блока «Мотор»



1. Выберите порт, к которому подключен настраиваемый мотор.
2. Задайте направление вращения: вперед, назад или выключение мотора.
3. Выберите «Увеличить» или «Снизить», чтобы мощность мотора плавно увеличивалась до заданного уровня или плавно снижалась до 0. Если выбрано «Постоянно», то заданная мощность достигается практически сразу.
4. Задайте уровень мощности мотора в диапазоне [0-100 %] при помощи движка или поля ввода.

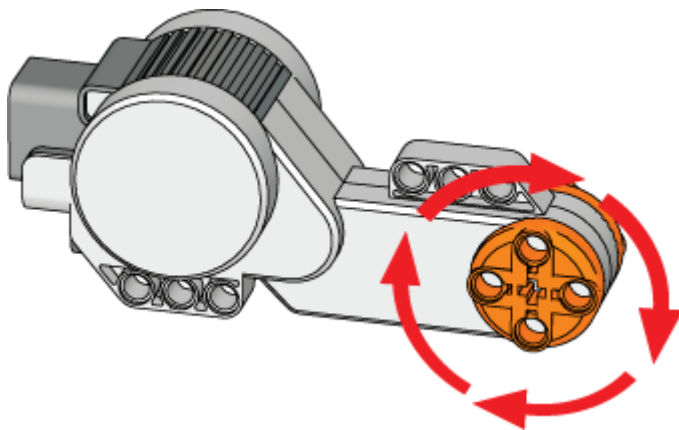
5. Установите флажок для включения управления мощностью мотора. Эта функция позволяет корректировать работу мотора при его перегрузке или пробуксовке. Для поддержания постоянного числа оборотов мощность может быть повышена до максимального значения 100 %.
6. Задайте продолжительность работы мотора: в течение неограниченного времени, по достижении заданного количества оборотов, или поворота вала мотора на заданное количество градусов, или в течении заданного времени в секундах.
7. Если установлен флажок «До завершения», то выполнение программы продолжится только после полной остановки мотора. Если снять этот флажок, то последующие блоки программы могут выполняться, пока мотор останавливается. Например, Блок «Звук», установленный после Блока «Мотор», может начать воспроизводить звуковой файл, в то время, когда мотор ещё работает.
8. Выберите, будет ли мотор тормозить или двигаться по инерции после выключения. Выберите Тормозить, если мотор должен остановиться в заданном положении и зафиксировать это положение. Торможение потребует дополнительного расхода заряда аккумулятора, поскольку мотор работает с повышенной нагрузкой, фиксируя положение. Движение по инерции (Катиться) рекомендуется использовать, если за этим блоком следует другой блок, управляющий тем же мотором.
9. В поле обратной связи отображается количество градусов поворота вала мотора или количество полных оборотов, совершённых им. Нажмите кнопку сброса чтобы обнулить показания. (Эту информацию можно получать, если мотор подключен к выбранному порту и установлена связь с NXT.)



Примечание: если мотор вращается вперед (положительное направление), значение в поле обратной связи отображается чёрным, а если он вращается назад - красным.

## Вращение мотора в сторону, противоположную направлению движения робота

При установке направления Блока «Движение» или Блока «Мотор» задается направление вращения мотора. Оно может отличаться от фактического направления движения робота: например, робот может двигаться назад при вращении мотора вперед. На рисунке показано вращение мотора вперед (по умолчанию).



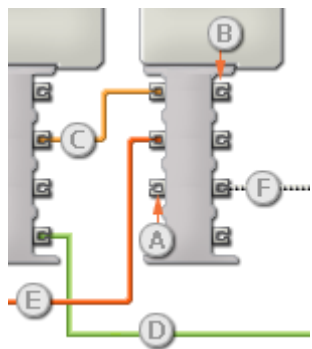
## Настройка концентратора данных Блока «Мотор»

Блоком «Мотор» можно управлять динамически, подключая к его концентратору данных шины данных (от концентраторов данных других блоков).

Перетащив блок в рабочую область, откройте его концентратор данных, щёлкнув на кнопке в нижней левой части блока.



Шины данных, передающие входящие данные на блок, подключены к разъёмам, расположенным на левой стороне их концентраторов данных. Шины данных, передающие исходящие данные, подключены к разъёмам, расположенным справа.



- [A] Входной разъём
- [B] Выходной разъём
- [C] Числовая шина данных (желтая)
- [D] Логическая шина данных (зеленая)
- [E] Текстовая шина данных (оранжевая)
- [F] Повреждённая шина данных (серая)

### **Передача данных от входного разъёма на выходной разъём**

Если для входного разъёма имеется соответствующий выходной разъём (см. выше A), входящие данные будут проходить от входного разъёма на выходной разъём без изменений. В этом случае можно использовать выходной разъём, если входной разъём подключен к шине данных входа; подключение шины данных выхода к такому выходному разъёму без подключенной шины данных входа приведет к «повреждению» шины данных выхода (она станет серой).

### **По шинам данных передаются данные различных типов**

Через каждую шину данных осуществляется обмен данными различных типов между блоками. Например, если шина данных протянута с логического разъёма на концентратор данных блока, её можно подключить только к логическому разъёму на концентраторе данных другого блока. На приведенной ниже схеме показано, данные какого типа может принимать или передавать каждый из разъёмов.

### **Цвет шин данных**

Шины данных окрашены в различные цвета: шины для передачи числовых данных - желтые; шины, для передачи логических данных - зеленые; шины для передачи текстовых данных - оранжевые.

### **«Повреждённые» шины данных**

При попытке подключить шину данных к разъёму данных несоответствующего типа, шина будет повреждена (ее цвет изменится на серый). Вы не сможете загрузить программу, если шина данных повреждена.

Если щёлкнуть на повреждённой шине, то в небольшом окне контекстной справки в правом нижнем углу рабочей области появится информация о причине повреждения.

### **Данные должны соответствовать возможному диапазону, в котором работает разъём**

Если шина данных входа передает значение, не попадающее в допустимый диапазон разъёма, к которому она подключена, блок либо проигнорирует такие данные, либо приведёт их к значению, попадающему в этот диапазон. Если разъём допускает ограниченное количество значений (например, только 0, 1 или 2), он проигнорирует входящий сигнал, если передаваемое значение не будет вписываться в допустимый диапазон.

Если разъём допускает более широкий диапазон значений (например, 0 – 100), любой входящий сигнал, не попадающий в допустимый диапазон, будет приведен к значению, соответствующее ему. Например, если разъём мощности блока «Движение» получает входящий сигнал со значением 150, блок преобразует входящее значение в 100 (т.е., в числовое значение, лежащее в пределах диапазона разъёма мощности ).

## **Программный блок отправки сообщений**

Этот блок позволяет отправить сообщение с помощью беспроводной связи на другое устройство NXT.

### **Настройка устройства NXT для беспроводной связи**

Перед отправкой или приемом сообщений с помощью беспроводной связи необходимо настроить параметры беспроводной связи соответствующих устройств NXT.

На (мастере) устройстве NXT, которое устанавливает связь с остальными устройствами NXT, выберите категорию Bluetooth, затем «Поиск», в результате чего начнется поиск других устройств. По завершении поиска выберите номер подключения (1, 2 или 3), присваиваемый второму (дополнительному) устройству NXT. При первой установке подключения между двумя NXT на обоих устройствах появится запрос ввода ключа установки на обоих устройствах NXT; эту операцию следует проделать на обоих устройствах. При следующем выборе на мастере NXT категории «Подключение» под номером выбранного подключения должно отобразиться дополнительное устройство NXT. На дополнительном устройстве NXT мастер NXT должен отображаться под номером подключения 0. После данных настроек мастер NXT будет готов к установке связи с дополнительным устройством NXT.

Если мастер NXT должен связываться с несколькими устройствами NXT (не более трех), то повторите вышеописанный процесс.

**Примечание!** Если между несколькими устройствами NXT установлена связь и мастер NXT также устанавливает связь с несколькими устройствами NXT, то необходимо предусмотреть задержку по меньшей мере в одну секунду, чтобы радиоприемник Bluetooth переключал каналы перед отправкой сообщений.

Этот процесс нужно выполнить только на одном устройстве NXT. Принимающее устройство NXT автоматически обновит подключение, когда мастером NXT будет настраиваться номер подключения.

### **Номер соединения**

Номер подключения каждого устройства NXT представляет его «адрес» в беспроводной сети. Для отправки сообщения на конкретное устройство NXT укажите его номер подключения.


Программный блок соединения через Bluetooth можно использовать также для соединения с устройствами NXT из списка контактов. Более подробная информация приведена в разделе программный блок соединения через Bluetooth

## Номер почтового ящика

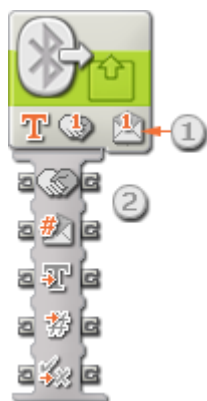
На каждом устройстве NXT имеется десять номеров почтовых ящиков, где сохраняются переданные с помощью беспроводной связи сообщения. Каждый номер почтового ящика может содержать до пяти сообщений. Если номер почтового ящика содержит пять сообщений, то при приеме следующего сообщения устройство NXT удалит самое раннее сообщение.

## Отправка сообщения

Для отправки сообщения с помощью беспроводной связи (после настройки номеров подключений на всех устройствах NXT) сначала выберите номер подключения на целевом устройстве NXT. Затем выберите формат сообщения (текст, число или логическое значение). Введите сообщение, набрав текст или число либо выбрав логическое условие (истина или ложь). В завершение выберите номер почтового ящика (где сохранится сообщение) на целевом устройстве NXT. Формат сообщения и само сообщение можно также указать с помощью шин входных данных.

 Примечание: Дополнительные сведения о приеме сообщения с помощью беспроводной связи см. в разделе «Программный блок приема сообщений».

## Настройки отображения



1. Это номер подключения на целевом устройстве NXT (принимающем сообщение).
2. Тип сообщения и само сообщение можно также ввести динамически с помощью шин входных данных. По мере необходимости выходные разъемы справа передают любые входящие данные на другие блоки. (Чтобы получить дополнительную информацию, см. ниже раздел "Концентратор данных".)

## Настройка Программного блока отправки сообщений



1. С помощью селективных кнопок выберите номер подключения на целевом устройстве NXT.
2. Выберите в раскрывающемся меню тип сообщения (текст, число или логическое значение).

3. Введите сообщение, набрав текст или число (если выбран текст или число соответственно) либо, если отправляется логическое сообщение, используя селективные кнопки для выбора значения «истина» или «ложь».
4. Выберите номер почтового ящика на конечном устройстве NXT.

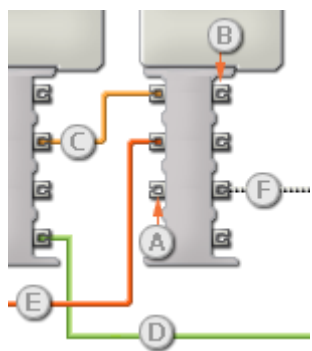
## Настройка концентратора данных Программного блока отправки сообщений

Вы можете управлять Программным блоком отправки сообщений динамически, подключая шины данных (от концентраторов данных других блоков) к концентратору данных Программного блока отправки сообщений.

Поместив блок в рабочую область, откройте концентратор данных блока, щелкнув по кнопке в нижней левой части блока.



Шины данных, передающие на блок входящие данные, подключены к разъемам, находящимся на левой стороне его концентратора данных. Шины данных, передающие исходящие данные, подключены к разъемам, находящимся на правой стороне.



- [A] Входной разъем
- [B] Выходной разъем
- [C] Числовая шина данных (желтая)
- [D] Логическая шина данных (зеленая)
- [E] Текстовая шина данных (оранжевая)
- [F] Поврежденная шина данных (серая)

### Передача данных от входного разъема на выходной разъем

Если для входного разъема есть соответствующий выходной разъем (см. выше A), входящие данные будут проходить от входного разъема на выходной разъем без изменений. В этом случае вы сможете использовать выходной разъем, если входной разъем подключен к шине данных входа; подключение шины данных выхода к такому выходному разъему без подключенной шины данных входа приведет к «неисправности» шины данных выхода (она будет окрашена в серый цвет).

### Шины данных осуществляют передачу специфических типов данных

Каждая шина данных осуществляет обмен специфическими типами данных между блоками. Например, если шина данных перетянута с логического разъема на концентратор данным блока, ее можно подключить только к логическому разъему на

концентраторе данных другого блока. На приведенной ниже схеме показано, какой вид данных может принимать или передавать каждый из разъемов.

## Окраски шин данных

Шины данных различаются особыми окрасками: шины, передающие числовые данные, имеют желтый цвет; шины, передающие логические данные, имеют зеленый цвет; шины, передающие текстовые данные, имеют оранжевый цвет.

## «Поврежденные» шины данных

Если вы пытаетесь подключить шину данных к разъему несоответствующего типа данных, шина будет повреждена (ее цвет изменится на серый). Вы не сможете загрузить вашу программу, если шина данных повреждена.

Если вы щелкнете по поврежденной шине, то в небольшом окне контекстной справки в правом нижнем углу рабочей области вы сможете прочесть информацию о причине повреждения.

## Данные должны соответствовать возможному диапазону, в котором работает разъем

Если шина данных входа передает значение, не попадающее в допустимый диапазон разъема, к которому она подключена, блок либо проигнорирует такие данные, либо трансформирует данные, попадающие в этот диапазон. Для разъемов, допускающих небольшое количество значений (например, только 0, 1 или 2), разъем проигнорирует входящий сигнал, если передаваемое значение не будет вписываться в диапазон разъема.

Для разъемов, допускающих более широкий диапазон значений (например, 0 – 100), разъем усилит любой входящий сигнал, если он не соответствует диапазону. Например, если разъем питания программного блока перемещения получает входящий сигнал со значением 150, блок преобразует входящее значение в 100 (т.е., в числовое значение, лежащее в пределах диапазона разъема мощности).

## Блок «Звук»

Этот Блок можно использовать для воспроизведения звукового файла или какого-либо одиночного звука. Чтобы составить мелодию из отдельных звуков, выстройте в ряд несколько Блоков "Звук", каждый из которых настроен для воспроизведения различных звуков (нот).

Если на панели настройки установить флажок "До завершения", переход к следующему Блоку программы совершится только после того, как закончится воспроизведение звукового файла или отдельного звука. Если этот флажок не установлен, то воспроизведение звукового файла или отдельного звука может продолжаться и во время исполнения следующего Блока программы.

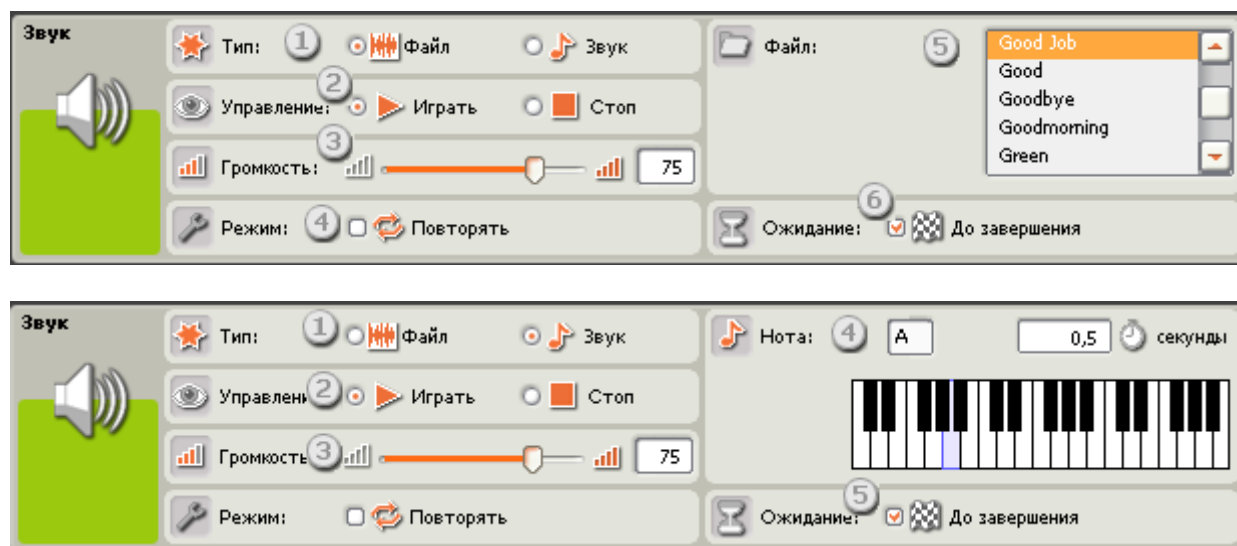
Если установить флажок "Повторять" звук будет воспроизводиться непрерывно.





1. Эта пиктограмма показывает настройку Блока: воспроизведение звукового файла или отдельного звука.
2. Эта пиктограмма показывает настройку Блока: включение или прекращение воспроизведения звука.
3. Эта пиктограмма показывает уровень громкости. Четыре оранжевые полосы на пиктограмме означают максимальную громкость.
4. Если протянуть шины данных от других Блоков на концентратор данных Блока «Звук», это повлияет на его свойства.

## Настройка Блока «Звук»



1. Укажите типа Блока, то есть, что должно воспроизводиться - звуковой файл или отдельный звук.
2. В секции Управление выберите действие Блока: начать или остановить воспроизведение звука.
3. Установите уровень громкости при помощи движка, или вписав значение в поле ввода.

Специальные параметры Блока "Файл":

4. Установите флажок "Повторять", чтобы звуковой файл проигрывался безостановочно. При этом поле "До завершения" станет недоступным.
5. В списке "Файл" выберите файл по названию мелодии. Чтобы услышать звук файла сделайте щелчок на его имени.
6. Если установлен флажок "Подождать завершения", звуковой файл будет воспроизведен полностью, прежде чем программа перейдет к следующему Блоку. В противном случае воспроизведение звукового файла может продолжаться и при исполнении следующего программного Блока. Если следующий Блок окажется тоже звуковым, воспроизведение первого звукового файла автоматически прекратится и начнется воспроизведение нового.

Специальные параметры Блока "Звук":

4. В секции "Нота" при помощи музыкальной клавиатуры укажите ноту, которую будет воспроизводить Блок «Звук». На клавиатуре уместаются три октавы. В полях над клавиатурой отображаются название ноты и продолжительность ее звучания в секундах.
5. Если установлен флажок "До завершения", звук будет воспроизводиться до конца, прежде чем программе будет разрешено перейти к исполнению следующего Блока. В противном случае воспроизведение звука может продолжаться и при исполнении следующего Блока.

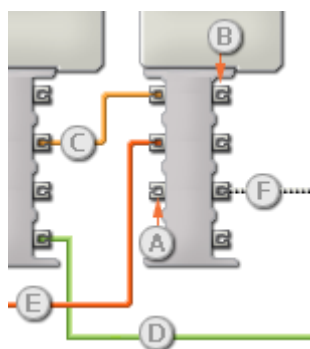
## Настройка концентратора данных Блока «Звук»

Блоком «Звук» можно управлять динамически, подключая к его концентратору данных шины данных (от концентраторов данных других Блоков).

Перетащив Блок в рабочую область, откройте его концентратор данных, щёлкнув на кнопке в нижней левой части Блока.



Шины данных, передающие входящие данные на Блок, подключены к разъёмам, расположенным на левой стороне их концентраторов данных. Шины данных, передающие исходящие данные, подключены к разъёмам, расположенным справа.



- [A] Входной разъём
- [B] Выходной разъём
- [C] Числовая шина данных (желтая)
- [D] Логическая шина данных (зеленая)
- [E] Текстовая шина данных (оранжевая)
- [F] Повреждённая шина данных (серая)

### Передача данных от входного разъёма на выходной разъём

Если для входного разъёма имеется соответствующий выходной разъём (см. выше А), входящие данные будут проходить от входного разъёма на выходной разъём без изменений. В этом случае можно использовать выходной разъём, если входной разъём подключен к шине данных входа; подключение шины данных выхода к такому выходному разъёму без подключенной шины данных входа приведет к «повреждению» шины данных выхода (она станет серой).

### По шинам данных передаются данные различных типов

Через каждую шину данных осуществляется обмен данными различных типов между Блоками. Например, если шина данных протянута с логического разъёма на концентратор данных Блока, её можно подключить только к логическому разъёму на концентраторе данных другого Блока. На приведенной ниже схеме показано, данные какого типа может принимать или передавать каждый из разъёмов.

### Цвет шин данных

Шины данных окрашены в различные цвета: шины для передачи числовых данных - желтые; шины, для передачи логических данных - зеленые; шины для передачи текстовых данных - оранжевые.

### «Повреждённые» шины данных

При попытке подключить шину данных к разъёму данных несоответствующего типа, шина будет повреждена (ее цвет изменится на серый). Вы не сможете загрузить программу, если шина данных повреждена.

Если щёлкнуть на повреждённой шине, то в небольшом окне контекстной справки в правом нижнем углу рабочей области появится информация о причине повреждения.

### **Данные должны соответствовать возможному диапазону, в котором работает разъём**

Если шина данных входа передает значение, не попадающее в допустимый диапазон разъёма, к которому она подключена, Блок либо проигнорирует такие данные, либо приведёт их к значению, попадающему в этот диапазон. Если разъём допускает ограниченное количество значений (например, только 0, 1 или 2), он проигнорирует входящий сигнал, если передаваемое значение не будет вписываться в допустимый диапазон.

Если разъём допускает более широкий диапазон значений (например, 0 – 100), любой входящий сигнал, не попадающий в допустимый диапазон, будет приведен к значению, соответствующее ему. Например, если разъём мощности Блока «Движение» получает входящий сигнал со значением 150, Блок преобразует входящее значение в 100 (т.е., в числовое значение, лежащее в пределах диапазона разъёма мощности ).

#### **Блоки группы Датчики**

- [Блок «Датчик освещённости»](#)
- [Блок «Кнопки NXT»](#)
- [Блок «Приём сообщений»](#)
- [Блок «Датчик оборотов»](#)
- [Блок «Датчик звука»](#)
- [Блок «Датчик температуры»](#)
- [Блок «Таймер»](#)
- [Блок «Датчик касания»](#)
- [Блок «Датчик расстояния»](#)

## **Блок «Датчик освещённости»**

Этот датчик фиксирует общую освещённость (рассеянный свет). Через шину данных датчик может передавать текущие значения освещённости или логический сигнал ("истина" или "ложь"), в зависимости от того, превышает текущий уровень освещённости пороговое значение, или он ниже.

«Пороговое значение» - это определённое значение из диапазона, в котором происходит изменение условий. Например, можно запрограммировать робота на движение вперёд, если уровень освещённости превысит 60%. Пороговое значение будет равно 60.

Задайте пороговое значение движком или введите его в соответствующее поле. Чтобы задать ту часть диапазона датчика расстояния (выше или ниже порогового значения ), в которой будет выдаваться команда "истина", используйте кнопки выбора или список. Та часть диапазона, в которой выдаётся сигнал "истина", будет окрашена, а участок генерирования сигнала "ложь" будет серым.

По умолчанию Блок «Датчик освещённости» настроен на выдачу команды "истина" при уровне освещённости, превышающем 50%. Активирована правая кнопка выбора, а

движок установлен на значение 50. Чтобы переключить диапазоны, и команда "истина" выдавалась при уровне освещённости ниже 50%, нужно выбрать левую кнопку.

Поле обратной связи на панели настройки показывает текущие показания датчика расстояния. (Получить отражённый сигнал можно лишь тогда, когда датчик подключен к выбранному порту и установлена связь с NXT.)

Для передачи любой информации, необходимо перетащить хотя бы одну выходную шину данных от концентратора данных этого блока на другой блок. (Дополнительную информацию см. в разделе "Концентратор данных".)



Совет: как задать пороговое значение для датчика освещённости?

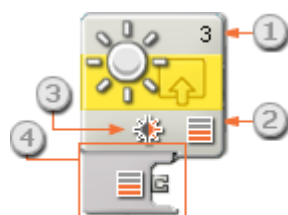
Если вы хотите, чтобы Блок «Датчик освещённости» выдавал сигнал "истина", когда на робота направлена лампа подсветки, несколько раз направьте её на робота, наблюдая за уровнями освещённости в поле обратной связи. Если подсветка создала уровень освещённости около 80%, можно установить пороговое значение на 70. Таким образом, при освещённости, превышающей 70% (как и в случае с подсветкой), будет выдан сигнал "истина". Более низкая освещённость будет игнорироваться.

Пороговое значение можно задать динамически, при помощи шины данных входа.

## Регистрация датчиком освещённости отражения собственной подсветки

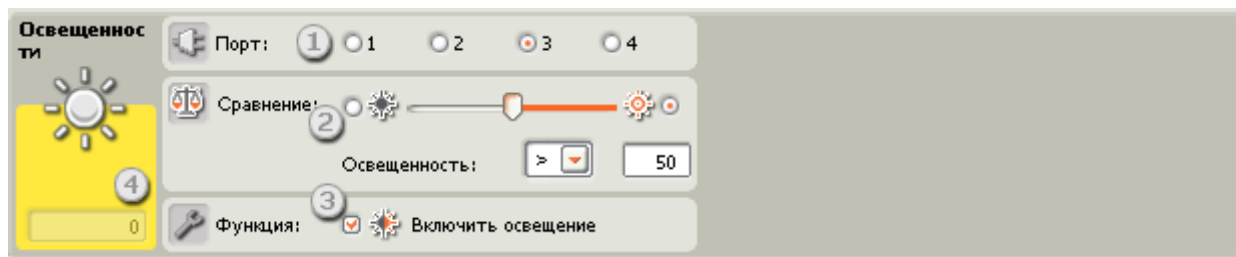
Если установлен флажок "Включить освещение", датчик освещённости включит свою собственную лампу подсветки и зарегистрирует её отражённый свет. Эта функция оказывается крайне полезной в сложных условиях освещённости, например, в помещениях с очень ярким освещением. Эта функция также позволяет использовать датчик освещённости в качестве дальномера на небольших расстояниях. Если установить флажок "Включить освещение", датчик, по мере приближения к объектам, отражающим свет, будет регистрировать отражённый свет нарастающей яркости. Это даёт возможность использовать датчик освещённости для предотвращения столкновения с препятствиями.

## Отображение параметров настройки блока



1. Число указывает, к какому порту NXT подключен Датчик освещённости. При необходимости это число можно изменить в панели настройки.
2. Эта пиктограмма показывает при каком уровне освещённости блок среагирует. Чем больше на пиктограмме окрашенных полосок, тем для более высокой освещённости задано пороговое значение.
3. Пиктограмма отображается в том случае, если активирована функция "Включить подсветку".
4. Когда программный блок перетащен в рабочую область, его концентратор данных открывается автоматически. Хотя бы одну шину данных следует перетащить с выходного разъёма этого блока на концентратор данных другого блока. (Дополнительную информацию см. в разделе "Концентратор данных".)

## Настройка Блока «Датчик освещённости»



1. Выберите порт, к которому подключен Датчик освещённости. По умолчанию Блоку «Датчик освещённости» назначен Порт 3. При необходимости эту настройку можно изменить.
2. Пороговое значение можно задать движком или ввести нужное значение в поле. Активируйте кнопку выбора справа от движка, чтобы блок срабатывал при уровнях освещённости, превышающих пороговое значение; активируйте левую кнопку выбора, чтобы блок срабатывал при уровнях освещённости более низких, чем пороговое значение. Для задания условия срабатывания датчика можно воспользоваться открывающимся списком.
3. Если установлен флажок "Включить освещение", датчик включает свою лампу подсветки и регистрирует отражённый свет.
4. В поле обратной связи отображаются текущие показания Датчика освещённости (0-100%). Этим можно воспользоваться для опробования различных пороговых значений.

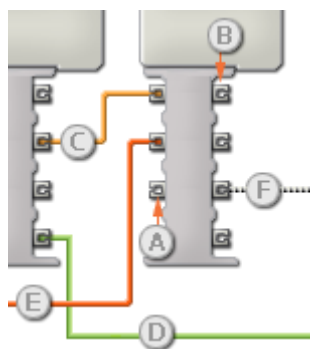
## Настройка концентратора данных Блока «Датчик освещённости»

Блоком «Датчик освещённости» можно управлять динамически, подключая шины данных (от концентраторов данных других блоков) к концентратору данных Блока «Датчик освещённости».

Перетащите блок в рабочую область и откройте его концентратор данных, щёлкнув на кнопке в нижней левой части блока.



Шины данных, передающие входящие данные на блок, подключены к разъёмам, расположенным на левой стороне их концентраторов данных. Шины данных, передающие исходящие данные, подключены к разъёмам, расположенным справа.



- [A] Входной разъём
- [B] Выходной разъём
- [C] Числовая шина данных (желтая)
- [D] Логическая шина данных (зеленая)

- [E] Текстовая шина данных (оранжевая)
- [F] Повреждённая шина данных (серая)

## **Передача данных от входного разъёма на выходной разъём**

Если для входного разъёма имеется соответствующий выходной разъём (см. выше А), входящие данные будут проходить от входного разъёма на выходной разъём без изменений. В этом случае можно использовать выходной разъём, если входной разъём подключен к шине данных входа; подключение шины данных выхода к такому выходному разъёму без подключенной шины данных входа приведет к «повреждению» шины данных выхода (она станет серой).

## **По шинам данных передаются данные различных типов**

Через каждую шину данных осуществляется обмен данными различных типов между блоками. Например, если шина данных протянута с логического разъёма на концентратор данных блока, её можно подключить только к логическому разъёму на концентраторе данных другого блока. На приведенной ниже схеме показано, данные какого типа может принимать или передавать каждый из разъёмов.

## **Цвет шин данных**

Шины данных окрашены в различные цвета: шины для передачи числовых данных - желтые; шины, для передачи логических данных - зеленые; шины для передачи текстовых данных - оранжевые.

## **«Повреждённые» шины данных**

При попытке подключить шину данных к разъёму данных несоответствующего типа, шина будет повреждена (ее цвет изменится на серый). Вы не сможете загрузить программу, если шина данных повреждена.

Если щёлкнуть на повреждённой шине, то в небольшом окне контекстной справки в правом нижнем углу рабочей области появится информация о причине повреждения.

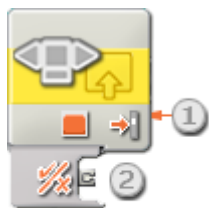
## **Данные должны соответствовать возможному диапазону, в котором работает разъём**

Если шина данных входа передает значение, не попадающее в допустимый диапазон разъёма, к которому она подключена, блок либо проигнорирует такие данные, либо приведёт их к значению, попадающему в этот диапазон. Если разъём допускает ограниченное количество значений (например, только 0, 1 или 2), он проигнорирует входящий сигнал, если передаваемое значение не будет вписываться в допустимый диапазон.

Если разъём допускает более широкий диапазон значений (например, 0 – 100), любой входящий сигнал, не попадающий в допустимый диапазон, будет приведен к значению, соответствующее ему. Например, если разъём мощности блока «Движение» получает входящий сигнал со значением 150, блок преобразует входящее значение в 100 (т.е., в числовое значение, лежащее в пределах диапазона разъёма мощности ).

## **Блок «Кнопка NXT»**

Этот блок посылает команду "истина" через шину данных, когда одна из кнопок блока NXT активирована. Выберите кнопку и действие, которое будет сигналом к отправке команды "истина".



1. Пиктограмма в нижней части блока показывает, какая из кнопок блока NXT была выбрана.
2. После встраивания программного блока в цепочку программы, его концентратор данных открывается автоматически. Необходимо протянуть шину данных от Блока «Кнопка NXT» к другому блоку, способному принимать логический сигнал (истина/ложь) (Дополнительную информацию см. в разделе "Концентратор данных").

## Настройка Блока «Кнопка NXT»



1. Выберите, при нажатии на которую из кнопок, NXT будет выдавать сигнал "истина".
2. Выберите "Щелкнуть", чтобы кнопка NXT срабатывала при резком надавливании и ослаблении. Выберите "Нажать", чтобы кнопка срабатывала при нажатии на неё. Выберите "Отпустить", чтобы кнопка срабатывала, когда её отпускают.
3. В поле обратной связи будет отображаться "1", если выбранная кнопка NXT была нажата или опущена (в соответствии с произведенными установками).

## Настройка концентратора данных Блока «Кнопка NXT»

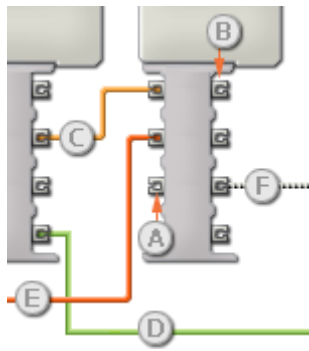
Блоком «Кнопки NXT» можно управлять динамически, подключая к его концентратору данных шины данных (от концентраторов данных других блоков).

Перетащив блок в рабочую область, откройте его концентратор данных, щёлкнув на кнопке в нижней левой части блока.



Шины данных, передающие входящие данные на блок, подключены к разъёмам, расположенным на левой стороне их концентраторов данных. Шины данных, передающие исходящие данные, подключены к разъёмам, расположенным справа.





- [A] Входной разъём
- [B] Выходной разъём
- [C] Числовая шина данных (желтая)
- [D] Логическая шина данных (зеленая)
- [E] Текстовая шина данных (оранжевая)
- [F] Повреждённая шина данных (серая)

### **Передача данных от входного разъёма на выходной разъём**

Если для входного разъёма имеется соответствующий выходной разъём (см. выше А), входящие данные будут проходить от входного разъёма на выходной разъём без изменений. В этом случае можно использовать выходной разъём, если входной разъём подключен к шине данных входа; подключение шины данных выхода к такому выходному разъёму без подключенной шины данных входа приведет к «повреждению» шины данных выхода (она станет серой).

### **По шинам данных передаются данные различных типов**

Через каждую шину данных осуществляется обмен данными различных типов между блоками. Например, если шина данных протянута с логического разъёма на концентратор данных блока, её можно подключить только к логическому разъёму на концентраторе данных другого блока. На приведенной ниже схеме показано, данные какого типа может принимать или передавать каждый из разъёмов.

### **Цвет шин данных**

Шины данных окрашены в различные цвета: шины для передачи числовых данных - желтые; шины, для передачи логических данных - зеленые; шины для передачи текстовых данных - оранжевые.

### **«Повреждённые» шины данных**

При попытке подключить шину данных к разъёму данных несоответствующего типа, шина будет повреждена (её цвет изменится на серый). Вы не сможете загрузить программу, если шина данных повреждена.

Если щёлкнуть на повреждённой шине, то в небольшом окне контекстной справки в правом нижнем углу рабочей области появится информация о причине повреждения.

### **Данные должны соответствовать возможному диапазону, в котором работает разъём**

Если шина данных входа передает значение, не попадающее в допустимый диапазон разъёма, к которому она подключена, блок либо проигнорирует такие данные, либо приведёт их к значению, попадающему в этот диапазон. Если разъём допускает ограниченное количество значений (например, только 0, 1 или 2), он проигнорирует

входящий сигнал, если передаваемое значение не будет вписываться в допустимый диапазон.

Если разъем допускает более широкий диапазон значений (например, 0 – 100), любой входящий сигнал, не попадающий в допустимый диапазон, будет приведен к значению, соответствующее ему. Например, если разъем мощности блока «Движение» получает входящий сигнал со значением 150, блок преобразует входящее значение в 100 (т.е., в числовое значение, лежащее в пределах диапазона разъема мощности ).

**На этой схеме показаны различные характеристики разъемов на концентраторе данных Блока «Кнопка NXT»:**







## Блок «Приём сообщений»

Для получения сообщения в беспроводном режиме передачи задайте тип сообщения и номер почтового ящика, они должны быть такими же, как и у NXT, с которого сообщение передаётся. Исходящий из этого блока сигнал может быть как входящим сообщением, так и логическим значением "истина/ложь" (при сопоставлении входящего и тестового сообщения).

## Настройка NXT для беспроводной связи

Перед отправкой или приёмом сообщений по беспроводной связи необходимо настроить параметры такой связи для NXT-абонентов.

На главном NXT, с которого устанавливается связь с остальными NXT, выберите Bluetooth, затем «Searching» (Поиск), в результате чего начнется поиск других устройств. Когда поиск завершится, выберите номер соединения (1, 2 или 3), который вы хотели бы присвоить другому, подчинённому NXT. При установке первого соединения между двумя NXT на дисплеях обоих устройств появится запрос на ввод Пароля; его необходимо ввести в оба NXT. Теперь при выборе на главном NXT «Connection (Соединение)» подчинённый NXT отобразится под номером выбранного соединения. На подчинённом NXT главный NXT должен отображаться под номером соединения 0. Главный NXT готов к установке связи с подчинённым NXT.

Если нужно, чтобы главный NXT связывался с несколькими другими NXT (не более, чем с тремя), необходимо снова произвести описанные выше настройки.

Настраивать нужно только один - главный NXT. Подчинённый NXT автоматически обновит соединение, когда главный NXT устанавливает номер соединения.

### Номер соединения

Для каждого NXT номер соединения - это его «адрес» в беспроводной сети. Для отправки сообщения на конкретный NXT необходимо указать его номер соединения.

## Номер почтового ящика

У каждого NXT имеется десять почтовых ящиков, в которых можно сохранять сообщения, переданные по беспроводной связи. Каждый почтовый ящик вмещает до пяти сообщений. Если ящик заполнен, т.е. уже содержит пять сообщений, то при приёме следующего сообщения NXT удалит самое раннее.

## Приём сообщения

Для приёма сообщения по беспроводной связи (после настройки номеров соединения на всех NXT) сначала нужно указать формат входящего сообщения (текст, число или логическое значение).

Чтобы проверить, принято ли конкретное сообщение (и чтобы Блок «Приём сообщений» выдал сигнал «истина»), введите в поле контрольный текст или число, либо проверьте сигнал: «истина» или «ложь». Если входящее сообщение совпадает с контрольным, то блок выдаёт сигнал «истина» через разъём «√/х».

Для завершения настройки блока выберите номер почтового ящика, в котором будут сохраняться входящие сообщения.



Примечание: дополнительные сведения об отправке сообщения по беспроводной связи см. в разделе «Блок «Приём сообщений»».

## Отображение параметров настройки блока



1. После того, как блок встроился в цепочку программы, автоматически открывается концентратор данных. В зависимости от формата входящего сообщения, оно будет отправлено через один из трех разъемов концентратора данных: текст - через разъем «Т», число - — через разъем «#», логическое значение - — через разъем «√/х».

Кроме того, если требуется сравнить входящее сообщение со своим контрольным сообщением, через разъем «√/х» можно отправить логический сигнал (истина/ложь); (Дополнительную информацию см. в разделе "Концентратор данных".)

## Настройка Блока «Приём сообщений»



1. В списке выберите тип принимаемого сообщения (текст, число или логическое значение).
2. Если требуется сравнить входящее сообщение с контрольным, введите контрольный текст или число (если был выбран текстовый или числовой формат соответственно) или кнопками выбора задайте контрольное логическое значение (истина или ложь).

3. Выберите номер почтового ящика, в котором будут сохраняться входящие сообщения.

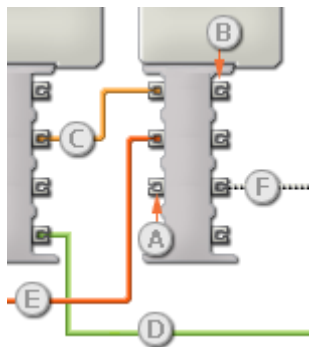
## Настройка концентратора данных Блока «Приём сообщений»

Блоком «Приём сообщений» можно управлять динамически, подключая к его концентратору данных шины данных (от концентраторов данных других блоков).

Перетащив блок в рабочую область, откройте его концентратор данных, щёлкнув на кнопке в нижней левой части блока.



Шины данных, передающие входящие данные на блок, подключены к разъёмам, расположенным на левой стороне их концентраторов данных. Шины данных, передающие исходящие данные, подключены к разъёмам, расположенным справа.



- [A] Входной разъём
- [B] Выходной разъём
- [C] Числовая шина данных (желтая)
- [D] Логическая шина данных (зеленая)
- [E] Текстовая шина данных (оранжевая)
- [F] Повреждённая шина данных (серая)

### Передача данных от входного разъёма на выходной разъём

Если для входного разъёма имеется соответствующий выходной разъём (см. выше A), входящие данные будут проходить от входного разъёма на выходной разъём без изменений. В этом случае можно использовать выходной разъём, если входной разъём подключен к шине данных входа; подключение шины данных выхода к такому выходному разъёму без подключенной шины данных входа приведет к «повреждению» шины данных выхода (она станет серой).

### По шинам данных передаются данные различных типов

Через каждую шину данных осуществляется обмен данными различных типов между блоками. Например, если шина данных протянута с логического разъёма на концентратор данных блока, её можно подключить только к логическому разъёму на концентраторе данных другого блока. На приведенной ниже схеме показано, данные какого типа может принимать или передавать каждый из разъёмов.

### Цвет шин данных

Шины данных окрашены в различные цвета: шины для передачи числовых данных - желтые; шины, для передачи логических данных - зеленые; шины для передачи текстовых данных - оранжевые.

### **«Повреждённые» шины данных**

При попытке подключить шину данных к разъёму данных несоответствующего типа, шина будет повреждена (ее цвет изменится на серый). Вы не сможете загрузить программу, если шина данных повреждена.

Если щёлкнуть на повреждённой шине, то в небольшом окне контекстной справки в правом нижнем углу рабочей области появится информация о причине повреждения.

### **Данные должны соответствовать возможному диапазону, в котором работает разъём**

Если шина данных входа передает значение, не попадающее в допустимый диапазон разъёма, к которому она подключена, блок либо проигнорирует такие данные, либо приведёт их к значению, попадающему в этот диапазон. Если разъём допускает ограниченное количество значений (например, только 0, 1 или 2), он проигнорирует входящий сигнал, если передаваемое значение не будет вписываться в допустимый диапазон.

Если разъём допускает более широкий диапазон значений (например, 0 – 100), любой входящий сигнал, не попадающий в допустимый диапазон, будет приведен к значению, соответствующее ему. Например, если разъём мощности блока «Движение» получает входящий сигнал со значением 150, блок преобразует входящее значение в 100 (т.е., в числовое значение, лежащее в пределах диапазона разъёма мощности ).

## **Блок «Датчик оборотов»**

Этот блок подсчитывает количество градусов (в одном полном обороте 360 градусов) или количество полных оборотов вала мотора. По шинам данных Блок «Датчик оборотов» может передавать текущее количество градусов или оборотов, совершённых валом мотора и логический сигнал ("истина" или "ложь"), в зависимости от того, больше или меньше оборотов совершил мотор, по сравнению с заданным пороговым значением .

Пороговое значение – это заданное значение, лежащее в диапазоне изменения измеряемого параметра. Например, можно запрограммировать робота, чтобы он остановился, когда число оборотов достигнет 10. В этом случае пороговое значение будет равно 10.

Задайте пороговое значение движком, или введите его в поле. Кнопками выбора укажите диапазон (выше или ниже порогового значения ), в котором будет выдаваться сигнал "истина", или выберите его в открывающемся списке.

По умолчанию Блок «Датчик Оборотов» выдаёт сигнал "истина" после того, как вал мотора повернётся более, чем на 360 градусов.

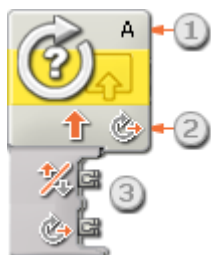
Поле обратной связи на панели настройки показывает текущее количество градусов или оборотов, которые совершил вал мотора. По умолчанию показания представлены в градусах. Если для параметра "Отработать" выбраны "Обороты", текущее значение будет отображаться в полных оборотах. Чтобы сбросить показания, нажмите кнопку Сброс. (Получить обратный сигнал можно лишь тогда, когда датчик подключен к выбранному порту и установлена связь с NXT).



## Шины данных

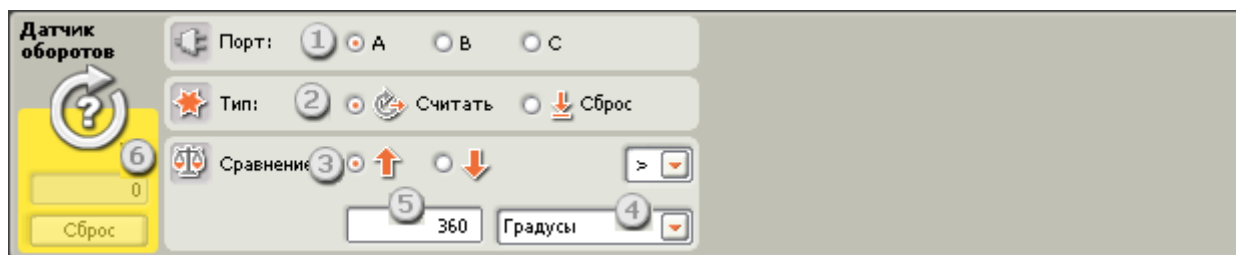
Чтобы Блок «Датчик оборотов» мог функционировать, необходимо перетащить шину данных с его концентратора данных на концентратор данных другого блока. С концентратора данных Блока «Датчик оборотов» можно перетащить шины данных двух типов: логическую шину данных, по которой будут передаваться сигналы истина/ложь, и числовую шину данных, по которой будет передаваться информация о текущем количестве оборотов или градусов. Пороговое значение также можно задать динамически, подсоединив шины входных данных от других блоков. (Дополнительную информацию см. в разделе "Концентратор данных".)

## Отображение параметров настройки блока



1. Эта буква указывает на то, какой порт будет контролироваться блоком Блок «Датчик оборотов». Этот порт при необходимости можно изменить на панели настройки.
2. Эта пиктограмма показывает, настроен ли блок на подсчёт количества оборотов (сплошная круговая стрелка) или количества градусов (пунктирная круговая стрелка).
3. Концентратор данных блока откроется автоматически, позволяя вытянуть логическую шину данных (истина/ложь) или числовую шину данных, по которой будут передаваться текущие показания датчика оборотов.

## Настройка Блока «Датчик оборотов»



1. Выберите порт, который будет контролироваться блоком (А, В или С).
2. Выберите задание для блока: подсчитывать текущие показания датчика оборотов или сбросить их.
3. Кнопками выбора задайте направление вращения мотора: вперед или назад.

Важно: удостоверьтесь, что направление вращения выбрано правильно, иначе блок не сможет корректно вести отсчёт относительно заданного порогового значения.

4. В открывающемся списке выберите единицу измерения - обороты или градусы.
5. Введите пороговое значение в поле и выберите в открывающемся списке участок диапазона для сигнала "истина", который может быть больше или меньше порогового значения.
6. В поле обратной связи отображается текущее количество полных оборотов или градусов. Кнопка "Сброс" служит для обнуления показаний.

Цифры в поле обратной связи (количество оборотов или градусов) будут чёрными при движении вперед (положительное направление) и красными при движении назад (отрицательное направление).

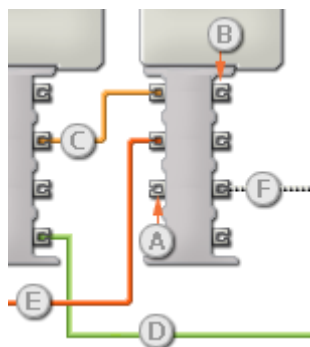
# Настройка концентратора данных Блока «Датчик оборотов»

Блоком «Датчик Оборотов» можно управлять динамически, подключая к его концентратору данных шины данных (от концентраторов данных других блоков).

Перетаскив блок в рабочую область, откройте его концентратор данных, щёлкнув на кнопке в нижней левой части блока.



Шины данных, передающие входящие данные на блок, подключены к разъёмам, расположенным на левой стороне их концентраторов данных. Шины данных, передающие исходящие данные, подключены к разъёмам, расположенным справа.



- [A] Входной разъём
- [B] Выходной разъём
- [C] Числовая шина данных (желтая)
- [D] Логическая шина данных (зеленая)
- [E] Текстовая шина данных (оранжевая)
- [F] Повреждённая шина данных (серая)

## Передача данных от входного разъёма на выходной разъём

Если для входного разъёма имеется соответствующий выходной разъём (см. выше A), входящие данные будут проходить от входного разъёма на выходной разъём без изменений. В этом случае можно использовать выходной разъём, если входной разъём подключен к шине данных входа; подключение шины данных выхода к такому выходному разъёму без подключенной шины данных входа приведет к «повреждению» шины данных выхода (она станет серой).

## По шинам данных передаются данные различных типов

Через каждую шину данных осуществляется обмен данными различных типов между блоками. Например, если шина данных протянута с логического разъёма на концентратор данных блока, её можно подключить только к логическому разъёму на концентраторе данных другого блока. На приведенной ниже схеме показано, данные какого типа может принимать или передавать каждый из разъёмов.

## Цвет шин данных

Шины данных окрашены в различные цвета: шины для передачи числовых данных - желтые; шины, для передачи логических данных - зеленые; шины для передачи текстовых данных - оранжевые.

### **«Повреждённые» шины данных**

При попытке подключить шину данных к разъёму данных несоответствующего типа, шина будет повреждена (ее цвет изменится на серый). Вы не сможете загрузить программу, если шина данных повреждена.

Если щёлкнуть на повреждённой шине, то в небольшом окне контекстной справки в правом нижнем углу рабочей области появится информация о причине повреждения.

### **Данные должны соответствовать возможному диапазону, в котором работает разъём**

Если шина данных входа передает значение, не попадающее в допустимый диапазон разъёма, к которому она подключена, блок либо проигнорирует такие данные, либо приведёт их к значению, попадающему в этот диапазон. Если разъём допускает ограниченное количество значений (например, только 0, 1 или 2), он проигнорирует входящий сигнал, если передаваемое значение не будет вписываться в допустимый диапазон.

Если разъём допускает более широкий диапазон значений (например, 0 – 100), любой входящий сигнал, не попадающий в допустимый диапазон, будет приведен к значению, соответствующее ему. Например, если разъём мощности блока «Движение» получает входящий сигнал со значением 150, блок преобразует входящее значение в 100 (т.е., в числовое значение, лежащее в пределах диапазона разъёма мощности ).

## **Блок «Датчик Звука»**

Этот Блок является детектором звукового сигнала. По шинам данных Блок «Датчик Звука» может передавать текущее показание датчика звука и логическую команду ("истина" или "ложь"), в зависимости от того превышает текущее показание пороговое значение, или ниже его.

«Пороговое значение» - это определённое значение из диапазона, в котором происходит изменение условий. Например, можно запрограммировать робота на движение вперёд, если уровень громкости звука превысит 60%. Пороговое значение будет равно 60.

Движком установите пороговое значение, или введите его в поле. Кнопками выбора укажите диапазон (выше или ниже порогового значения ), в котором будет выдаваться сигнал "истина", или выберите его в открывающемся списке. Участок диапазона, в котором выдаётся сигнал "истина", выделен цветом, а участок генерирования сигнала "ложь" будет серым.

По умолчанию Блок «Датчик Звука» настроен на выдачу команды "истина" при уровне громкости, превышающем 50%. Активирована правая кнопка выбора, а движок установлен на значение 50. Чтобы переключить диапазоны, и команда "истина" выдавалась при уровне громкости ниже 50%, нужно выбрать левую кнопку.

Поле обратной связи на панели настройки показывает текущие показания датчика звука. (Получить отражённый сигнал можно лишь тогда, когда датчик подключен к выбранному порту и установлена связь с NXT.)

Для передачи любой информации, необходимо перетащить хотя бы одну выходную шину данных от концентратора данных этого блока на другой блок. (Дополнительную информацию см. в разделе "Концентратор данных".)

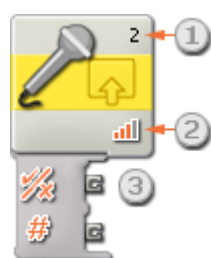


Совет: как задать пороговое значение для датчика звука?

Если нужно, чтобы Блок «Датчик Звука» выдавал сигнал "истина" при хлопке в ладоши, хлопните несколько раз, обращая внимание на текущее показание датчика звука в поле обратной связи. Если уровень громкости при хлопке достиг примерно 80%, установите пороговое значение на 70. Тогда при регистрации звуков, громкость которых превышает 70% (например, от хлопка в ладоши), Блок «Датчик Звука» будет выдавать логическую команду "истина". Более тихие звуки будут проигнорированы.

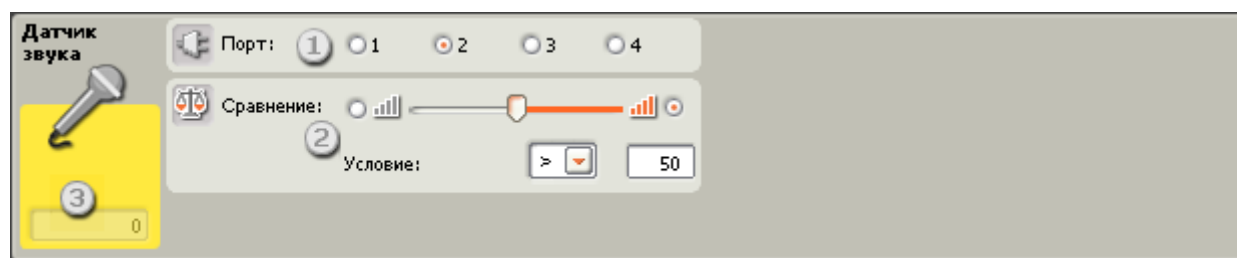
Пороговое значение можно задать динамически, при помощи шины данных входа.

## Отображение параметров настройки блока



1. Число указывает на порт NXT, к которому подключен датчик звука. При необходимости это число можно изменить в панели настройки.
2. Эта пиктограмма показывает, при каком уровне громкости Блок «Датчик Звука» среагирует. Чем больше на пиктограмме окрашенных полосок, тем для более высокого уровня громкости задано пороговое значение.
3. Когда программный блок перетащен в рабочую область, его концентратор данных открывается автоматически. Хотя бы одну шину данных следует перетащить с выходного разъёма этого блока на концентратор данных другого блока. (Дополнительную информацию см. в разделе "Концентратор данных".)

## Настройка Блока «Датчик Звука»



1. Выберите порт, к которому подключен датчик звука. По умолчанию для датчика звука блоку будет назначен порт 2. При необходимости эту настройку можно изменить.
2. Пороговое значение можно задать движком или ввести нужное значение в поле. Активируйте кнопку выбора справа от движка, чтобы блок срабатывал при уровнях громкости, превышающих пороговое значение; активируйте левую кнопку выбора, чтобы блок срабатывал при уровнях громкости более низких, чем пороговое значение. Чтобы установить диапазон подачи команды "истина", можно воспользоваться открывающимся списком.
3. В поле обратной связи отображаются текущие показания Датчика звука (0-100%). Этим можно воспользоваться для опробования различных пороговых значений.

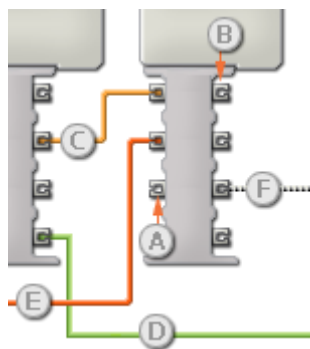
# Настройка концентратора данных Блока «Датчик Звука»

Блоком «Датчик Звука» можно управлять динамически, подключая к его концентратору данных шины данных (от концентраторов данных других блоков).

Перетаскив блок в рабочую область, откройте его концентратор данных, щёлкнув на кнопке в нижней левой части блока.



Шины данных, передающие входящие данные на блок, подключены к разъёмам, расположенным на левой стороне их концентраторов данных. Шины данных, передающие исходящие данные, подключены к разъёмам, расположенным справа.



- [A] Входной разъём
- [B] Выходной разъём
- [C] Числовая шина данных (желтая)
- [D] Логическая шина данных (зеленая)
- [E] Текстовая шина данных (оранжевая)
- [F] Повреждённая шина данных (серая)

## Передача данных от входного разъёма на выходной разъём

Если для входного разъёма имеется соответствующий выходной разъём (см. выше A), входящие данные будут проходить от входного разъёма на выходной разъём без изменений. В этом случае можно использовать выходной разъём, если входной разъём подключен к шине данных входа; подключение шины данных выхода к такому выходному разъёму без подключенной шины данных входа приведет к «повреждению» шины данных выхода (она станет серой).

## По шинам данных передаются данные различных типов

Через каждую шину данных осуществляется обмен данными различных типов между блоками. Например, если шина данных протянута с логического разъёма на концентратор данных блока, её можно подключить только к логическому разъёму на концентраторе данных другого блока. На приведенной ниже схеме показано, данные какого типа может принимать или передавать каждый из разъёмов.

## Цвет шин данных

Шины данных окрашены в различные цвета: шины для передачи числовых данных - желтые; шины, для передачи логических данных - зеленые; шины для передачи текстовых данных - оранжевые.

### **«Повреждённые» шины данных**

При попытке подключить шину данных к разъёму данных несоответствующего типа, шина будет повреждена (ее цвет изменится на серый). Вы не сможете загрузить программу, если шина данных повреждена.

Если щёлкнуть на повреждённой шине, то в небольшом окне контекстной справки в правом нижнем углу рабочей области появится информация о причине повреждения.

### **Данные должны соответствовать возможному диапазону, в котором работает разъём**

Если шина данных входа передает значение, не попадающее в допустимый диапазон разъёма, к которому она подключена, блок либо проигнорирует такие данные, либо приведёт их к значению, попадающему в этот диапазон. Если разъём допускает ограниченное количество значений (например, только 0, 1 или 2), он проигнорирует входящий сигнал, если передаваемое значение не будет вписываться в допустимый диапазон.

Если разъём допускает более широкий диапазон значений (например, 0 – 100), любой входящий сигнал, не попадающий в допустимый диапазон, будет приведен к значению, соответствующее ему. Например, если разъём мощности блока «Движение» получает входящий сигнал со значением 150, блок преобразует входящее значение в 100 (т.е., в числовое значение, лежащее в пределах диапазона разъёма мощности ).

## **Блок "Датчик температуры"**

Используя шины данных, этот блок может передавать текущее значение температуры и логический сигнал ("истина" или "ложь") в зависимости от того, выше или ниже порогового значения находится текущее значение температуры.

Вы можете изменить единицы измерения температуры по умолчанию с градусов Цельсия на градусы Фаренгейта (или наоборот) на панели настройки.

Пороговое значение должно лежать в рабочем диапазоне изменений температуры для данных условий. Например, вы можете запрограммировать своего робота, чтобы он выводил на дисплей сообщение "Холодно", если температура опускается ниже 15 °C. В этом случае пороговым значением будет 15.

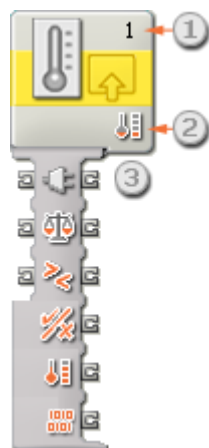
Укажите пороговое значение, воспользовавшись движком или введя число в поле ввода. Чтобы указать диапазон, в котором будет генерироваться сигнал "истина", используйте селективные кнопки или выпадающее меню. Участок диапазона, генерирующий сигнал "истина", будет цветным, участок генерирования сигнала "ложь" будет серым.

В настройках по умолчанию для блока датчика температуры установлено, что сигнал "истина" генерируется при значениях температуры ниже 25 °C. Выбрана селективная кнопка в левом участке диапазона, движок установлен на 25 °C. Чтобы переключить участки диапазона, соответствующие сигналу "истина" (установить как "истинные" значения выше 25 °C), нужно выбрать правую селективную кнопку.

Блок обратной связи на панели настройки показывает текущее значение температуры в градусах Цельсия. (Чтобы гарантировать получение отклика, убедитесь в том, что датчик подключен к выбранному порту и что установлена связь с блоком NXT.)

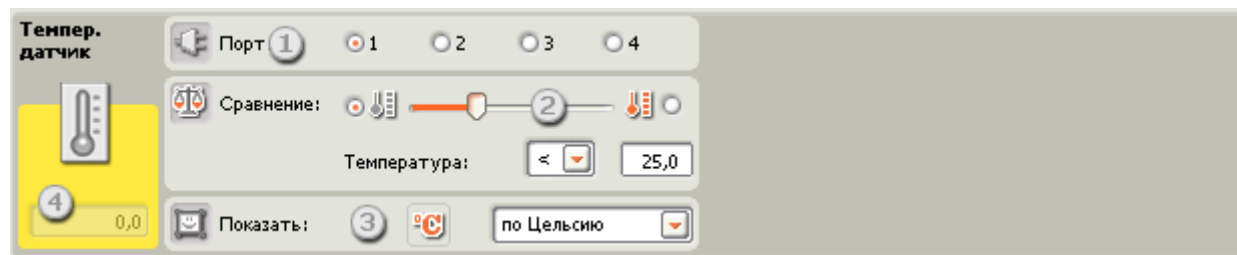
Для отправки информации вы должны перетащить хотя бы одну шину выходных данных с концентратора данных этого блока на другой блок. (Чтобы получить дополнительную информацию, см. ниже раздел "Концентратор данных".)

## Отображение параметров настройки



1. Число показывает, какой порт NXT подключен к температурному датчику. Если необходимо, вы можете изменить это число в панели настройки.
2. Эта пиктограмма показывает уровень порогового значения. Чем больше окрашенных полосок, тем выше величина порогового значения.
3. После помещения программного блока в рабочую область концентратор данных блока откроется автоматически. Хотя бы одну шину данных следует перетащить с выходного разъема блока на концентратор данных другого блока.

## Настройка Блока «Датчик температуры»



1. Выберите порт, к которому подключен датчик температуры. По умолчанию для датчика температуры блоку будет назначен порт 1. Если необходимо, вы можете изменить эту настройку.
2. Чтобы установить пороговое значение, воспользуйтесь движком или введите значение прямо в поле ввода. Выберите селективную кнопку справа от движка, если вы хотите, чтобы блок срабатывал на температуру, более высокую, чем пороговое значение; выберите левую селективную кнопку для срабатывания блока на температуру, более низкую, чем пороговое значение. Чтобы установить диапазон подачи команды "истина", вы также можете воспользоваться ниспадающим меню.
3. Выберите шкалу измерения для регистрируемых значений - шкала Цельсия или шкала Фаренгейта.
4. В окне обратной связи выводятся результаты измерения текущей температуры (от -20 до 120 °C или от -4 до 248 °F). Если настройки сконфигурированы для градусов Цельсия, показания датчика 25 означают температуру 25 °C. Если конфигурация настроена на градусы Фаренгейта, показания датчика 77 означают температуру 77 °F.

## Настройка концентратора данных

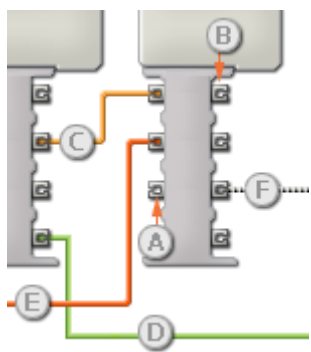


Вы можете управлять блоком температуры динамически, подключая шины данных (от концентраторов данных других блоков) к концентратору данных блока температуры.

Поместив блок в рабочую область, откройте концентратор данных блока, щелкнув по кнопке в нижней левой части блока.



Шины данных, передающие входящие данные на блок, подключены к разъемам, находящимся на левой стороне их концентраторов данных. Шины данных, передающие исходящие данные, подключены к разъемам, находящимся на правой стороне.



- [A] Входной разъем
- [B] Выходной разъем
- [C] Числовая шина данных (желтая)
- [D] Логическая шина данных (зеленая)
- [E] Текстовая шина данных (оранжевая)
- [F] Поврежденная шина данных (серая)

## Передача данных от входного разъема на выходной разъем

Если для входного разъема есть соответствующий выходной разъем (см. выше A), входящие данные будут проходить от входного разъема на выходной разъем без изменений. В этом случае вы сможете использовать выходной разъем, если входной разъем подключен к шине данных входа; подключение шины данных выхода к такому выходному разъему без подключенной шины данных входа приведет к «неисправности» шины данных выхода (она будет окрашена в серый цвет).

## Шины данных осуществляют передачу специфических типов данных

Каждая шина данных осуществляет обмен специфическими типами данных между блоками. Например, если шина данных выходит с логического разъема одного блока, ее можно подключить только к логическому разъему другого блока. На приведенной ниже схеме показано, какой вид данных может принимать или передавать каждый из разъемов.

## Окраски шин данных

Шины данных различаются особыми окрасками: шины, передающие числовые данные, имеют желтый цвет; шины, передающие логические данные, имеют зеленый цвет; шины, передающие текстовые данные, имеют оранжевый цвет.

## «Поврежденные» шины данных

Если вы пытаетесь подключить шину данных к разъему несоответствующего типа данных, шина будет повреждена (ее цвет изменится на серый). Вы не сможете загрузить вашу программу, если шина данных повреждена.

Если вы щелкнете по поврежденной шине, то в небольшом окне контекстной справки в правом нижнем углу рабочей области вы сможете прочесть информацию о причине повреждения.

### **Данные должны соответствовать возможному диапазону, в котором работает разъем**

Если шина данных входа передает значение, не попадающее в допустимый диапазон разъема, к которому она подключена, блок либо проигнорирует такие данные, либо трансформирует данные, попадающие в этот диапазон. Для разъемов, допускающих небольшое количество значений (например, только 0, 1 или 2), разъем проигнорирует входящий сигнал, если передаваемое значение не будет вписываться в диапазон разъема.

Для разъемов, допускающих более широкий диапазон значений (например, 0 – 100), разъем усилит любой входящий сигнал, если он не соответствует диапазону. Например, если разъем питания программного блока перемещения получает входящий сигнал со значением 150, блок преобразует входящее значение в 100 (т.е., в числовое значение, лежащее в пределах диапазона разъема мощности).

## **Блок «Таймер»**

Когда программа начинает выполняться, три встроенных в NXT таймера автоматически начинают отсчет времени. Используя этот блок, можно выбрать следующие действия: снимать текущие показания времени или сбросить таймер для начала нового отсчета времени с нуля.

По шинам данных этот блок может передавать текущее показание таймера и логический сигнал ("истина" или "ложь") в зависимости от того, выше или ниже порогового значения текущее показание таймера.

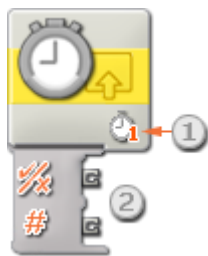
Пороговое значение – это заданное значение, лежащее в диапазоне изменения измеряемого параметра. Например, можно запрограммировать робота останавливаться, когда таймер отсчитывает последние 20 секунд. В этом случае пороговое значение будет равно 20 секундам.

Задайте пороговое значение движком, или введите его в поле. Кнопками выбора укажите диапазон (выше или ниже порогового значения), в котором будет выдаваться сигнал "истина", или выберите его в открывающемся списке.

Чтобы Блок «Таймер» работал, нужно перетащить хотя бы одну шину выходных данных с концентратора данных этого блока на концентратор данных другого блока. Из концентратора данных Блока «Таймер» можно перетаскивать шины выходных данных двух типов: шину логических данных, которая будет передавать сигнал истина/ложь, и шину числового значения, которая будет передавать текущие показания таймера.

Пороговое значение можно вводить динамически, подключив шину входных данных к концентратору данных Блока «Таймер». (Дополнительную информацию, см. в разделе "Концентратор данных".)

## **Отображение параметров настройки блока**

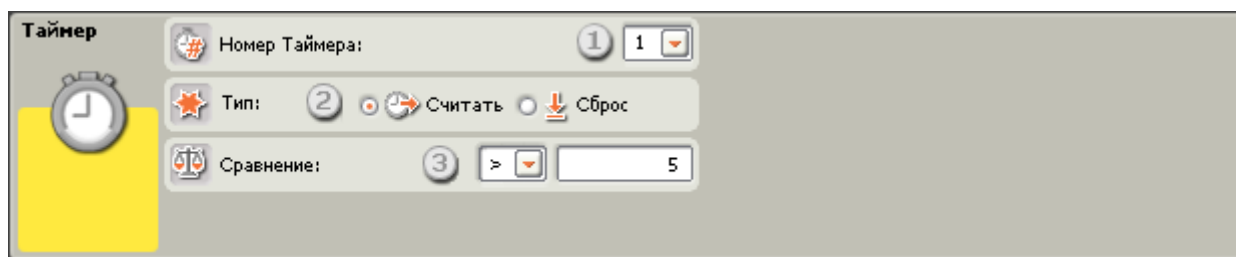


1. Это число указывает, показания какого из трех таймеров NXT будет выводиться. При необходимости, на панели настройки можно выбрать другой таймер.
2. Когда программный блок перетащен в рабочую область, его концентратор данных открывается автоматически. Хотя бы одну шину данных следует перетащить с выходного разъёма блока на концентратор данных другого блока, чтобы иметь возможность передавать текущие показания таймера. Пороговое значение также можно вводить динамически, подключая шину входных данных к концентратору данных Блока «Таймер».



Примечание: для отправки информации нужно перетащить хотя бы одну шину выходных данных с концентратора данных Блока «Таймер» на другой блок.

## Настройка Блока «Таймер»:



1. Выберите, какой из трех таймеров NXT будет отображаться и контролироваться.
2. Если выбрано "Считать", Блок «Таймер» будет отслеживать назначенный таймер, уже установленный где-либо в программе. Если был выбран "Сброс", блок обнулит назначенный таймер и снова запустит его. В дальнейшем в программу можно вставить ещё и контрольный Блок «Таймер».
3. Введите пороговое значение (в секундах) в поле, и в открывающемся списке укажите, при каких значениях должен выдаваться сигнал "истина" : больших или меньших порогового значения.

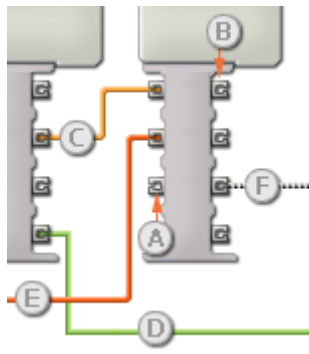
## Настройка концентратора данных Блока «Таймер»

Блоком «Таймер» можно управлять динамически, подключая к его концентратору данных шины данных (от концентраторов данных других блоков).

Перетащив блок в рабочую область, откройте его концентратор данных, щёлкнув на кнопке в нижней левой части блока.



Шины данных, передающие входящие данные на блок, подключены к разъёмам, расположенным на левой стороне их концентраторов данных. Шины данных, передающие исходящие данные, подключены к разъёмам, расположенным справа.



- [A] Входной разъём
- [B] Выходной разъём
- [C] Числовая шина данных (желтая)
- [D] Логическая шина данных (зеленая)
- [E] Текстовая шина данных (оранжевая)
- [F] Повреждённая шина данных (серая)

## Передача данных от входного разъёма на выходной разъём

Если для входного разъёма имеется соответствующий выходной разъём (см. выше А), входящие данные будут проходить от входного разъёма на выходной разъём без изменений. В этом случае можно использовать выходной разъём, если входной разъём подключен к шине данных входа; подключение шины данных выхода к такому выходному разъёму без подключенной шины данных входа приведет к «повреждению» шины данных выхода (она станет серой).

## По шинам данных передаются данные различных типов

Через каждую шину данных осуществляется обмен данными различных типов между блоками. Например, если шина данных протянута с логического разъёма на концентратор данных блока, её можно подключить только к логическому разъёму на концентраторе данных другого блока. На приведенной ниже схеме показано, данные какого типа может принимать или передавать каждый из разъёмов.

## Цвет шин данных

Шины данных различаются особыми окрасками: шины, передающие числовые данные, имеют желтый цвет; шины, передающие логические данные, имеют зеленый цвет; шины, передающие текстовые данные, имеют оранжевый цвет.

## «Повреждённые» шины данных

При попытке подключить шину данных к разъёму данных несоответствующего типа, шина будет повреждена (ее цвет изменится на серый). Вы не сможете загрузить программу, если шина данных повреждена.

Если щёлкнуть на повреждённой шине, то в небольшом окне контекстной справки в правом нижнем углу рабочей области появится информация о причине повреждения.

## Данные должны соответствовать возможному диапазону, в котором работает разъём

Если шина данных входа передает значение, не попадающее в допустимый диапазон разъёма, к которому она подключена, блок либо проигнорирует такие данные, либо приведёт их к значению, попадающему в этот диапазон. Если разъём допускает ограниченное количество значений (например, только 0, 1 или 2), он проигнорирует

входящий сигнал, если передаваемое значение не будет вписываться в допустимый диапазон.

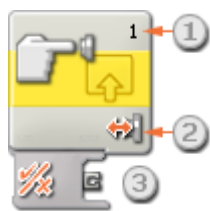
Если разъем допускает более широкий диапазон значений (например, 0 – 100), любой входящий сигнал, не попадающий в допустимый диапазон, будет приведен к значению, соответствующее ему. Например, если разъем мощности блока «Движение» получает входящий сигнал со значением 150, блок преобразует входящее значение в 100 (т.е., в числовое значение, лежащее в пределах диапазона разъема мощности ).

## Блок «Датчик касания»

Этот блок проверяет состояние датчика касания на определенном этапе выполнения программы. Он посылает зарегистрированный сигнал в виде логической команды ("истина" или "ложь") через шину данных. Если датчик сработал, блок отправит команду "истина"; если датчик не сработал, блок отправит команду "ложь".

Для того, чтобы какая-либо информация могла быть передана, необходимо перетащить шину выходных данных с концентратора данных этого блока на другой блок. (Дополнительную информацию см. в разделе "Концентратор данных".)

Используйте кнопки выбора чтобы задать действие, при выполнении которого блок будет выдавать сигнал "истина".



1. Число указывает на порт NXT, к которому подключен датчик касания. При необходимости это число можно изменить в панели настройки.
2. Эта пиктограмма показывает в результате какого действия Датчик касания выдаст команду "истина" (Щелчок, Нажатие или Освобождение).
3. Когда программный блок перетащен в рабочую область, его концентратор данных открывается автоматически. Шину данных следует перетащить с выходного разъема этого блока на концентратор данных другого блока.

## Настройка Блока «Датчик касания»



1. Выберите порт, к которому подключен датчик касания. По умолчанию, для датчика касания блоку будет назначен порт 1. При необходимости эту настройку можно изменить.
2. Существует три варианта срабатывания Блока «Датчик касания». Выберите "Щелкнуть", чтобы блок срабатывал при резком нажатии и освобождении кнопки датчика касания (длительность действия менее 0,5 секунды). Выберите "Нажать", чтобы блок срабатывал при нажатии на кнопку датчика касания. Выберите "Отпустить", чтобы блок срабатывал после освобождения кнопки датчика касания.
3. Используйте поле обратной связи для проверки датчика касания. Если датчик сработал, в этом поле отобразится «1». чтобы очистить поле после нажатия на датчик касания, нажмите "Сброс".

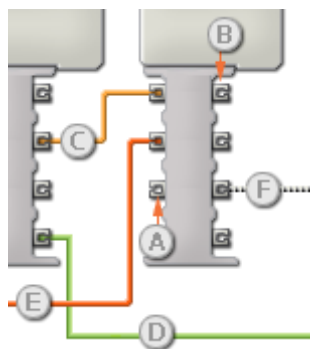
## Настройка концентратора данных Блока «Датчик касания»

Блоком «Датчик касания» можно управлять динамически, подключая шины данных (от концентраторов данных других блоков) к концентратору данных Блока «Датчик касания».

Перетащите Блок «Датчик касания» в рабочую область и откройте его концентратор данных, щёлкнув на кнопке в нижней левой части блока.



Шины данных, по которым данные поступают на блок, подключены к разъёмам, расположенным на левой стороне их концентраторов данных. Шины данных, передающие исходящие данные, подключены к разъёмам, расположенным справа.



- [A] Входной разъём
- [B] Выходной разъём
- [C] Числовая шина данных (желтая)
- [D] Логическая шина данных (зеленая)
- [E] Текстовая шина данных (оранжевая)
- [F] Повреждённая шина данных (серая)

### Передача данных от входного разъёма на выходной разъём

Если входной разъём соединён с соответствующим выходным разъёмом (см. выше A), входящие данные будут проходить от входного разъёма на выходной разъём без изменений. В этом случае можно использовать выходной разъём, если входной разъём подключен к шине данных входа; подключение шины данных выхода к такому выходному разъёму без подключенной шины данных входа приведет к «повреждению» шины данных выхода (она станет серой).

### По шинам данных передаются данные различных типов

Через каждую шину данных осуществляется обмен данными различных типов между блоками. Например, если шина данных протянута с логического разъёма на концентратор данных блока, ее можно подключить только к логическому разъёму на концентраторе данных другого блока. На приведенной ниже схеме показано, данные какого типа может принимать или передавать каждый из разъёмов.

### Цвет шин данных

Шины данных окрашены в различные цвета: шины для передачи числовых данных - желтые; шины, для передачи логических данных - зеленые; шины для передачи текстовых данных - оранжевые.

### **«Повреждённые» шины данных**

При попытке подключить шину данных к разъёму данных несоответствующего типа, шина будет повреждена (ее цвет изменится на серый). Вы не сможете загрузить программу, если шина данных повреждена.

Если щёлкнуть на повреждённой шине, то в небольшом окне контекстной справки в правом нижнем углу рабочей области появится информация о причине повреждения.

### **Данные должны соответствовать возможному диапазону, в котором работает разъём**

Если шина данных входа передает значение, не попадающее в допустимый диапазон разъёма, к которому она подключена, блок либо проигнорирует такие данные, либо приведёт их к значению, попадающему в этот диапазон. Если разъём допускает ограниченное количество значений (например, только 0, 1 или 2), он проигнорирует входящий сигнал, если передаваемое значение не будет вписываться в допустимый диапазон.

Если разъём допускает более широкий диапазон значений (например, 0 – 100), любой входящий сигнал, не попадающий в допустимый диапазон, будет приведен к значению, соответствующее ему. Например, если разъём мощности блока «Движение» получает входящий сигнал со значением 150, блок преобразует входящее значение в 100 (т.е., в числовое значение, лежащее в пределах диапазона разъёма мощности ).

## **Блок «Датчик расстояния»**

Этот блок может обнаруживать объекты на расстоянии до 250 см (100 дюймов). Используя шины данных, блок может передавать текущее значение ультразвукового сигнала и логическую команду ("истина" или "ложь") в зависимости от того, выше или ниже текущий сигнал, чем пороговое значение срабатывания.

«Пороговое значение» - это определённое значение из диапазона, в котором происходит изменение условий. Например, можно запрограммировать робота на столкновение, если он приближается к объекту ближе 76 см (30 дюймов), то есть попадает в пределы максимального диапазона действия датчика расстояния. Пороговое значение будет равно 30.

Задайте пороговое значение движком, или введите его в соответствующее поле. Чтобы задать ту часть диапазона датчика расстояния (выше или ниже порогового значения ), в которой будет выдаваться команда "истина", используйте кнопки выбора или список. Та часть диапазона, в которой выдаётся сигнал "истина", будет окрашена, а участок генерирования сигнала "ложь" будет серым.

Для Блока «Датчик расстояния» по умолчанию сигнал "истина" выдаётся при приближении к объекту менее, чем на 127 см (50 дюймов). Активирована левая кнопка выбора, а движок установлен на значение 50 (127). Чтобы переключить диапазоны, и команда "истина" выдавалась при дальности, превышающей 127 см (50 дюймов, нужно активировать правую кнопку выбора.



В поле обратной связи отображаются текущие показания датчика расстояния. (Получить отражённый сигнал можно лишь тогда, когда датчик подключен к выбранному порту и установлена связь с NXT.)

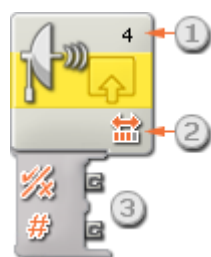
Для передачи любой информации, необходимо перетащить хотя бы одну выходную шину данных от концентратора данных этого блока на другой блок.



Совет: как установить пороговое значение для датчика расстояния?

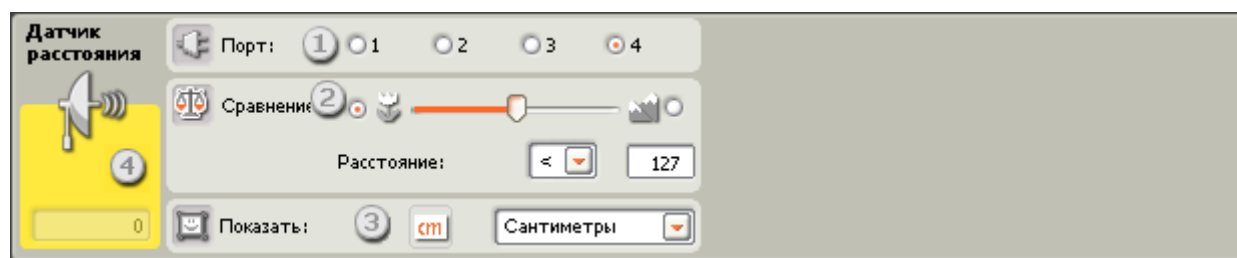
Если вы хотите, чтобы при приближении робота противника к вашему роботу выдавалась команда "истина", направляйте свой датчик расстояния на робота-противника (или на иной объект с такой же отражающей способностью), когда он находится на различных расстояниях от датчика. Наблюдайте за показаниями датчика расстояния в поле обратной связи. Если ваш датчик «засёк» робота-противника на расстоянии около 127 см (50 дюймов), и вы не хотите, чтобы он подъехал к вашему роботу ещё ближе, вы можете установить пороговое значение на 50 (127). Теперь, если объект с такой же отражательной способностью окажется ближе 127 см (50 дюймов), ваш Блок «Датчик расстояния» выдаст команду "истина". Подключив шину данных к Блоку «Датчик расстояния» или к Блоку «Мотор», вы сможете начать атаку или отступление.

## Отображение параметров настройки блока



1. Число указывает к какому из портов NXT подключен датчик расстояния. При необходимости это число можно изменить в панели настройки.
2. Эта пиктограмма показывает как настроен датчик расстояния: на обнаружение близких или удаленных объектов. Чем больше цветных полосок, тем длиннее дистанция обнаружения.
3. Когда блок перетащен в рабочую область, его концентратор данных открывается автоматически. Хотя бы одну шину данных следует протянуть с выходного разъёма данного блока на концентратор данных другого блока. Пороговое значение можно задать динамически, если подключить шину данных входа. (Дополнительную информацию см. в разделе "Концентратор данных".)

## Настройка Блока «Датчик расстояния»



1. Выберите порт, к которому подключен датчик расстояния. По умолчанию для него назначается порт 4. При необходимости эту настройку можно изменить.
2. Если активировать левую (от движка) кнопку выбора, блок запустится при регистрации объекта, находящегося на расстоянии меньшем, чем пороговое значение; активируйте правую кнопку выбора, чтобы запускать блок при регистрации объекта на расстоянии большем, чем пороговое значение. Установите пороговое значение ползунком или введите непосредственно в поле ввода (0–250 , для

измерения в сантиметрах, или 0-100 для дюймов). Учитывайте то, что объекты с высокой отражающей способностью можно обнаружить на большем расстоянии, чем слабо отражающие объекты.

3. Выберите единицу измерения регистрируемых значений - сантиметры или дюймы.
4. Поле обратной связи отображает текущие показания датчика расстояния (0-250 см, или 0-100 дюймов). Значение "0" соответствует наименьшему расстоянию, которое может зарегистрировать датчик. Если выбраны сантиметры, цифра 250 означает максимальное расстояние, на котором датчик может работать (т.е., приблизительно, 250 см). Если выбраны дюймы, цифра 100 означает максимальное расстояние, на котором датчик может работать (т.е., приблизительно, 100 дюймов).



Примечание: если в одном помещении работают несколько (более одного) датчиков расстояния, их показания могут взаимно искажаться.

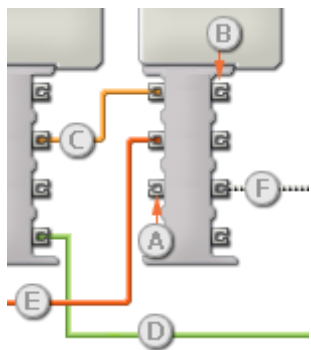
## Настройка концентратора данных Блока «Датчик расстояния»

Блоком «Датчик расстояния» можно управлять динамически, подключая шины данных (от концентраторов данных других блоков) к концентратору данных Блока «Датчик расстояния».

Перетащите блок в рабочую область и откройте его концентратор данных, щёлкнув на кнопке в нижней левой части блока.



Шины данных, передающие входящие данные на блок, подключены к разъёмам, расположенным на левой стороне их концентраторов данных. Шины данных, передающие исходящие данные, подключены к разъёмам, расположенным справа.



- [A] Входной разъём
- [B] Выходной разъём
- [C] Числовая шина данных (желтая)
- [D] Логическая шина данных (зеленая)
- [E] Текстовая шина данных (оранжевая)
- [F] Повреждённая шина данных (серая)

### Передача данных от входного разъёма на выходной разъём

Если для входного разъёма имеется соответствующий выходной разъём (см. выше A), входящие данные будут проходить от входного разъёма на выходной разъём без изменений. В этом случае можно использовать выходной разъём, если входной разъём подключен к шине данных входа; подключение шины данных выхода к такому

выходному разъёму без подключенной шины данных входа приведет к «повреждению» шины данных выхода (она станет серой).

## **По шинам данных передаются данные различных типов**

Через каждую шину данных осуществляется обмен данными различных типов между блоками. Например, если шина данных протянута с логического разъёма на концентратор данных блока, её можно подключить только к логическому разъёму на концентраторе данных другого блока. На приведенной ниже схеме показано, данные какого типа может принимать или передавать каждый из разъёмов.

## **Цвет шин данных**

Шины данных окрашены в различные цвета: шины для передачи числовых данных - желтые; шины, для передачи логических данных - зеленые; шины для передачи текстовых данных - оранжевые.

## **«Повреждённые» шины данных**

При попытке подключить шину данных к разъёму данных несоответствующего типа, шина будет повреждена (ее цвет изменится на серый). Вы не сможете загрузить программу, если шина данных повреждена.

Если щёлкнуть на повреждённой шине, то в небольшом окне контекстной справки в правом нижнем углу рабочей области появится информация о причине повреждения.

## **Данные должны соответствовать возможному диапазону, в котором работает разъём**

Если шина данных входа передает значение, не попадающее в допустимый диапазон разъёма, к которому она подключена, блок либо проигнорирует такие данные, либо приведёт их к значению, попадающему в этот диапазон. Если разъём допускает ограниченное количество значений (например, только 0, 1 или 2), он проигнорирует входящий сигнал, если передаваемое значение не будет вписываться в допустимый диапазон.

Если разъём допускает более широкий диапазон значений (например, 0 – 100), любой входящий сигнал, не попадающий в допустимый диапазон, будет приведен к значению, соответствующее ему. Например, если разъём мощности блока «Движение» получает входящий сигнал со значением 150, блок преобразует входящее значение в 100 (т.е., в числовое значение, лежащее в пределах диапазона разъёма мощности ).

Блоки группы Операторы

- [Блок «Цикл»](#)
- [Блок «Стоп»](#)
- [Блок «Переключатель»](#)
- [Блок «Ожидание»](#)

## **Блок «Цикл»**

Этот блок используется для повтора последовательностей команд. Назначьте условие, необходимое для завершения цикла: истекшее время, количество повторов, логическая команда или сигнал от датчика. Также можно задать бесконечное повторение цикла.

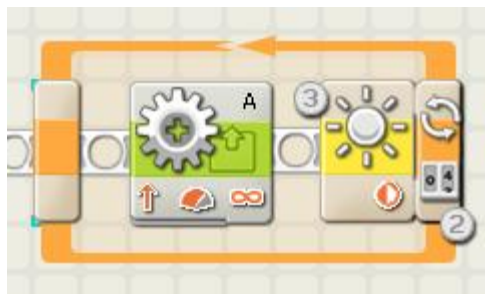
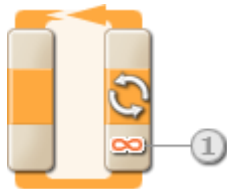
[Отображение параметров настройки блока](#)

[Добавление программных блоков в Блок «Цикл»](#)

[Перемещение Блока «Цикл»](#)

[Настройка Блока «Цикл»](#)

## Отображение параметров настройки блока



1. Если в секции Управления Блоком «Цикл» выбрано условие “Постоянно”, справа внизу блока отображается знак бесконечности ( $\infty$ ).
2. Если на панели настроек установлен флажок “Показать счётчик”, появится разъём, который позволит зацикливать программу на любом её участке на заданное количество завершённых циклов (если подключить разъём к концентратору данных другого блока через шину данных). Количество завершённых на данный момент циклов можно использовать для управления самим циклом. (См. ниже " Подсчёт" в разделе "Настройка Блока «Цикл»").
3. Если для управления циклом назначается датчик, Блок «Цикл» растягивается вправо, открывая пиктограмму выбранного датчика. Например, если для управления циклом назначен Датчик освещённости, в расширившейся части программного блока появится его пиктограмма. А в нижней части программного блока будет отображена соответствующая информация о выбранном способе управления.

## Добавление программных блоков в Блок «Цикл»

Если Блок «Цикл» вставлен в цепочку программы, её небольшой участок окажется внутри Блока «Цикл»; с ним будут связаны программные блоки, перетаскиваемые в верхнюю часть этой области цепочки. С каждым новым блоком, перетаскиваемым внутрь цикла, который уже содержит программные блоки, область цикла будет расширяться. За счёт этого создаётся пространство для новых блоков, которые будут встраиваться в цепочку программы.

## Перемещение Блока «Цикл»

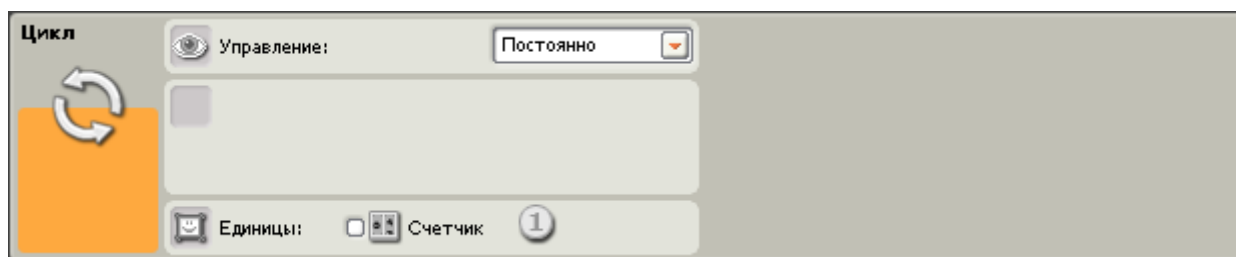
Чтобы переместить Блок Цикл со всеми входящими в него Блоками, щёлкните именно на Блоке Цикл, чтобы его выделить, а затем перетащите на новое место.

## Настройка Блока «Цикл»

В списке управления можно условие работы цикла:

- [Постоянно \[по умолчанию\]](#)
- [Время](#)
- Датчик (появляется новое поле со списком доступных датчиков)
  - [Касания](#)
  - [Освещённости](#)
  - [Звука](#)
  - [Расстояния](#)
  - [Кнопка NXT](#)
  - [Оборотов](#)
  - [Таймер](#)
  - [Принять сообщение](#)
  - [Температуры](#)
- [Подсчёт](#)
- [Логика](#)

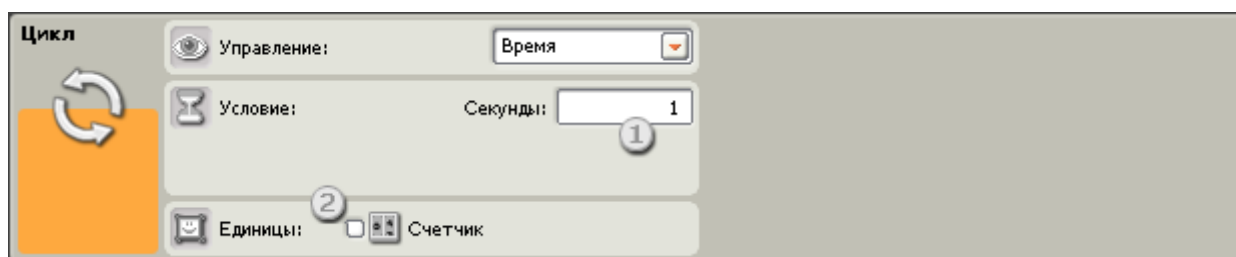
## Постоянно



Если выбрать “Постоянно,” действия всех программных блоков внутри цикла будут бесконечно и безостановочно повторяться.

1. Установите флажок “Счётчик”, чтобы использовать количество завершённых циклов в качестве входа другого блока. (Пример: увеличение мощности мотора пропорционально количеству циклов.)

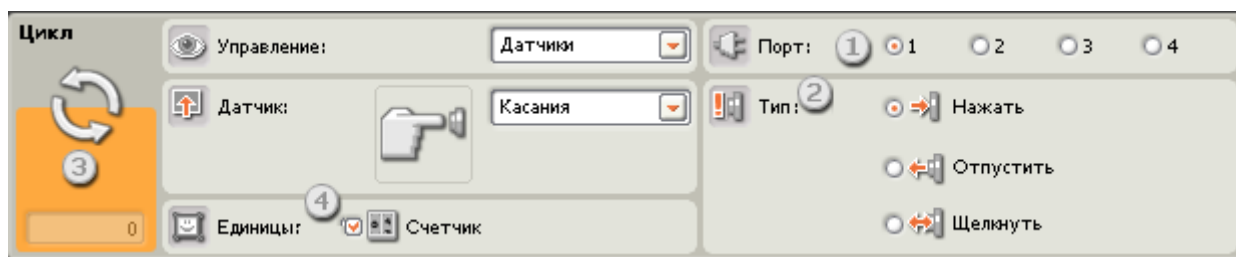
## Время



Если выбрать “Время,” программные блоки внутри цикла будут повторять свои действия до тех пор, пока не пройдет определенное количество секунд. По истечении заданного периода времени цикл завершится.

1. Введите в это окно то количество секунд, в течение которых цикл должен будет работать. Например, если выбрать 5 секунд, цикл завершится по истечении 5 секунд.
2. Если на панели настроек установлен флажок “Счётчик”, появится разъём, который позволит заикливать программу на любом её участке на заданное количество завершённых циклов (если подключить разъём к концентратору данных другого блока через шину данных).

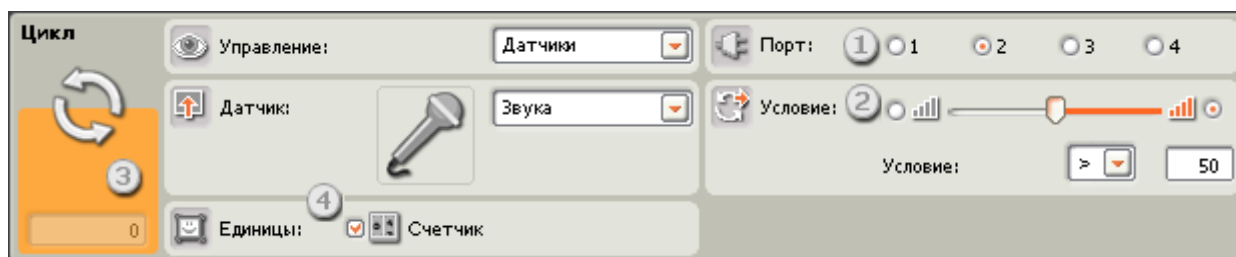
## Датчик касания



Если выбрать “Датчик касания,” программные блоки внутри цикла будут повторять свои действия до тех пор, пока датчик касания не зарегистрирует щелчок, нажатие или освобождение. После этого цикл завершит свою работу, а выполнение программы продолжится.

1. Выберите порт, к которому подключен датчик касания.
2. Кнопками выбора задайте щелчок, нажатие или освобождение датчика касания, в результате чего выполнение цикла должно прекратиться. Выберите "Щелчок", чтобы блок срабатывал при щелчке по кнопке датчика касания (длительность воздействия менее 0,5 с). Выберите "Нажать", чтобы блок срабатывал при нажатии на кнопку датчика касания. Выберите "Отпустить", чтобы блок срабатывал при освобождении кнопки датчика касания.
3. В поле обратной связи можно проверить состояние датчика касания. Если датчик работа активирован, в нём появится цифра “1”.
4. Если на панели настроек установлен флажок “Счётчик”, появится разъём, который позволит зацикливать программу на любом её участке на заданное количество завершённых циклов (если подключить разъём к концентратору данных другого блока через шину данных).

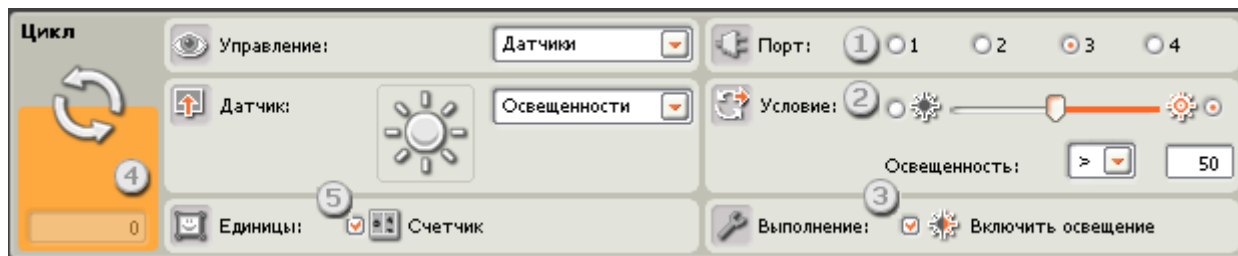
### Датчик звука



Если выбрать “Датчик звука,” программные блоки внутри цикла будут повторять свои действия до тех пор, пока датчик звука не зарегистрирует звук определенной громкости.

1. Выберите порт, к которому подключен датчик звука. По умолчанию для датчика звука блоку будет назначен порт 2.
2. Пороговое значение можно задать движком или ввести нужное значение в поле. Активируйте кнопку выбора справа от движка, чтобы, выполнение цикла прекращалось, когда громкость звука превысит пороговое значение; активируйте левую кнопку выбора, чтобы цикл завершался при громкости звука ниже порогового значения. Также можно воспользоваться открывающимся списком, чтобы установить движком предел, при котором выполнение цикла будет прекращаться.
3. В поле обратной связи отображаются текущие показания датчика звука (0-100%). Этим можно воспользоваться для опробования различных пороговых значений.
4. Если на панели настроек установлен флажок “Счётчик”, появится разъём, который позволит зацикливать программу на любом её участке на заданное количество завершённых циклов (если подключить разъём к концентратору данных другого блока через шину данных).

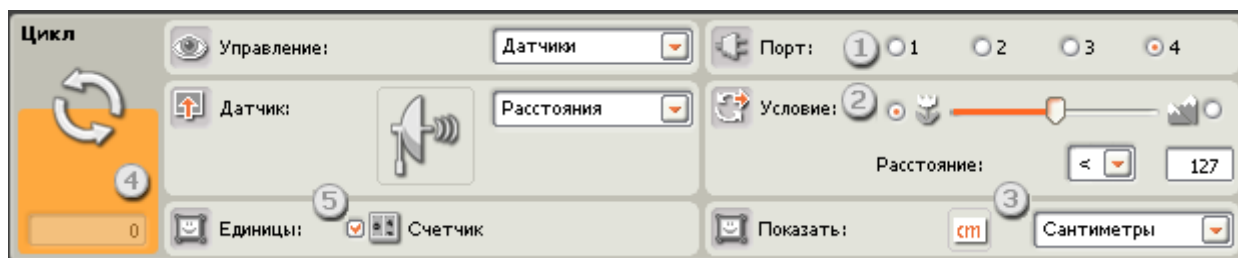
### Датчик освещённости



Если выбран “Датчик освещённости,” программные блоки внутри цикла будут повторять свои действия, пока датчик освещённости не зарегистрирует определенную яркость света. После этого выполнение цикла прекратится, а выполнение программы продолжится.

1. Выберите порт, к которому подключен датчик освещённости. По умолчанию него блоку будет назначен порт 3.
2. Пороговое значение можно задать движком или ввести нужное значение в поле. Активируйте кнопку выбора справа от движка, чтобы выполнение цикла прекращалось, когда уровень освещённости превышает пороговое значение; активируйте левую кнопку выбора, чтобы выполнение цикла прекращалось при уровнях освещённости более низких, чем пороговое значение. Также можно воспользоваться открывающимся списком, чтобы установить движком предел, при котором выполнение цикла будет прекращаться.
3. Если установлен флажок “Включить освещение”, датчик включит свой собственный слабый источник света и зарегистрирует отражённый свет.
4. В поле обратной связи будут отображаться текущие показания датчика.
5. Если на панели настроек установлен флажок “Счётчик”, появится разъём, который позволит заикливать программу на любом её участке на заданное количество завершённых циклов (если подключить разъём к концентратору данных другого блока через шину данных).

## Датчик расстояния

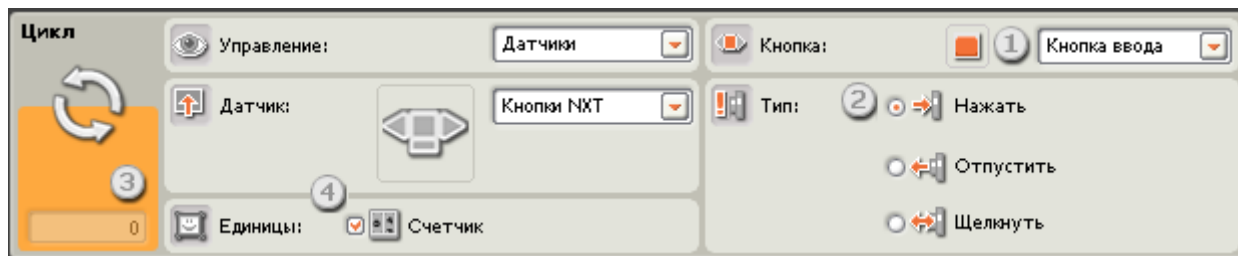


Если выбрать “Датчик расстояния,” программные блоки внутри цикла будут повторять свои действия до тех пор, пока этот датчик не обнаружит объект на заданном расстоянии.

1. Выберите порт, к которому подключен датчик расстояния. По умолчанию для него блоку будет назначен порт 4.
2. Пороговое значение можно задать движком или ввести нужное значение в поле. Активируйте кнопку выбора слева от движка, чтобы выполнение цикла прекращалось, когда датчик расстояния обнаруживает объект на расстоянии, меньшем порогового значения ; активируйте правую кнопку выбора, чтобы выполнение цикла прекращалось, когда датчик расстояния обнаруживает объект на расстоянии, превышающем пороговое значение. Также можно воспользоваться открывающимся списком, чтобы установить движком предел, при котором выполнение цикла будет прекращаться.
3. Выберите единицу измерения для регистрируемых значений - сантиметры или дюймы.
4. В поле обратной связи будут отображаться текущие показания датчика расстояния.
5. Если на панели настроек установлен флажок “Счётчик”, появится разъём, который позволит заикливать программу на любом её участке на заданное количество завершённых циклов (если подключить разъём к концентратору данных другого блока через шину данных).

## Кнопки NXT

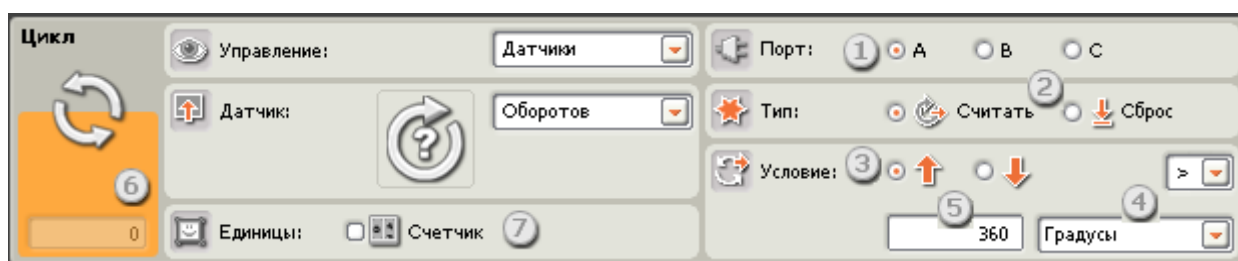




Если выбрать “Кнопки NXT,” работа программных блоков внутри цикла будет повторяться, пока с выбранными кнопками NXT не будет произведено какое-либо действие (щелчок, нажатие или освобождение).

1. Выберите кнопку NXT, воздействие на которую вызовет команду "истина" (и прекратит выполнение цикла).
2. Выберите "Щелкнуть" чтобы Блок срабатывал после щелчка по выбранной кнопке. Выберите "Нажать", чтобы Блок срабатывал срабатывал при нажатии кнопки. Выберите "Отпустить", чтобы Блок срабатывал, после отпускания кнопки.
3. В поле обратной связи появится “1”, если с выбранной кнопкой NXT было произведено какое-либо действие.
4. Если на панели настроек установлен флажок “Счётчик”, появится разъём, который позволит заикливать программу на любом её участке на заданное количество завершённых циклов (если подключить разъём к концентратору данных другого блока через шину данных).

## Датчик оборотов

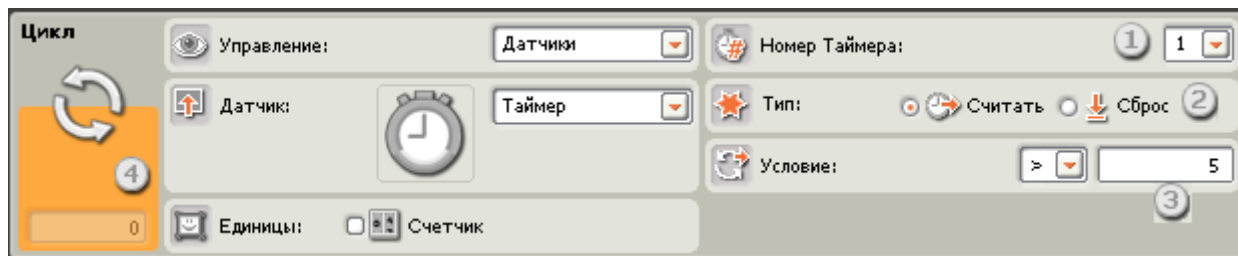


Если выбрать вариант “Считать”, программные блоки внутри цикла будут повторять свои действия до тех пор, пока датчик оборотов не зафиксирует достижение определенного значения; затем выполнение цикла прекратится, а программа продолжит свою работу. Если выбрать вариант “Сброс”, датчик будет сбрасываться на "ноль" после каждого цикла; блок будет продолжать выполнение цикла до тех пор, пока датчик оборотов не зафиксирует достижение порогового значения во время одного из циклов.

1. Выберите порт, к которому подключен мотор.
2. Выберите настройку для датчика оборотов - "Считать" или "Сброс".
3. Кнопками выбора задайте направление вращения: вперед или назад.
4. В открывающемся списке выберите "подсчёт оборотов" или "подсчёт градусов".
5. Введите количество оборотов или градусов, которые вы хотите наработать, прежде, чем прекратиться выполнение цикла (позволив программе продолжить работу).
6. Поле обратной связи будет отображать текущее количество совершённых оборотов или градусов поворота вала мотора.
7. Если на панели настроек установлен флажок “Счётчик”, появится разъём, который позволит заикливать программу на любом её участке на заданное количество завершённых циклов (если подключить разъём к концентратору данных другого блока через шину данных).

## Таймер

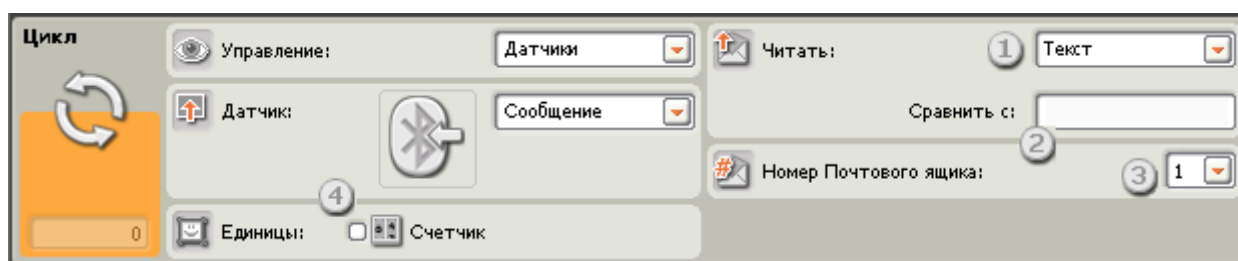




Если выбрать вариант “Считать”, программные блоки внутри цикла будут повторять свои действия до тех пор, пока один из встроенных таймеров NXT не зарегистрирует окончание заданного промежутка времени; затем выполнение цикла прекратится, а выполнение программы будет продолжено. Если выбрать “Сброс”, таймер будет обнуляться после завершения каждого цикла; блок будет продолжать выполнение цикла до тех пор, пока в каком-либо из циклов отсчёт таймера не достигнет своего порогового значения.

1. Выберите таймер NXT, за которым вы хотите наблюдать.
2. Выберите тип работы с таймером - "Считать" или "Сброс".
3. Задайте условие существования цикла: знак и время (в секундах).
4. Если на панели настроек установлен флажок “Счётчик”, появится разъём, который позволит заикливать программу на любом её участке на заданное количество завершённых циклов (если подключить разъём к концентратору данных другого блока через шину данных).

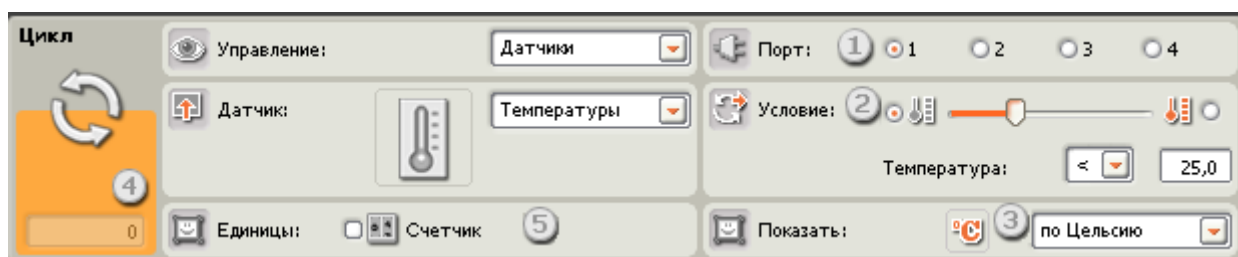
## Приём сообщений



Если выбрать “ Приём сообщений ”, программные блоки внутри цикла будут повторять свои действия до тех пор, пока NXT не получит определенное сообщение через Bluetooth. После этого выполнение цикла прекратится, а выполнение программы продолжится.

1. В открывающемся списке “Читать” выберите тип ожидаемого сообщения (текстовое, числовое или логическое).
2. Для сравнения входящего сообщения с тестовым, наберите контрольный текст или число (в зависимости от того, какой формат был выбран: текстовый или числовой), или кнопками выбора задайте тестовую логическую команду ("истина" или "ложь").
3. Выберите номер почтового ящика для приёма сообщения.
4. Если на панели настроек установлен флажок “Счётчик”, появится разъём, который позволит заикливать программу на любом её участке на заданное количество завершённых циклов (если подключить разъём к концентратору данных другого блока через шину данных).

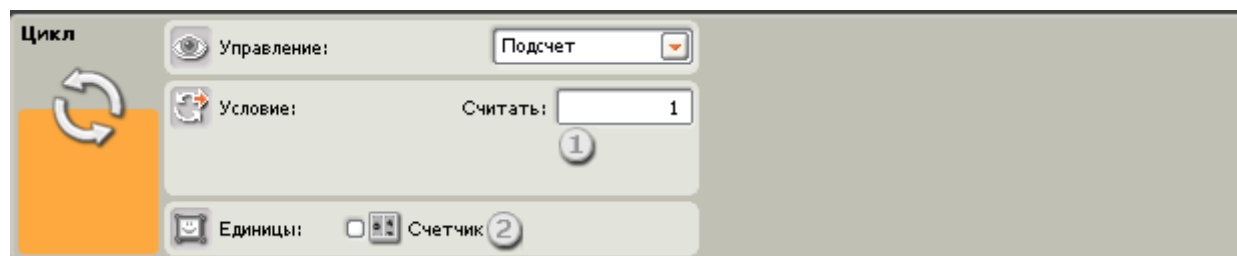
## Датчик температуры



Если выбрать “Датчик температуры”, программные блоки внутри цикла будут повторять свои действия до тех пор, пока датчик температуры не зарегистрирует определенную температуру. После этого выполнение цикла прекратится, а выполнение программы продолжится.

1. Выберите порт, к которому подключен датчик температуры. По умолчанию для датчика температуры будет назначен порт 4.
2. Пороговое значение можно задать движком или ввести нужное значение в поле Температура. При помощи кнопок справа и слева от движка выберите условие остановки цикла (подачи команды "истина") – выше или ниже порогового значения. То же самое можно сделать используя список Температура.
3. Выберите шкалу измерения регистрируемых значений - Цельсия или Фаренгейта.
4. В поле обратной связи будут отображаться текущие показания датчика температуры.
5. Если на панели настроек установлен флажок “Счётчик”, появится разъём, который позволит заикливать программу на любом её участке на заданное количество завершённых циклов (если подключить разъём к концентратору данных другого блока через шину данных).

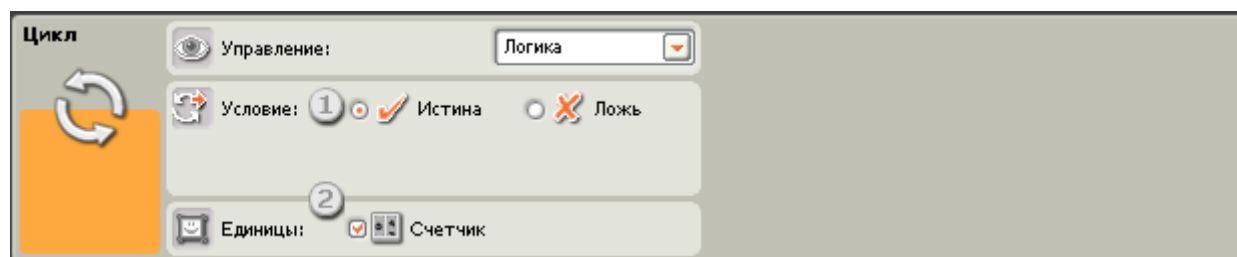
## Подсчёт



Если выбрать “Подсчёт,” программные блоки внутри цикла будут повторять свои действия, пока не будет выполнено заданное количество повторений.

1. Введите в это поле количество повторений, после выполнения которых цикл завершится. Например, если ввести “2”, цикл завершится после того, как программные блоки сработают дважды.
2. Установите флажок “Счётчик”, это даёт возможность соединить правую сторону цикла с левой (где появился разъём "счётчик"). После того как соединение установлено, программа сможет подсчитывать количество завершённых циклов и после выполнения заданного количества циклов остановит их.

## Логика



Если выбрать условие “Логика” и подключить шину входа данных к правой стороне Блока «Цикл», цикл будет повторяться, пока Блок «Цикл» не получит через шину данных логическую команду "истина" или "ложь". После получения такой логической команды, выполнение цикла будет прекращено.

1. Выберите логическую команду, по которой выполнение цикл будет прекращено - "истина" или "ложь".
2. Если на панели настроек установлен флажок “Счётчик”, появится разъём, который позволит заикливать программу на любом её участке на заданное количество завершённых циклов (если подключить разъём к концентратору данных другого блока через шину данных).

### 3. Программный блок остановки

4. Этот блок прекратит выполнение Вашей программы и остановит работу любых моторов, ламп или устройств, генерирующих звук.



### 6. Настройка программного блока остановки



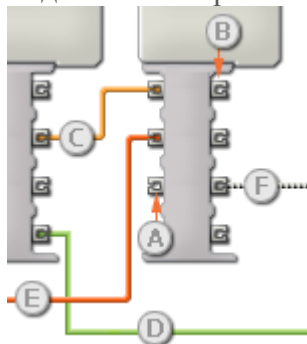
7.  
8. У блока остановки нет настраиваемых параметров.

### 9. Настройка концентратора данных программного блока остановки

10. Вы можете управлять блоком остановки динамически, подключая шины данных (от концентраторов данных других блоков) к концентратору данных блока остановки.
11. Поместив блок в рабочую область, откройте концентратор данных блока, щелкнув по кнопке в нижней левой части блока.



12.  
13. Шины данных, передающие входящие данные на блок, подключены к разъемам, находящимся на левой стороне их концентраторов данных. Шины данных, передающие исходящие данные, подключены к разъемам, находящимся на правой стороне.



14.  
15. [A] Входной разъем  
[B] Выходной разъем  
[C] Числовая шина данных (желтая)  
[D] Логическая шина данных (зеленая)  
[E] Текстовая шина данных (оранжевая)  
[F] Поврежденная шина данных (серая)

### 16. Передача данных от входного разъема на выходной разъем

17. Если для входного разъема есть соответствующий выходной разъем (см. выше A), входящие данные будут проходить от входного разъема на выходной разъем без изменений. В этом случае вы сможете использовать выходной разъем, если входной разъем подключен к шине данных входа; подключение шины данных выхода к такому выходному разъему без подключенной шины данных входа приведет к «неисправности» шины данных выхода (она будет окрашена в серый цвет).

### 18. Шины данных осуществляют передачу специфических типов данных

19. Каждая шина данных осуществляет обмен специфическими типами данных между блоками. Например, если шина данных перетянута с логического разъема на концентратор данным блока, ее можно подключить только к логическому разъему на концентраторе данных другого блока. На

приведенной ниже схеме показано, какой вид данных может принимать или передавать каждый из разъемов.

## 20. Окраски шин данных

21. Шины данных различаются особыми окрасками: шины, передающие числовые данные, имеют желтый цвет; шины, передающие логические данные, имеют зеленый цвет; шины, передающие текстовые данные, имеют оранжевый цвет.

## 22. «Поврежденные» шины данных

23. Если вы пытаетесь подключить шину данных к разъему несоответствующего типа данных, шина будет повреждена (ее цвет изменится на серый). Вы не сможете загрузить вашу программу, если шина данных повреждена.

24. Если вы щелкнете по поврежденной шине, то в небольшом окне контекстной справки в правом нижнем углу рабочей области вы сможете прочесть информацию о причине повреждения.

## 25. Данные должны соответствовать возможному диапазону, в котором работает разъем

26. Если шина данных входа передает значение, не попадающее в допустимый диапазон разъема, к которому она подключена, блок либо проигнорирует такие данные, либо трансформирует данные, попадающие в этот диапазон. Для разъемов, допускающих небольшое количество значений (например, только 0, 1 или 2), разъем проигнорирует входящий сигнал, если передаваемое значение не будет вписываться в диапазон разъема.

27. Для разъемов, допускающих более широкий диапазон значений (например, 0 – 100), разъем усилит любой входящий сигнал, если он не соответствует диапазону. Например, если разъем питания программного блока перемещения получает входящий сигнал со значением 150, блок преобразует входящее значение в 100 (т.е., в числовое значение, лежащее в пределах диапазона разъема мощности).

# Блок «Переключатель»

Этот Блок используют для выбора одной из двух последовательностей команд. Например, если Блок «Переключатель» сконфигурирован под датчик касания, то он должен запустить одну последовательность Блоков в случае, если датчик нажат, и другую последовательность - если датчик не нажат.

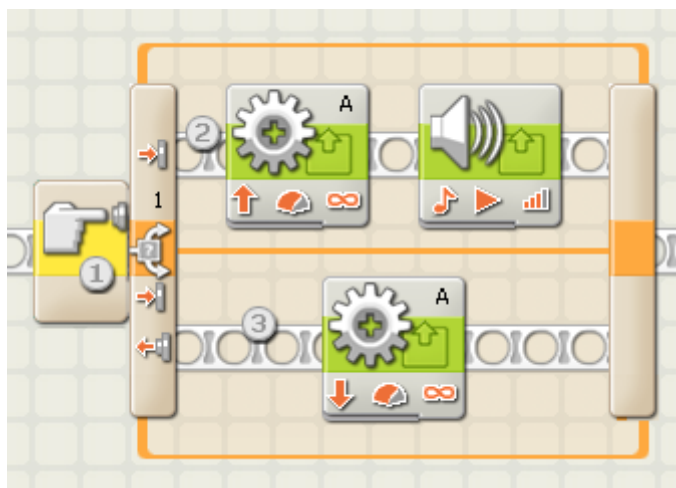
[Отображение параметров настройки Блока](#)

[Добавление Блоков в Блок «Переключатель»](#)

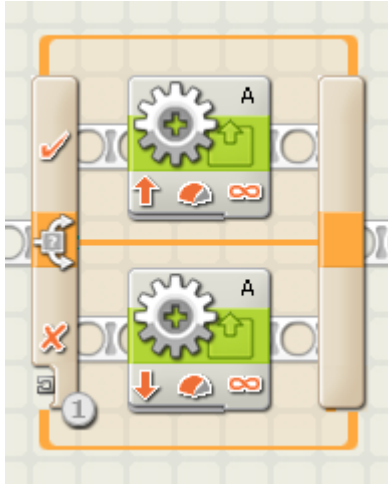
[Перемещение Блока «Переключатель»](#)

[Настройка Блока «Переключатель»](#)

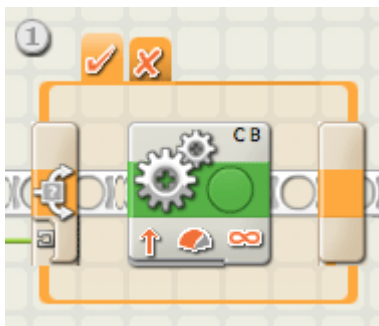
## Отображение параметров настройки Блока :



1. Эта пиктограмма показывает датчик или какое-либо условие, при выполнении которого Блок «Переключатель» должен будет запустить выполнение той или иной последовательности программных Блоков. В этом случае программа будет переключаться в зависимости от текущего состояния датчика касания.
2. Если датчик касания нажат, будут запущены Блоки на верхней цепочке программы.
3. Если датчик касания не нажат, будут запущены Блоки на нижней цепочке программы.



1. Если вы решили управлять Блоком «Переключатель» при помощи условия "Значение", в нижней части левого края Блока появится разъем шины данных; чтобы осуществлять переключение, необходимо присоединить к этому разъему логическую или числовую шину данных, идущую от какого-то другого программного Блока.



1. Если снять флажок “Вид: Без закладок”, то для отображения возможных последовательностей программных Блоков будет предложен интерфейс с набором закладок. Щёлкнув на закладке, можно просмотреть или отредактировать программные Блоки и узнать, при выполнении каких условий эти Блоки были запущены.



Примечание: для подключения внешней шины данных к Блоку, находящемуся внутри Блока «Переключатель», нужно отменить выбор “ Вид: Без закладок”, чтобы показать Блок «Переключатель» с вкладками.

## Добавление программных Блоков в Блок «Переключатель»:

Перетаскивайте Блоки на свободное место внутри рамки, ограничивающей Блок «Переключатель». При этом рамка будет раздвигаться, и Блок можно будет встроить в нужном месте. Если внутри рамки уже есть Блоки, перетаскивайте новые Блоки на нужное место цепочки программы, при этом соседние Блоки раздвинутся.

## Перемещение Блока «Переключатель»:

Блок «Переключатель» можно выделить и перетащить в нужное место, только если щёлкнуть на нём самом.

## Настройка Блока «Переключатель»:

В открывающемся списке секции «Контролировать» можно выбрать одно из двух условий: «Значение» или «Датчик»:

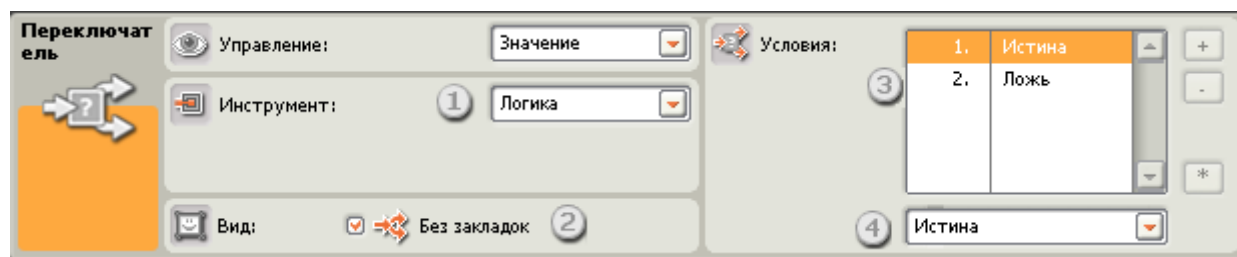
- [Значение](#)
- Датчик (при таком выборе открывается следующий список)
  - [Касания](#)
  - [Освещённости](#)
  - [Звука](#)
  - [Расстояния](#)
  - [Кнопка NXT](#)
  - [Оборотов](#)
  - [Таймер](#)
  - [Приём сообщений](#)
- [Температуры\\*](#)

### Значение

Блок «Переключатель» с управлением по "Значению" может принимать как логические, так и числовые сигналы через шину данных, подключенную к его разъёму, расположенному в левой части Блока. Блок определит тип подведённой шины данных (т.е., логическая она или числовая), и предложит различные варианты переключения, соответствующие каждому случаю.

Если подключена логическая ("истина"/"ложь") шина данных, то при получении Блоком «Переключатель» команды "истина", программа запустит выполнение верхней цепочки программы. При получении команды "ложь", будет запущена нижняя цепочка.

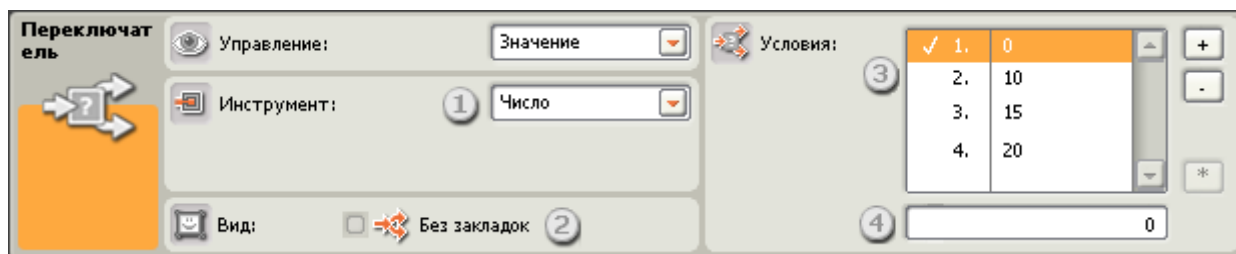
При подключении числовой или текстовой шины данных, можно задать точное значение входного сигнала, по которому будет запущена та или иная цепочка (программы). Более того, если отключить отображение “Вид: Без закладок”, можно будет запустить более двух цепочек программы.



1. В этом поле будет показан тип входного сигнала, определённый Блоком.
2. Если снять флажок в окошке “Вид: Без закладок”, то для отображения возможных последовательностей Блоков будет предложен интерфейс с набором закладок. Щёлкнув на закладке, можно просмотреть или отредактировать Блоки и узнать, при выполнении каких условий были запущены эти Блоки.

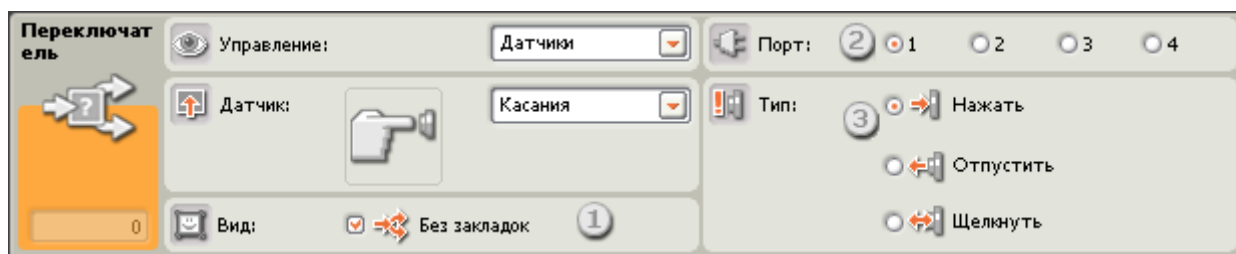
- Каждый ряд в таблице представляет собой вариант переключения. Если включено отображение “Вид: Без закладок”, самый верхний вариант, номер 1 - это условие, при котором будет запущена верхняя цепочка (программы). По второму будет запущена нижняя цепочка.

Если числовая или текстовая шина данных подключена к Блоку «Переключатель», а отображение “Вид: Без закладок” отменено (при этом включается диалоговый интерфейс с отображением закладок), можно добавить дополнительные строки таблицы, что позволит управлять большим количеством цепочек программы, как показано на рисунке внизу.



- Это поле выполняет две функции. Если параметр Инструмент имеет значение "Логика", то в этом поле можно назначить условие ("истина") включения одной из двух цепочек. Если это значение "Число" или "Текст", в это поле для каждой цепочки вводят число или текст, которые будут её запускать.
- При помощи этих кнопок можно добавлять или удалять варианты переключения. Они активны только если снят флажок “Вид: Без закладок”, и к входному разъёму Блока «Переключатель» протянута числовая или текстовая шина данных.
- Кнопка “\*” присваивает выделенной цепочке статус "запуск по умолчанию". Когда Блок «Переключатель» настроен на числовой входной сигнал, и поступающий сигнал не совпадает с заданными числовыми значениями, Блок начнёт работать независимо от того, какой вариант установлен по умолчанию.

## Датчик касания

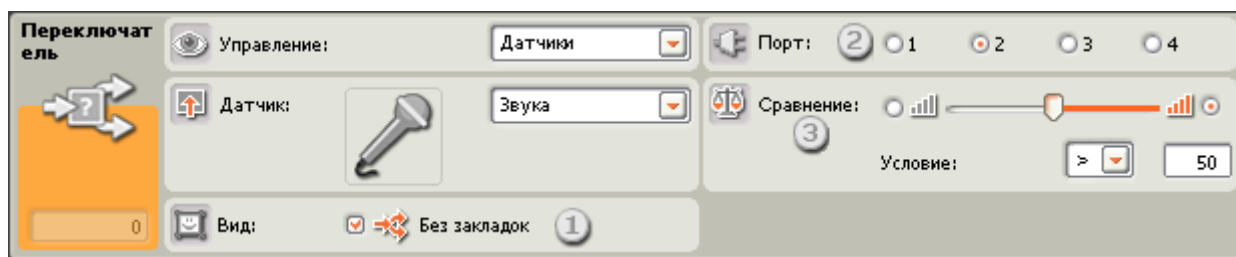


Выберите этот пункт, чтобы программа переключалась в зависимости от состояния датчика касания.

- Если снять флажок “Вид: Без закладок”, то для отображения возможных вариантов программы будет предложен интерфейс с набором закладок. Щёлкнув на закладке, можно просмотреть и отредактировать Блоки каждого варианта и узнать, при выполнении каких условий эти варианты будут выбраны.
- Выберите порт, к которому подключен датчик касания. По умолчанию для датчика касания будет назначен порт 1. При необходимости эту настройку можно изменить.
- Кнопками выбора можно задать при каком состоянии датчика касания Блок «Переключатель» запустит Блоки в верхней цепочке; нижняя цепочка включится в том случае, если не было произведено никаких действий. При выборе "Нажать", Блок переключается при нажатии кнопки датчика касания. При выборе "Отпустить", Блок переключается после освобождения кнопки датчика. Если выбран вариант "Щелкнуть", Блок переключается при щелчке кнопкой датчика.

## Датчик звука

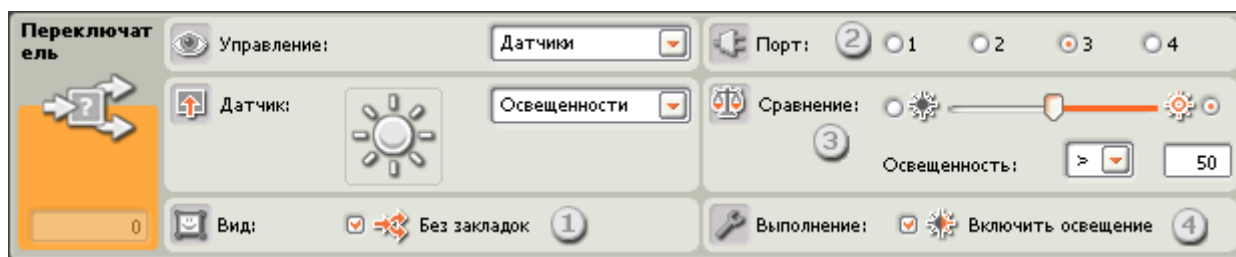




Выберите этот пункт, чтобы программа переключалась в зависимости от показаний датчика звука.

1. Если снять флажок “Вид: Без закладок”, то для отображения возможных последовательностей программных Блоков будет предложен интерфейс с набором закладок. Щёлкнув на закладке, можно просмотреть или отредактировать Блоки и узнать, при выполнении каких условий были запущены эти Блоки.
2. Выберите порт, к которому подключен датчик звука. По умолчанию для датчика звука Блоку будет назначен порт 2. При необходимости эту настройку можно изменить.
3. Задайте пороговое значение движком или введите значение непосредственно в поле. При установке по умолчанию (50%), будет запущена верхняя цепочка (программы), если датчик зарегистрирует звук, уровень которого превышает 50%; если уровень громкости будет меньше 50%, включится нижняя цепочка.

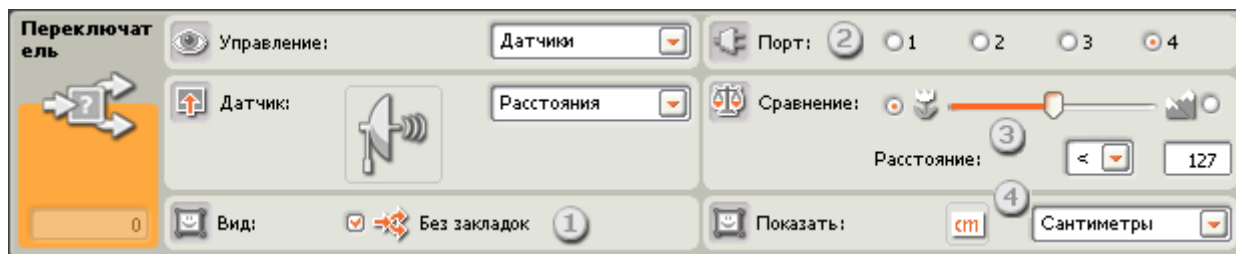
### Датчик освещённости



Выберите этот пункт, чтобы программа переключалась в зависимости от показаний датчика освещённости.

1. Если снять флажок “Вид: Без закладок”, то для отображения возможных последовательностей программных Блоков будет предложен интерфейс с набором закладок. Щёлкнув на закладке, можно просмотреть или отредактировать программные Блоки и узнать, при выполнении каких условий были запущены эти Блоки.
2. Выберите порт, к которому подключен датчик освещённости. По умолчанию для датчика освещённости Блоку будет назначен порт 3. При необходимости эту настройку можно изменить.
3. Задайте пороговое значение движком или введите значение непосредственно в поле. При установке по умолчанию (50%), будет запущена верхняя цепочка (программы), если датчик зарегистрирует уровень освещённости выше 50%; если уровень освещённости будет меньше 50%, включится нижняя цепочка.
4. Если установлен флажок “Включить освещение”, датчик включит свою лампу подсветки и зарегистрирует отражённый свет.
5. В поле обратной связи отображаются текущие показания датчика освещённости (0-100%).

### Датчик расстояния

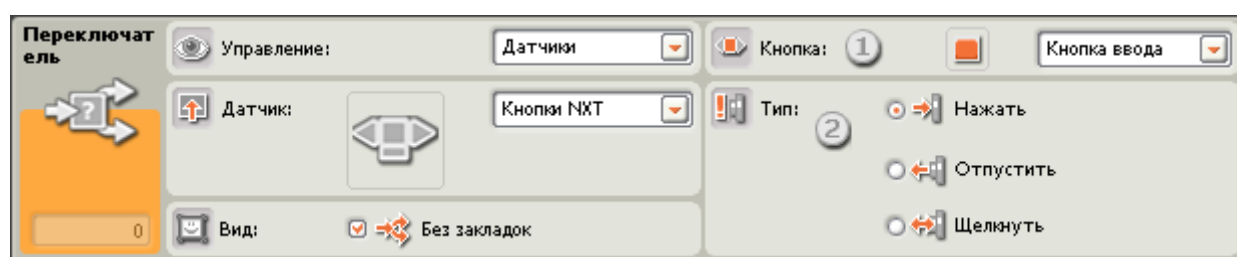




Выберите этот пункт, чтобы программа переключалась в зависимости от показаний датчика расстояния.

1. Если снять флажок “Вид: Без закладок”, то для отображения возможных последовательностей программных Блоков будет предложен интерфейс с набором закладок. Щёлкнув на закладке, можно просмотреть или отредактировать программные Блоки и узнать, при выполнении каких условий были запущены эти Блоки.
2. Выберите порт, к которому подключен датчик расстояния. По умолчанию для датчика расстояния Блоку будет назначен порт 4. При необходимости эту настройку можно изменить.
3. Задайте пороговое значение движком или введите его непосредственно в поле. При установке по умолчанию 50 (127), будет запущена верхняя цепочка (программы), если датчик обнаружит объект на расстоянии превышающем 50 дюймов (127 см), если объект будет зарегистрирован на расстоянии меньшем, чем 50 дюймов (127 см), включится нижняя цепочка.
4. Выберите единицы измерения расстояния - сантиметры или дюймы.
5. В поле обратной связи отображаются текущие показания датчика расстояния.

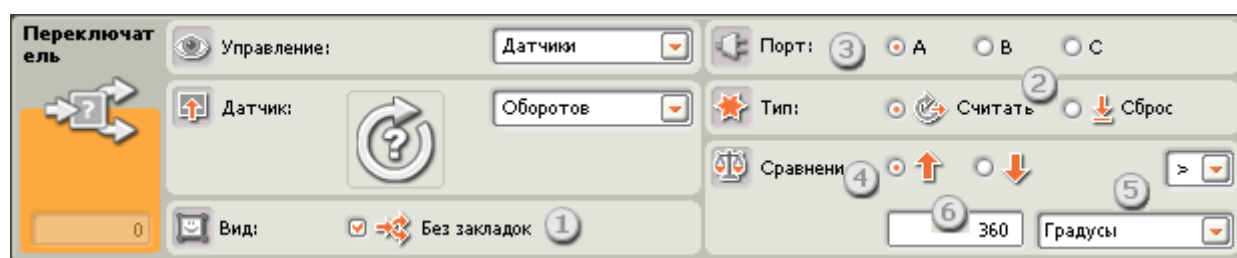
## Кнопки NXT



При выборе датчика “Кнопки NXT”, будет запущена верхняя цепочка (программы), если будет зарегистрировано какое-либо воздействие на выбранную кнопку NXT. Если никаких действий не будет зарегистрировано, включится нижняя цепочка (программы).

1. Выберите кнопку NXT, при нажатии которой будет отправлена команда "истина" (и будет запущена верхняя цепочка программы).
2. Выберите "Щелкнуть", чтобы переключение происходило после щелчка кнопкой NXT. Выберите "Нажать", чтобы переключение происходило при нажатии на неё. Выберите "Отпустить", чтобы переключение происходило при освобождении кнопки.

## Датчик оборотов

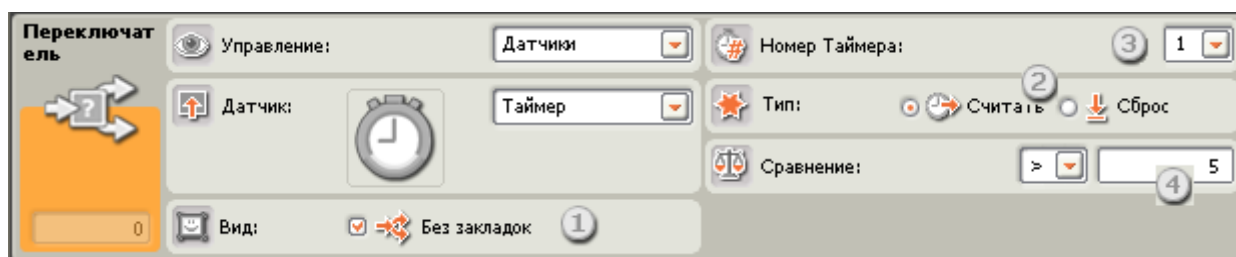


Выберите этот пункт, чтобы программа переключалась в зависимости от количества оборотов или градусов поворота вала мотора.

1. Если снять флажок “Вид: Без закладок”, то для отображения возможных последовательностей программных Блоков будет предложен интерфейс с набором закладок. Щёлкнув на закладке, можно просмотреть или отредактировать программные Блоки и узнать, при выполнении каких условий были запущены эти Блоки.
2. Если параметр Тип имеет значение "Сброс", Блок «Переключатель» будет считывать показания датчика оборотов и обнулять счётчик. Если выбрать "Считать", показания датчика обнуляться не будут.
3. Укажите порт, к которому подключен мотор, используемый Блоком «Переключатель».

- Кнопками выбора укажите направление вращения мотора (вперед или назад), которое должно контролироваться Блоком.
- Выберите из списка единицу измерения "Обороты" или "Градусы".
- Введите пороговое значение в выбранных единицах измерения. По умолчанию пороговое значение равно 360 градусов. При такой настройке запустится верхняя цепочка (программы), если будет зарегистрирован поворот вала мотора больше, чем 360 градусов. При меньших значениях программа пойдет по нижней цепочке.
- В поле обратной связи отображается текущее количество оборотов или градусов.

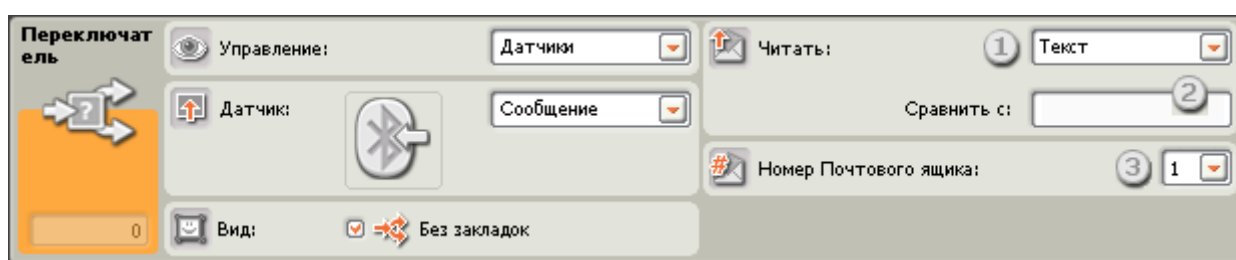
## Таймер



Выберите этот пункт, чтобы программа переключалась в зависимости от текущих показаний таймера.

- Если снять флажок “Вид: Без закладок”, то для отображения возможных последовательностей программных Блоков будет предложен интерфейс с набором закладок. Щёлкнув на закладке, можно просмотреть или отредактировать программные Блоки и узнать, при выполнении каких условий были запущены эти Блоки.
- Если выбрать "Сброс", Блок «Переключатель» будет считывать показания таймера и обнулять его. Если выбрать "Считать", показания таймера обнуляться не будут.
- Выберите один из трех таймеров NXT, который будет использован.
- Введите пороговое значение. По умолчанию задано 5 секунд. При такой настройке запустится верхняя цепочка (программы), если текущее показание превысит 5 секунд, а если будет зарегистрировано меньше 5 секунд, включится нижняя цепочка.

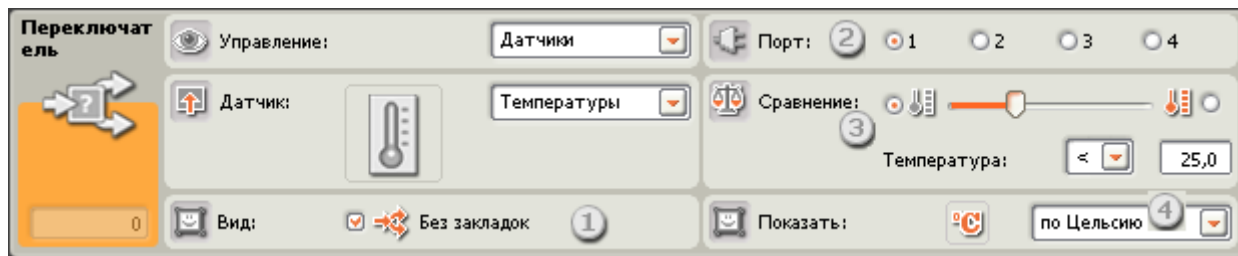
## Сообщение



Если выбрать “Сообщение”, то при поступлении определенного сообщения по Bluetooth будет запущена верхняя цепочка (программы). Если такое сообщение не поступит, включится нижняя цепочка.

- В открывающемся списке можно выбрать тип ожидаемого сообщения (текстовое, числовое или логическое).
- Для сравнения входящего сообщения с контрольным, наберите тестовый текст или число (в зависимости от того, какой формат был выбран: текстовый или числовой), или кнопками выбора задайте контрольную логическую команду ("истина" или "ложь"). Если входящее сообщение совпадает с тестовым текстом или значением, будет запущена верхняя цепочка (программы). В противном случае включится нижняя цепочка.
- Выберите номер почтового ящика, в который будет поступать сообщение.

## Датчик температуры\*

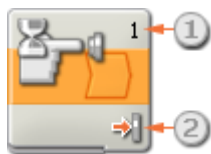


Выберите этот пункт, чтобы программа переключалась в зависимости от показаний Датчика температуры.

1. Если снять флажок “Вид: Без закладок”, то для отображения возможных последовательностей программных Блоков будет предложен интерфейс с набором закладок. Щёлкнув на закладке, можно просмотреть или отредактировать программные Блоки и узнать, при выполнении каких условий были запущены эти Блоки.
2. Выберите порт, к которому подключен датчик температуры. По умолчанию для датчика температуры Блоку будет назначен порт 4. При необходимости эту настройку можно изменить.
3. Пороговое значение можно задать движком или ввести нужное значение в поле. Значение по умолчанию 25° C или 77° F. В открывающемся списке выберите “<”, тогда, если температура окажется ниже, чем 25° C/77° F, будет запущена верхняя цепочка (программы), если температура превысит 25° C/77° F, включится нижняя цепочка. При выборе “>” в открывающемся списке, эти операции будут выполняться наоборот, как и при активации противоположной кнопки выбора.
4. Выберите шкалу измерения температуры - градусы Цельсия или Фаренгейта.
5. В поле обратной связи отображаются текущие значения температуры.

## Блок «Ожидание»

Этот Блок придаёт роботу способность следить за окружающей его обстановкой, ожидая наступления определенных условий, чтобы продолжить свои действия. Воспользуйтесь движком, или введите число, чтобы задать пороговое значение времени или показания датчика.



1. Цифра или буква в верхнем левом углу Блока «Ожидание» указывает порт, отслеживаемый блоком. Порты при необходимости можно сменить в панели настройки.
2. При выборе ожидания сигнала датчика освещённости, звука или расстояния, эта пиктограмма показывает установленный уровень порогового значения. Чем больше окрашенных полосок, тем выше пороговое значение. Если выбрано ожидание сигнала датчика касания, пиктограмма отображает, какое воздействие на этот датчик продолжит выполнение программы (Щелчок, Нажатие или Освобождение кнопки датчика).

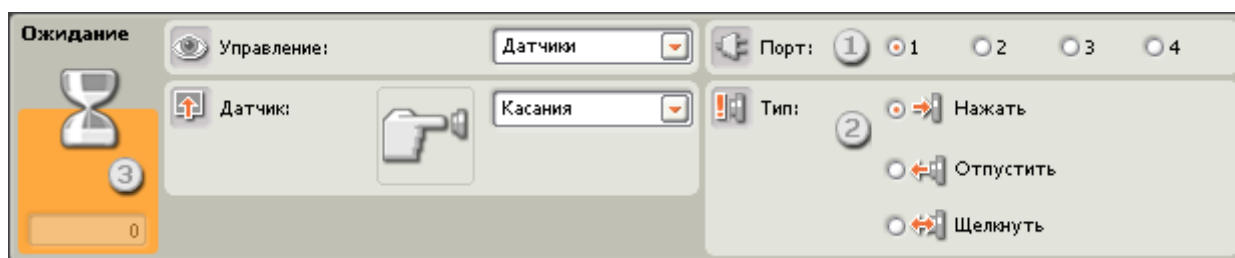
## Настройка Блока «Ожидание»

В списке "Управление", выбирается условие, управляющее Блоком: время, или показание датчика:

- Датчик (открывается второй список)
  - [Касания](#)
  - [Звука](#)
  - [Освещённости](#)
  - [Расстояния](#)

- [Кнопки NXT](#)
- [Оборотов](#)
- [Таймер](#)
- [Сообщение](#)
- [Температуры\\*](#)
- [Время](#)

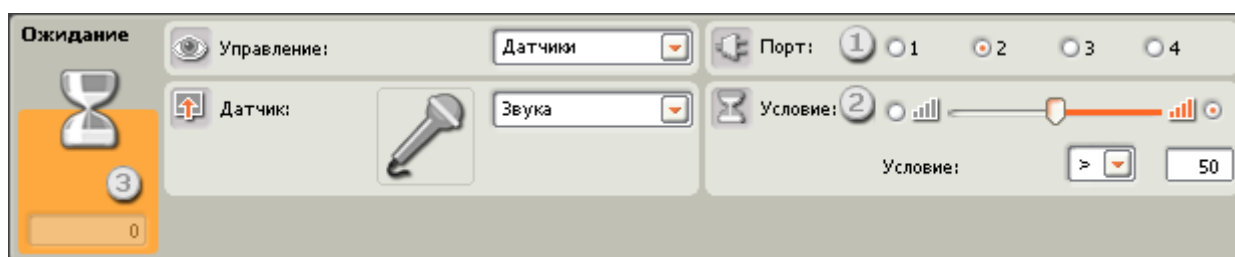
## Датчик Касания



При выборе "Датчик касания" программа будет ожидать заданного состояния датчика касания (Щелчок, Нажатие или Освобождение), прежде чем перейти к следующему блоку.

1. Выберите порт, к которому подключен датчик касания. По умолчанию назначается порт 1.
2. Укажите действие (Щелкнуть, Нажать или Отпустить), которое должна ожидать программа для перехода к следующему Блоку. Если выбрано "Щелкнуть", Блок будет срабатывать при резком надавливании и ослаблении давления на кнопку датчика касания. "Нажать" - Блок будет срабатывать при нажатии на кнопку датчика касания. "Отпустить" - Блок сработает при ослаблении давления на кнопку датчика касания.
3. Поле обратной связи позволяет контролировать состояние датчика касания. Если датчик работа активирован, в поле появится "1".

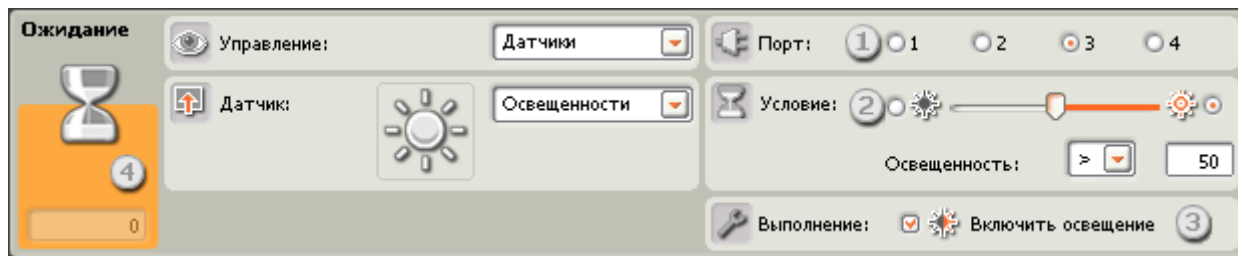
## Датчик Звука



Выберите "Датчик звука", и программа будет ожидать, когда датчик звука зарегистрирует звук заданной громкости.

1. Выберите порт, к которому подключен датчик звука. По умолчанию для датчика звука назначен порт 2.
2. Задайте пороговое значение движком, или напечатайте его в поле ввода. Активируйте правую (относительно движка) кнопку выбора, чтобы Блок срабатывал при уровнях громкости, превышающих пороговое значение. Активируйте левую кнопку выбора, чтобы Блок срабатывал при уровнях громкости меньших, чем пороговое значение. Установить движок в диапазоне подачи команды "истина" можно и при помощи открывающегося списка.
3. В поле обратной связи отображаются текущие показания датчика звука (0-100%). Его можно использовать для опробования различных пороговых значений.

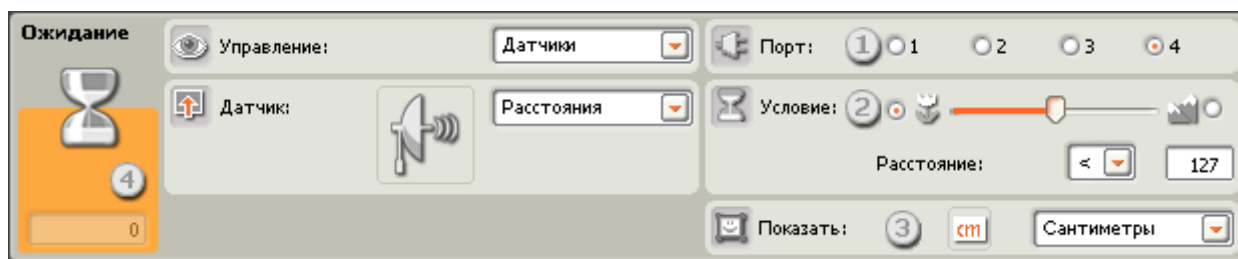
## Датчик освещённости



Выберите "Датчик освещённости", и программа будет ожидать, когда датчик освещённости зарегистрирует заданный уровень освещённости.

1. Выберите порт, к которому подключен датчик освещённости. По умолчанию для него блоку будет назначен порт 3.
2. Задайте пороговое значение движком, или введите его в поле. Активируйте правую (относительно движка) кнопку выбора, чтобы Блок срабатывал при уровнях освещённости, превышающих пороговое значение. Активируйте левую кнопку выбора, чтобы Блок срабатывал при уровнях освещённости меньших, чем пороговое значение. Установить движок в диапазоне подачи команды "истина" можно и при помощи открывающегося списка.
3. Если установлен флажок "Включить освещение", датчик включит собственную лампу подсветки и зарегистрирует отражённый свет.
4. Поле обратной связи отображает текущие показания датчика освещённости (0-100%). Его можно использовать для опробования различных пороговых значений.

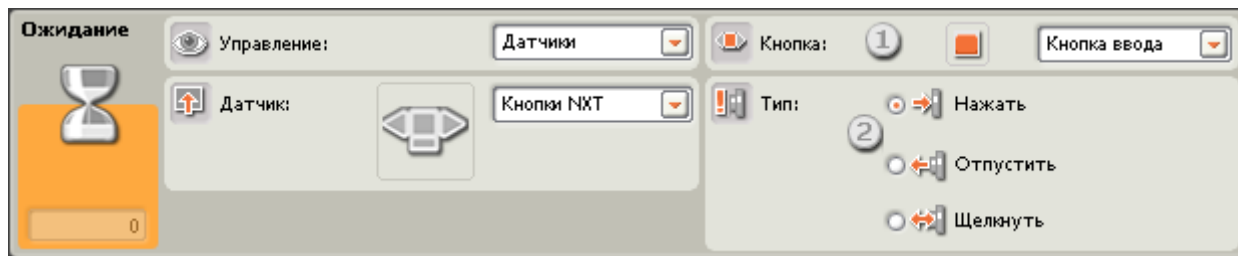
### Датчик Расстояния



Выберите "Датчик расстояния", и программа будет ожидать, когда датчик расстояния обнаружит объекта на определенном удалении.

1. Выберите порт, к которому подключен датчик расстояния. По умолчанию для него блоку будет назначен порт 4.
2. При активации левой кнопки выбора, Блок запускается при регистрации объекта, находящегося на расстоянии меньшем, чем пороговое значение; если активирована правая кнопка выбора, Блок включится при обнаружении объекта на расстоянии большем, чем пороговое значение. Задайте движком пороговое значение, или введите его непосредственно в поле (0-250, если расстояние измеряется в сантиметрах, или 0-100 - для дюймов). Помните о том, что объекты с хорошо отражающей поверхностью можно обнаружить на большем расстоянии, чем объекты со слабо отражающей поверхностью.
3. Выберите единицу измерения расстояния - сантиметры или дюймы.
4. В поле обратной связи отображаются текущие показания датчика расстояния (0-250 см или 0-100 дюймов). Показание "0" соответствует минимальному расстоянию, на котором датчик может обнаруживать объекты. Максимальное расстояние обнаружения составляет приблизительно 250 см (показание 250) или 100 дюймов (показание 100).

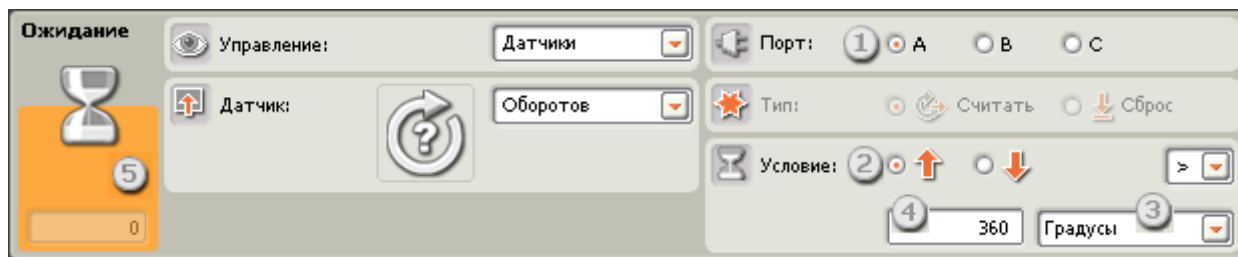
### Кнопки NXT



Если выбрать "Кнопки NXT", программа будет ожидать совершения определённого действия (Щелчок, Нажатие или Освобождение) выбранной кнопкой NXT, прежде чем продолжить выполнение.

1. Выберите, при активации какой кнопки NXT будет выдан сигнал "истина" для продолжения выполнения программы.
2. Щелкнуть - сигнал возникает после быстрого нажатия и освобождения выбранной кнопки. "Нажать" - сигнал возникает после при нажатии выбранной кнопки. "Отпустить" - сигнал возникает после при освобождении выбранной кнопки.

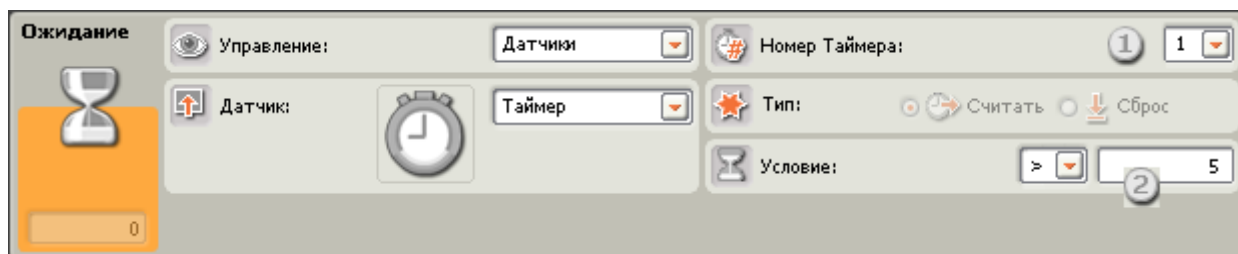
## Датчик оборотов



При выборе "Датчика оборотов" выполнение программы продолжится после того, как вал мотора повернется на заданное количество оборотов или градусов.

1. Выберите порт, к которому подключен мотор.
2. Кнопками выбора задайте направление вращения мотора: вперед или назад.
3. В открывающемся списке выберите "Обороты" или "Градусы".
4. Задайте, на какое количество оборотов или градусов должен повернуться вал мотора, после чего выполнение программы сможет продолжиться.
5. В поле обратной связи будет отображаться текущее количество оборотов или градусов. Для очистки поля обратной связи нажмите кнопку Сброс.

## Таймер

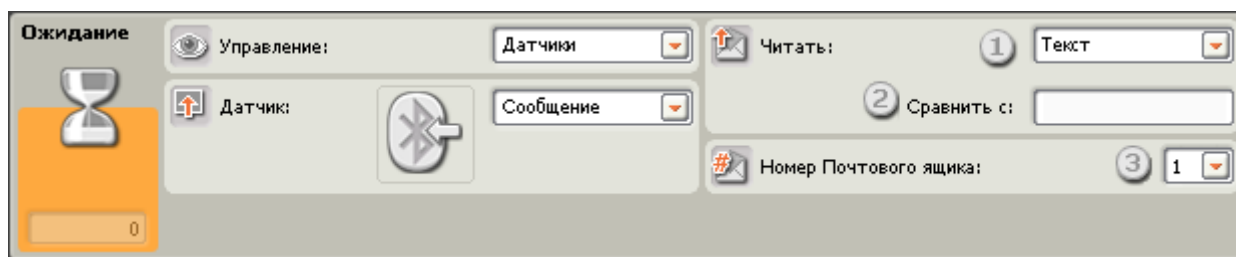


Если выбрать "Таймер", выполнение программы продолжится после того, как один из внутренних таймеров NXT отсчитает заданный промежуток времени.

1. Выберите таймер NXT, который будет отображаться и контролироваться.
2. Введите значение времени (в секундах) в поле. Это значение можно увеличивать или уменьшать при помощи стрелок вверх-вниз.

## Приём сообщений

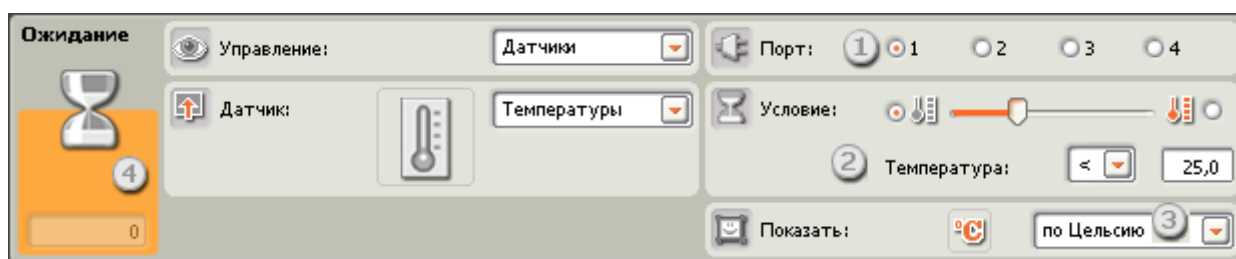




Если выбрать "Сообщение", выполнение программы продолжится после того, как NXT получит определенное сообщение по Bluetooth.

1. В открывающемся списке можно выбрать тип ожидаемого сообщения (текстовое, числовое или логическое).
2. Для сравнения входящего сообщения с контрольным, наберите тестовый текст или число (если для сообщения выбран соответственно текстовый или числовой формат), или кнопками выбора задайте контрольное логическое значение ("истина" или "ложь").
3. Выберите номер почтового ящика, в который будет попадать сообщение.

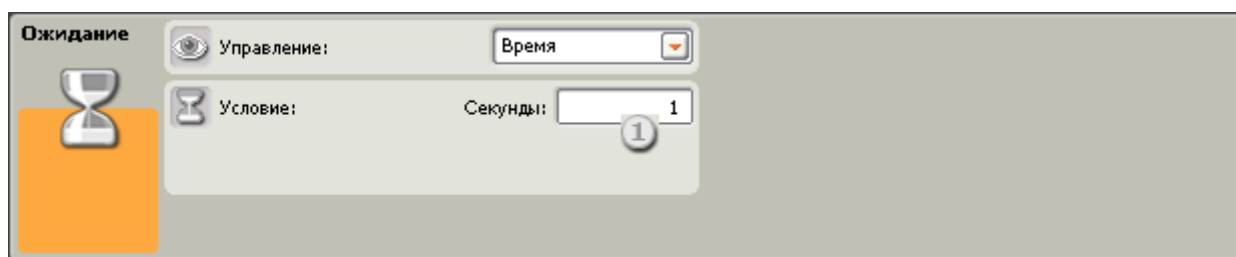
### Датчик температуры\*



Если выбрать "Датчик температуры"\*, выполнение программы продолжится после того, как датчик температуры зарегистрирует её заданное значение.

1. Выберите порт, к которому подключен датчик температуры. По умолчанию для него блоку будет назначен порт 4.
2. Задайте пороговое значение движком, или введите его в поле. Активируйте правую (относительно движка) кнопку выбора, чтобы Блок срабатывал при температуре, превышающей пороговое значение. Активируйте левую кнопку выбора, чтобы Блок срабатывал при температурах меньших, чем пороговое значение. Установить движок в диапазоне подачи команды "истина" можно и при помощи открывающегося списка.
3. Выберите шкалу измерения температуры - градусы Цельсия или Фаренгейта.
4. Поле обратной связи будет отображать текущие значения температуры.

### Время



Выберите "Время", чтобы выполнение программы приостанавливалось на заданный промежуток времени, прежде чем перейти к следующему блоку.

1. Выберите количество секунд, которое нужно ждать, прежде чем продолжить выполнение программы. Время можно задать с точностью до десятых долей секунды (например: 12,3 или 12.3 (в английском языке)).

- [Блок «Сравнение»](#)
- [Блок «Логика»](#)
- [Блок «Математика»](#)
- [Блок «Случайное число»](#)
- [Блок «Интервал»](#)
- [Блок «Переменная»](#)
- [Блок «Константа»](#)

## Программный блок сравнения

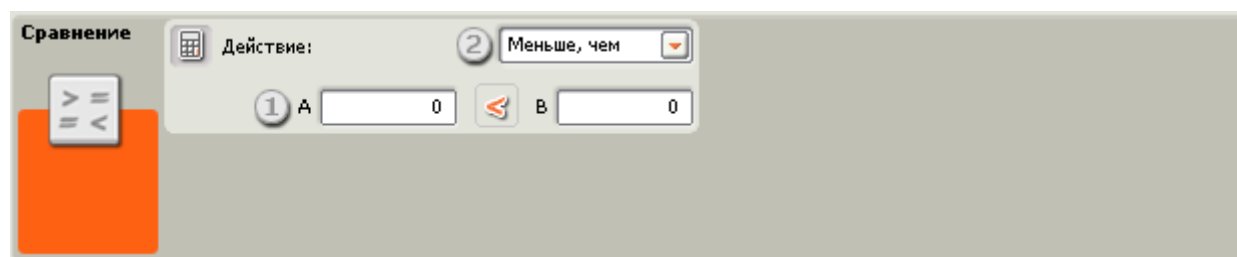
Этот блок может определить, является ли число больше ( $>$ ) или меньше ( $<$ ), чем другое число, или равно ( $=$ ) другому числу. Числовые значения могут быть введены путем набора или динамически, при помощи шин данных.



1. Программный блок сравнения, перемещенный в рабочую область, получит открытый концентратор данных с двумя портами ввода (слева) и тремя портами вывода (справа). Оба порта ввода должны быть подключены к другим программным блокам посредством шин данных (кроме случая, когда один из портов ввода получает сигнал с постоянным значением, которое Вы вводите).

Выходной сигнал из блока сравнения будет подаваться с самого нижнего выходного разъема; подключите этот разъем при помощи шины данных к концентратору данных другого блока. Два выходных разъема напротив входных разъемов позволяют вам передавать значения входа на другие блоки, если это потребуется. (Чтобы получить дополнительную информацию, см. ниже раздел "Концентратор данных".)

## Настройка программного блока сравнения



1. Числовые значения могут быть введены путем набора или динамически, при помощи шин данных. При подключении шин данных поля ввода будут деактивированы (подсвечены серым).
2. Ниспадающее меню позволит Вам выбрать из трех типов сравнения, которые Вы можете осуществить:
  - Больше, чем ( $>$ )
  - Меньше, чем ( $<$ )
  - Равно ( $=$ )

**Больше, чем ( $>$ )**



В данном случае, если первое вводное числовое значение больше или равно второму, программный блок сравнения подаст сигнал "истина". В противном случае он сообщит о том, что это значение - "ложь".

### Меньше, чем (<)

В данном случае, если первое вводное числовое значение меньше или равно второму, программный блок сравнения подаст сигнал "истина". В противном случае он сообщит о том, что это значение - "ложь".

### Равно (=)

В данном случае, если первое вводное числовое значение равно второму, программный блок сравнения подаст сигнал "истина". В противном случае он сообщит о том, что это значение - "ложь".

Если вводное A = вводимому B, выходное значение = "истина"

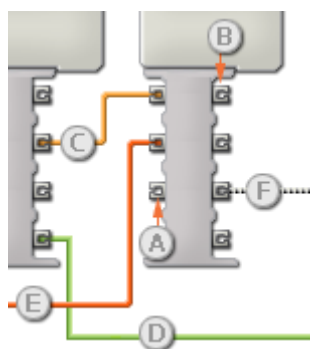
## Настройка концентратора данных программного блока сравнения

Вы можете управлять блоком сравнения динамически, подключая шины данных (от концентраторов данных других блоков) к концентратору данных блока сравнения.

Поместив блок в рабочую область, откройте концентратор данных блока, щелкнув по кнопке в нижней левой части блока.



Шины данных, передающие входящие данные на блок, подключены к разъемам, находящимся на левой стороне их концентраторов данных. Шины данных, передающие исходящие данные, подключены к разъемам, находящимся на правой стороне.



- [A] Входной разъем
- [B] Выходной разъем
- [C] Числовая шина данных (желтая)
- [D] Логическая шина данных (зеленая)
- [E] Текстовая шина данных (оранжевая)
- [F] Поврежденная шина данных (серая)

### Передача данных от входного разъема на выходной разъем

Если для входного разъема есть соответствующий выходной разъем (см. выше А), входящие данные будут проходить от входного разъема на выходной разъем без изменений. В этом случае вы сможете использовать выходной разъем, если входной разъем подключен к шине данных входа; подключение шины данных выхода к такому выходному разъему без подключенной шины данных входа приведет к «неисправности» шины данных выхода (она будет окрашена в серый цвет).

## **Шины данных осуществляют передачу специфических типов данных**

Каждая шина данных осуществляет обмен специфическими типами данных между блоками. Например, если шина данных перетянута с логического разъема на концентратор данных блока, ее можно подключить только к логическому разъему на концентраторе данных другого блока. На приведенной ниже схеме показано, какой вид данных может принимать или передавать каждый из разъемов.

## **Окраски шин данных**

Шины данных различаются особыми окрасками: шины, передающие числовые данные, имеют желтый цвет; шины, передающие логические данные, имеют зеленый цвет; шины, передающие текстовые данные, имеют оранжевый цвет.

## **«Поврежденные» шины данных**

Если вы пытаетесь подключить шину данных к разъему несоответствующего типа данных, шина будет повреждена (ее цвет изменится на серый). Вы не сможете загрузить вашу программу, если шина данных повреждена.

Если вы щелкнете по поврежденной шине, то в небольшом окне контекстной справки в правом нижнем углу рабочей области вы сможете прочесть информацию о причине повреждения.

## **Данные должны соответствовать возможному диапазону, в котором работает разъем**

Если шина данных входа передает значение, не попадающее в допустимый диапазон разъема, к которому она подключена, блок либо проигнорирует такие данные, либо трансформирует данные, попадающие в этот диапазон. Для разъемов, допускающих небольшое количество значений (например, только 0, 1 или 2), разъем проигнорирует входящий сигнал, если передаваемое значение не будет вписываться в диапазон разъема.

Для разъемов, допускающих более широкий диапазон значений (например, 0 – 100), разъем усилит любой входящий сигнал, если он не соответствует диапазону. Например, если разъем питания программного блока перемещения получает входящий сигнал со значением 150, блок преобразует входящее значение в 100 (т.е., в числовое значение, лежащее в пределах диапазона разъема мощности).

# **Программный блок логики**

Этот блок выполняет логическую операцию с входящими сигналами и посылает ответные сигналы истина/ложь через шину данных. Входящие сигналы (которые также должны иметь значения "истина" или "ложь") могут быть назначены при помощи селективных кнопок или поданы через шины данных.

Блок логики принимает только два возможных значения данных - "истина" и "ложь" - для входа и для выхода. Часто эти значения записываются в виде чисел "1" и "0", где

любое истинное утверждение записывается как "1", а каждое ложное утверждение записывается как "0".

Такая система записи очень удобна, если ее применять в компьютерах, потому что в памяти хранятся быстро записываемые двоичные значения, у которых есть только два варианта - 1 и 0. Компьютеры справляются с большими числами и сложными вычислениями, комбинируя их при помощи 1 и 0.



Примечание: вы можете подавать на вход этого блока значения 1 и 0, однако их передачу должны осуществлять логические шины данных.

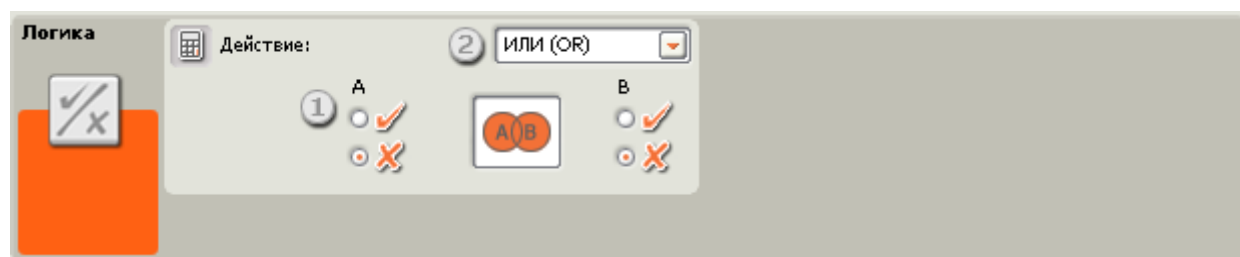
Четыре логических операции, которые может выполнять этот блок, позволяют вам произвести ряд сравнений. Эти операции (четыре логических оператора: And - И, Or - ИЛИ, Xor - Исключающее ИЛИ и Not - НЕ) подробно описаны ниже.



1. Блок логики, помещенный в рабочую область, откроется с двумя портами ввода и тремя портами вывода. Порты ввода должны быть подключены к другим программным блокам посредством шин данных (кроме случая, когда один из них получает сигнал с постоянным значением, которое вы устанавливаете при помощи селективных кнопок).

Выходной сигнал логической операции будет подаваться с самого нижнего выходного разъема; подключите этот разъем при помощи шины данных к концентратору данных другого блока. Два выходных разъема напротив входных разъемов позволяют вам передавать значения входа на другие блоки, если это потребуется. (Чтобы получить дополнительную информацию, см. ниже раздел "Концентратор данных".)

## Настройка программного блока логики



1. Воспользуйтесь селективными кнопками, чтобы выбрать входы A и B, разрешите шинам входных данных передавать входящие данные динамически.
2. В ниспадающем меню вам будет предложено выбрать одну из четырех операций, которые вы можете выполнить на входах:
  - Операция "And" ("И")
  - Операция "Or" ("ИЛИ")
  - Операция "Xor" ("Исключающее ИЛИ")
  - Операция "Not" ("НЕ")



## Логическая операция “And” (“И”)

При операции логического умножения “And” (“И”), если два ваших входных значения являются “истинной”, выходные значения также будут иметь значение “истина”; во всех других случаях выходному значению будет присвоено “ложь”.

Это становится понятным, если вы рассмотрите “логическую таблицу”:

Вход А	Вход В	Выход
Ложь (0)	Ложь (0)	Ложь (0)
Ложь (0)	Истина (1)	Ложь (0)
Истина (1)	Ложь (0)	Ложь (0)
Истина (1)	Истина (1)	Истина (1)

Вы видите, что только тогда, когда оба входных значения имеют значение “истина”, на выходе тоже “истина”; во всех остальных случаях на выходе значение “ложь”.



## Логическая операция “Or” (“ИЛИ”)

При операции логического сложения “Or” (“ИЛИ”), если одно или два входных значения являются “истинной”, выходные значения также будут иметь значение “истина”.

Вход А	Вход В	Выход
Ложь (0)	Ложь (0)	Ложь (0)
Ложь (0)	Истина (1)	Истина (1)
Истина (1)	Ложь (0)	Истина (1)
Истина (1)	Истина (1)	Истина (1)



## Операция “Xor” (исключающее “ИЛИ”)

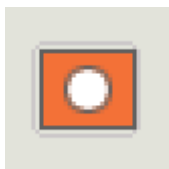
При операции “Xor” (исключающее “ИЛИ”), если какое-нибудь одно из двух входных значений является “истинной”, выходное значение также будет иметь значение “истина”.

Вход А	Вход В	Выход
Ложь (0)	Ложь (0)	Ложь (0)
Ложь (0)	Истина (1)	Истина (1)
Истина (1)	Ложь (0)	Истина (1)

Истина (1)

Истина (1)

Ложь (0)



## Операция отрицания “Not” (“НЕ”)

Иногда операцию “Not” (“НЕ”) называют “обратным преобразованием.” Эта операция имеет только одно входное значение. Если это входное значение является "истиной", выходному значению присваивается "ложь"; если входное значение "ложь", выходному значению присваивается "истина". Входное значение, попросту, меняется на противоположное.

Вход А

Выход

Истина (1)

Ложь (0)

Ложь (0)

Истина (1)

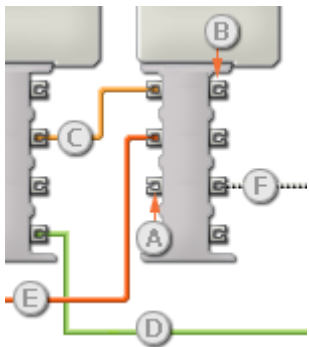
## Настройка концентратора данных программного блока логики

Вы можете управлять блоком логики динамически, подключая шины данных (от концентраторов данных других блоков) к концентратору данных блока логики.

Поместив блок в рабочую область, откройте концентратор данных блока, щелкнув по кнопке в нижней левой части блока.



Шины данных, передающие входящие данные на блок, подключены к разъемам, находящимся на левой стороне их концентраторов данных. Шины данных, передающие исходящие данные, подключены к разъемам, находящимся на правой стороне.



[A] Входной разъем

[B] Выходной разъем

[C] Числовая шина данных (желтая)

[D] Логическая шина данных (зеленая)

[E] Текстовая шина данных (оранжевая)

[F] Поврежденная шина данных (серая)

## Передача данных от входного разъема на выходной разъем

Если для входного разъема есть соответствующий выходной разъем (см. выше А), входящие данные будут проходить от входного разъема на выходной разъем без изменений. В этом случае вы сможете использовать выходной разъем, если входной разъем подключен к шине данных входа; подключение шины данных выхода к такому выходному разъему без подключенной шины данных входа приведет к «неисправности» шины данных выхода (она будет окрашена в серый цвет).

## Шины данных осуществляют передачу специфических типов данных

Каждая шина данных осуществляет обмен специфическими типами данных между блоками. Например, если шина данных перетянута с логического разъема на концентратор данным блока, ее можно подключить только к логическому разъему на концентраторе данных другого блока. На приведенной ниже схеме показано, какой вид данных может принимать или передавать каждый из разъемов.

## Окраски шин данных

Шины данных различаются особыми окрасками: шины, передающие числовые данные, имеют желтый цвет; шины, передающие логические данные, имеют зеленый цвет; шины, передающие текстовые данные, имеют оранжевый цвет.

## «Поврежденные» шины данных

Если вы пытаетесь подключить шину данных к разъему несоответствующего типа данных, шина будет повреждена (ее цвет изменится на серый). Вы не сможете загрузить вашу программу, если шина данных повреждена.

Если вы щелкнете по поврежденной шине, то в небольшом окне контекстной справки в правом нижнем углу рабочей области вы сможете прочесть информацию о причине повреждения.

## Данные должны соответствовать возможному диапазону, в котором работает разъем

Если шина данных входа передает значение, не попадающее в допустимый диапазон разъема, к которому она подключена, блок либо проигнорирует такие данные, либо трансформирует данные, попадающие в этот диапазон. Для разъемов, допускающих небольшое количество значений (например, только 0, 1 или 2), разъем проигнорирует входящий сигнал, если передаваемое значение не будет вписываться в диапазон разъема.

Для разъемов, допускающих более широкий диапазон значений (например, 0 – 100), разъем усилит любой входящий сигнал, если он не соответствует диапазону. Например, если разъем питания программного блока перемещения получает входящий сигнал со значением 150, блок преобразует входящее значение в 100 (т.е., в числовое значение, лежащее в пределах диапазона разъема мощности).

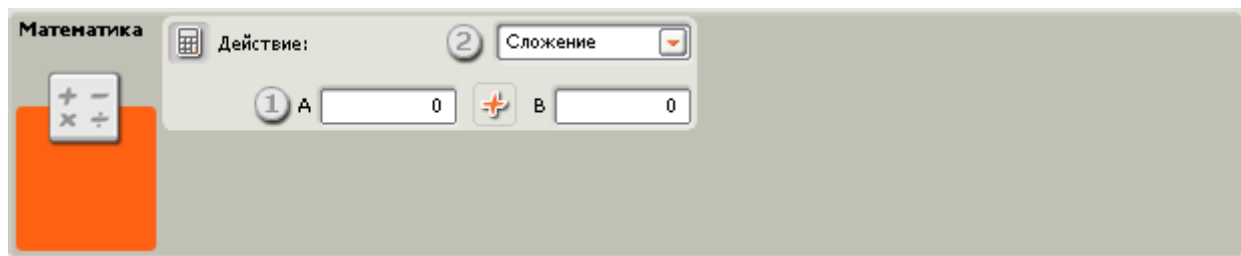
## Математический программный блок

Этот блок выполняет такие простые арифметические действия, как сложение, вычитание, умножение и деление. Кроме того, этот блок может вычислять абсолютное значение и квадратный корень. Числовые значения могут быть введены путем набора или динамически, при помощи шин данных.



1. Математический программный блок, перемещенный в рабочую область, получит открытый концентратор данных с двумя разъемами ввода (слева) и тремя разъемами вывода (справа). Числовые значения для арифметического уравнения могут быть введены путем набора или динамически, при помощи шин данных.

Выходной сигнал из уравнения будет подаваться с самого нижнего выходного разъема; подключите этот разъем при помощи шины данных к концентратору данных другого блока. Два выходных разъема напротив входных разъемов позволяют Вам передавать числовые данные входа на другие блоки, если это потребуется. (Чтобы получить дополнительную информацию, см. ниже раздел "Концентраторы данных".)



1. Входящие значения могут быть введены путем набора или динамически, при помощи шин данных. При подключении шин данных поля ввода будут деактивированы (подсвечены серым).
2. В выпадающем меню Вам будет предложено выбрать одно из четырех арифметических действий, которые Вы можете выполнить:
  - Сложение(+) [по умолчанию]
  - Вычитание (-)
  - Умножение (x)
  - Деление (/)
  - Абсолютное значение (|X|)
  - Квадратный корень ( $\sqrt{x}$ )

## Сложение

В данном случае два вводных числа будут сложены вместе, и их сумма отправляется на вывод.

Ввод A + Ввод B = Вывод

## Вычитание

В данном случае второе введенное число будет вычтено из первого числа, и их разница будет отправлена на вывод.

Ввод A - Ввод B = Вывод

## Умножение

В данном случае два введенных числа будут перемножены, и итог отправляется на вывод.

Ввод A x Ввод B = Вывод

## Деление

В данном случае первое введенное число будет разделено на второе введенное число, и их частное будет отправлено на вывод.

Ввод A / Ввод B = Вывод

## Абсолютная величина

В этом случае на выход будет передано абсолютное значение первого исходного числа

Вход A = Выход

## Квадратный корень

В этом случае на выход будет передан квадратный корень первого исходного числа.

Вход A = Выход

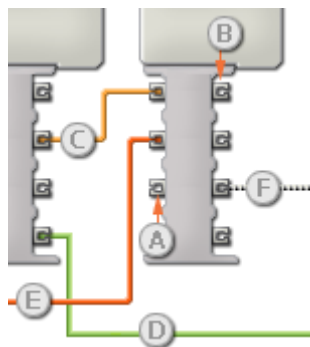
## Настройка концентратора данных математического программного блока

Вы можете управлять математическим программным блоком динамически, подключая шины данных (от концентраторов данных других блоков) к концентратору данных математического программного блока.

Поместив блок в рабочую область, откройте концентратор данных блока, щелкнув по кнопке в нижней левой части блока.



Шины данных, передающие на блок входящие данные, подключены к разъемам, находящимся на левой стороне его концентратора данных. Шины данных, передающие исходящие данные, подключены к разъемам, находящимся на правой стороне.



[A] Входной разъем  
[B] Выходной разъем



- [C] Числовая шина данных (желтая)
- [D] Логическая шина данных (зеленая)
- [E] Текстовая шина данных (оранжевая)
- [F] Поврежденная шина данных (серая)

## **Передача данных от входного разъема на выходной разъем**

Если для входного разъема есть соответствующий выходной разъем (см. выше А), входящие данные будут проходить от входного разъема на выходной разъем без изменений. В этом случае вы сможете использовать выходной разъем, если входной разъем подключен к шине данных входа; подключение шины данных выхода к такому выходному разъему без подключенной шины данных входа приведет к «неисправности» шины данных выхода (она будет окрашена в серый цвет).

## **Шины данных осуществляют передачу специфических типов данных**

Каждая шина данных осуществляет обмен специфическими типами данных между блоками. Например, если шина данных перетянута с логического разъема на концентратор данным блока, ее можно подключить только к логическому разъему на концентраторе данных другого блока. На приведенной ниже схеме показано, какой вид данных может принимать или передавать каждый из разъемов.

## **Окраски шин данных**

Шины данных различаются особыми окрасками: шины, передающие числовые данные, имеют желтый цвет; шины, передающие логические данные, имеют зеленый цвет; шины, передающие текстовые данные, имеют оранжевый цвет.

## **«Поврежденные» шины данных**

Если вы пытаетесь подключить шину данных к разъему несоответствующего типа данных, шина будет повреждена (ее цвет изменится на серый). Вы не сможете загрузить вашу программу, если шина данных повреждена.

Если вы щелкнете по поврежденной шине, то в небольшом окне контекстной справки в правом нижнем углу рабочей области вы сможете прочесть информацию о причине повреждения.

## **Данные должны соответствовать возможному диапазону, в котором работает разъем**

Если шина данных входа передает значение, не попадающее в допустимый диапазон разъема, к которому она подключена, блок либо проигнорирует такие данные, либо трансформирует данные, попадающие в этот диапазон. Для разъемов, допускающих небольшое количество значений (например, только 0, 1 или 2), разъем проигнорирует входящий сигнал, если передаваемое значение не будет вписываться в диапазон разъема.

Для разъемов, допускающих более широкий диапазон значений (например, 0 – 100), разъем усилит любой входящий сигнал, если он не соответствует диапазону. Например, если разъем питания программного блока перемещения получает входящий сигнал со значением 150, блок преобразует входящее значение в 100 (т.е., в числовое значение, лежащее в пределах диапазона разъема мощности).

# **Программный блок случайных событий**

Этот блок генерирует случайное число. Генерирование таких чисел позволяет формировать непредсказуемое поведение вашего робота. Например, если вы используете шину данных для подключения Программного блока случайных событий к разъему Продолжительность концентратора данных программного блока Перемещение, ваш робот один раз может идти вперед 4 секунды, а в следующий раз, когда вы запустите программу - 8 секунд.

Случайное число будет генерироваться в пределах установленных вами минимального и максимального значения включительно. Если вы хотите сгенерировать случайные числа от 5 до 10, включая 5 и 10 как возможные числа, установите минимальное значение равным 5, а максимальное значение равным 10. Если вы не хотите, чтобы число 5 или 10 входило в диапазон возможных случайных чисел, выберите минимальное значение 6, а максимальное - 9.

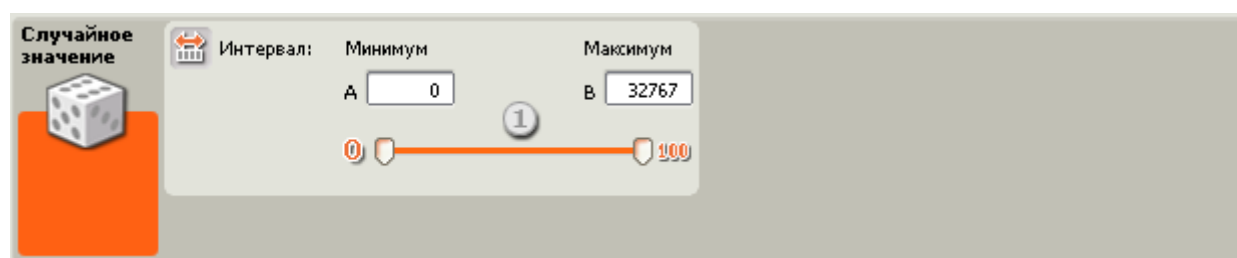
Установите минимальный и максимальный предел при помощи бегунка, или введя эти значения в окне ввода. Для получения динамического номера подсоедините шины входных данных к концентратору данных блока.

Выходные данные блока (то есть, случайное число) могут быть отправлены только через шину данных с разъема выходной величины (#). (Чтобы получить дополнительную информацию, см. ниже раздел "Концентратор данных".)



1. Концентратор данных блока откроется автоматически, когда блок будет присоединен к коммутатору последовательности действий.

## Настройка программного блока случайных событий



1. Установите минимальное и максимальные значения при помощи двух бегунков, или введя эти цифры с клавиатуры. Верхний предел бегунка имеет отметку 100, но если вводить значение верхнего предела в окне ввода, оно может быть больше 100. Вы также можете ввести отрицательное число в окно ввода минимального предела; это заменит значение, установленное бегунком. Для динамической установки минимального и максимального пределов подсоедините шины входных данных к концентратору данных блока.

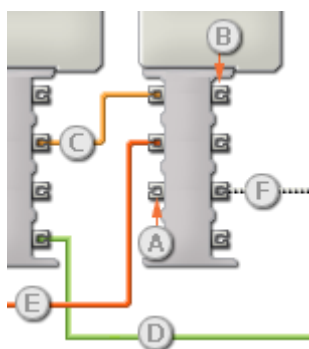
## Настройка концентратора данных программного блока случайных событий

Вы можете управлять блоком случайных событий динамически, подключая шины данных (от концентраторов данных других блоков) к концентратору данных блока случайных событий.

Поместив блок в рабочую область, откройте концентратор данных блока, щелкнув по кнопке в нижней левой части блока.



Шины данных, передающие входящие данные на блок, подключены к разъемам, находящимся на левой стороне их концентраторов данных. Шины данных, передающие исходящие данные, подключены к разъемам, находящимся на правой стороне.



- [A] Входной разъем
- [B] Выходной разъем
- [C] Числовая шина данных (желтая)
- [D] Логическая шина данных (зеленая)
- [E] Текстовая шина данных (оранжевая)
- [F] Поврежденная шина данных (серая)

## Передача данных от входного разъема на выходной разъем

Если для входного разъема есть соответствующий выходной разъем (см. выше A), входящие данные будут проходить от входного разъема на выходной разъем без изменений. В этом случае вы сможете использовать выходной разъем, если входной разъем подключен к шине данных входа; подключение шины данных выхода к такому выходному разъему без подключенной шины данных входа приведет к «неисправности» шины данных выхода (она будет окрашена в серый цвет).

## Шины данных осуществляют передачу специфических типов данных

Каждая шина данных осуществляет обмен специфическими типами данных между блоками. Например, если шина данных перетянута с логического разъема на концентратор данным блока, ее можно подключить только к логическому разъему на концентраторе данных другого блока. На приведенной ниже схеме показано, какой вид данных может принимать или передавать каждый из разъемов.

## Окраски шин данных

Шины данных различаются особыми окрасками: шины, передающие числовые данные, имеют желтый цвет; шины, передающие логические данные, имеют зеленый цвет; шины, передающие текстовые данные, имеют оранжевый цвет.

## «Поврежденные» шины данных

Если вы пытаетесь подключить шину данных к разъему несоответствующего типа данных, шина будет повреждена (ее цвет изменится на серый). Вы не сможете загрузить вашу программу, если шина данных повреждена.

Если вы щелкнете по поврежденной шине, то в небольшом окне контекстной справки в правом нижнем углу рабочей области вы сможете прочесть информацию о причине повреждения.

## Данные должны соответствовать возможному диапазону, в котором работает разъем

Если шина данных входа передает значение, не попадающее в допустимый диапазон разъема, к которому она подключена, блок либо проигнорирует такие данные, либо трансформирует данные, попадающие в этот диапазон. Для разъемов, допускающих небольшое количество значений (например, только 0, 1 или 2), разъем проигнорирует входящий сигнал, если передаваемое значение не будет вписываться в диапазон разъема.

Для разъемов, допускающих более широкий диапазон значений (например, 0 – 100), разъем усилит любой входящий сигнал, если он не соответствует диапазону. Например, если разъем питания программного блока перемещения получает входящий сигнал со значением 150, блок преобразует входящее значение в 100 (т.е., в числовое значение, лежащее в пределах диапазона разъема мощности).

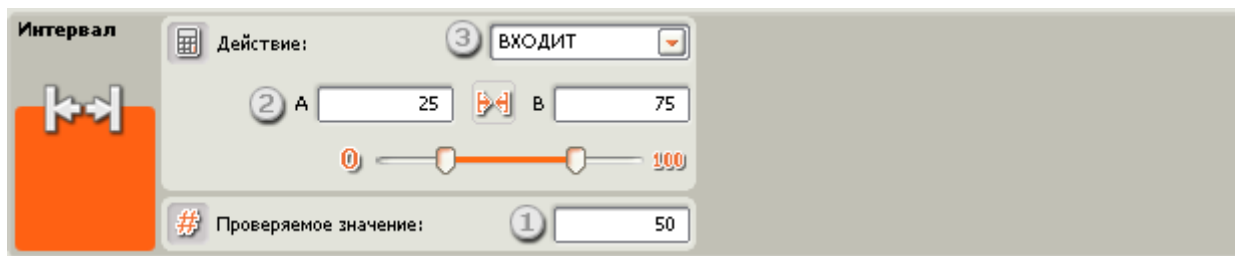
## Блок «Интервал»

Этот блок может определить, принадлежит ли число определенному интервалу чисел, или находится вне этого интервала. Числовые значения можно ввести путем набора, задать при помощи движка или ввести динамически, при помощи шин данных. Выходной логический сигнал ("истина"/"ложь") посылается через шину данных..



1. У Блока «Интервал», встроенного в цепочку программы, будет открыт концентратор данных с тремя входными разъёмами (слева) и четырьмя выходными разъёмами (справа). К входным разъёмам можно подсоединять другие программные блоки посредством шин данных (кроме случаев, когда одному или нескольким входным разъёмам задается введенное вами число).

Блок «Интервал» отправляет логическую команду с самого нижнего выходного разъёма; подключите этот разъём шиной данных к концентратору данных другого блока. Три выходных разъёма напротив входных разъёмов позволяют вам передавать значения входа на другие блоки, если это потребуется. (Дополнительную информацию см. в разделе "Концентратор данных".)



1. Тестовое числовое значение можно набрать или ввести динамически при помощи шины данных. При подсоединении шины данных тестовое поле будет деактивировано (станет серым).
2. Верхний и нижний пределы интервала можно задать движком, ввести с клавиатуры или динамически, посредством шин данных. При подключении шин данных поля ввода будут деактивированы (станут серыми).
3. В открывающемся списке можно выбрать одну из двух операций:
  - В пределах интервала ([ ])
  - За пределами интервала ( [ )

### В пределах интервала

В этом случае, если тестовое число попадает в интервал или равно верхнему или нижнему предельному значению, Блок «Интервал» подтвердит это число как “истину”. В противном случае он сообщит о том, что это значение - “ложь”.

### За пределами интервала

В этом случае, если тестовое число не попадает в интервал (и не равно верхнему или нижнему предельному значению), Блок «Интервал» подтвердит это значение как “истину”. В противном случае он сообщит о том, что это значение - “ложь”.

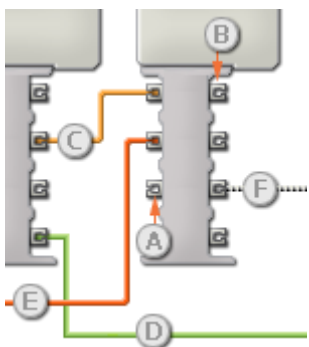
## Настройка концентратора данных Блока «Интервал»

Блоком «Интервал» можно управлять динамически, подключая к его концентратору данных шины данных (от концентраторов данных других блоков).

Перетаскив блок в рабочую область, откройте его концентратор данных, щёлкнув на кнопке в нижней левой части блока.



Шины данных, передающие входящие данные на блок, подключены к разъёмам, расположенным на левой стороне их концентраторов данных. Шины данных, передающие исходящие данные, подключены к разъёмам, расположенным справа.



- [A] Входной разъём
- [B] Выходной разъём
- [C] Числовая шина данных (желтая)
- [D] Логическая шина данных (зеленая)
- [E] Текстовая шина данных (оранжевая)
- [F] Повреждённая шина данных (серая)

## **Передача данных от входного разъёма на выходной разъём**

Если для входного разъёма имеется соответствующий выходной разъём (см. выше A), входящие данные будут проходить от входного разъёма на выходной разъём без изменений. В этом случае можно использовать выходной разъём, если входной разъём подключен к шине данных входа; подключение шины данных выхода к такому выходному разъёму без подключенной шины данных входа приведет к «повреждению» шины данных выхода (она станет серой).

## **По шинам данных передаются данные различных типов**

Через каждую шину данных осуществляется обмен данными различных типов между блоками. Например, если шина данных протянута с логического разъёма на концентратор данных блока, её можно подключить только к логическому разъёму на концентраторе данных другого блока. На приведенной ниже схеме показано, данные какого типа может принимать или передавать каждый из разъёмов.

## **Цвет шин данных**

Шины данных окрашены в различные цвета: шины для передачи числовых данных - желтые; шины, для передачи логических данных - зеленые; шины для передачи текстовых данных - оранжевые.

## **«Повреждённые» шины данных**

При попытке подключить шину данных к разъёму данных несоответствующего типа, шина будет повреждена (ее цвет изменится на серый). Вы не сможете загрузить программу, если шина данных повреждена.

Если щёлкнуть на повреждённой шине, то в небольшом окне контекстной справки в правом нижнем углу рабочей области появится информация о причине повреждения.

## **Данные должны соответствовать возможному диапазону, в котором работает разъём**

Если шина данных входа передает значение, не попадающее в допустимый диапазон разъёма, к которому она подключена, блок либо проигнорирует такие данные, либо приведёт их к значению, попадающему в этот диапазон. Если разъём допускает ограниченное количество значений (например, только 0, 1 или 2), он проигнорирует входящий сигнал, если передаваемое значение не будет вписываться в допустимый диапазон.

Если разъём допускает более широкий диапазон значений (например, 0 – 100), любой входящий сигнал, не попадающий в допустимый диапазон, будет приведен к значению, соответствующее ему. Например, если разъём мощности блока «Движение» получает входящий сигнал со значением 150, блок преобразует входящее значение в 100 (т.е., в числовое значение, лежащее в пределах диапазона разъёма мощности ).

# Блок «Переменная»

Вообразите, что переменная - это место хранения значения в памяти NXT. Другие программные блоки могут считывать текущее значение переменной (и даже изменять его), связываясь с Блоком «Переменная» через шины данных.

## Создание переменной

1. Чтобы создать переменную, сначала выберите команду «Редактор переменных» в меню Правка. В открывшемся окне Редактор переменных задайте новой переменной короткое и понятное имя, и укажите тип данных, который эта переменная будет поддерживать (т.е. текстовые, числовые или логические).
2. Затем перетащите Блок «Переменная» в свою программу и выберите имя переменной из списка.
3. Выберите режим "считать" либо "записать" переменную. Выбор режима записи переменной позволит вам в дальнейшем изменять значение переменной, записывая ее новое значение вместо прежнего. Для этого нужно протянуть шину входных данных к концентратору данных блока, из которого будет поступать новая информация.
4. Можно задать постоянное исходное значение переменной, введя его с клавиатуры в поле (или кнопками выбора, если это логическая переменная).

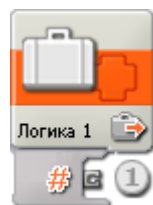


Примечание: если потребуется изменить тип данных или имя переменной, вернитесь к команде «Редактор переменных» в меню Правка, и внесите изменения в настройки.

## Использование переменной

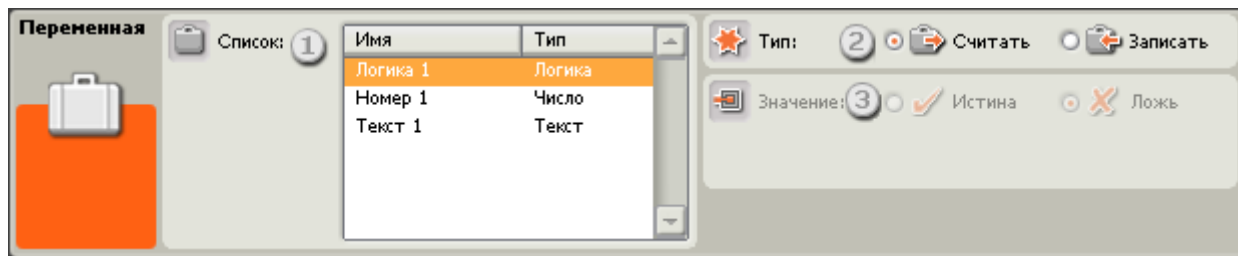
Значение переменной со временем может меняться. Например, можно создать переменную под названием "Зарегистрированный уровень освещённости" и задать ей исходное значение 50. И пока робот с датчиком освещённости движется к месту назначения, при помощи Блока «Переменная» (с настройкой "записать"), можно зафиксировать уровень освещённости в этом месте. Для этого подключите шину выходных данных от Блока «Датчик освещённости» к входному разъёму Блока «Переменная».

Впоследствии в программе можно будет использовать Блок «Переменная» (с настройкой "считать") для регистрации уровня освещённости в этом месте. Другие блоки, подсоединенные к выходному разъёму блока посредством шины данных, могут получать доступ к информации (т.е. зарегистрированному уровню освещённости), сохраненной в переменной.



1. После того как Блок «Переменная» перетащен в рабочую область, его концентратор данных открывается автоматически. Чтобы другие блоки могли считывать текущее значение переменной, к выходному разъёму этого блока нужно протянуть шину данных. Чтобы записать новое значение переменной, необходимо подключить шину входных данных к концентратору данных Блока «Переменная». (Дополнительную информацию см. в разделе "Концентратор данных".)

## Настройка Блока «Переменная»:



1. Выберите из списка переменную, которую вы создали (меню Правка > Редактор переменных).
2. Выберите режим считывания текущего значения переменной или режим записи нового значения переменной.
3. Чтобы присвоить переменной постоянное значение, введите его с клавиатуры (число или текст), или задайте кнопками выбора, если переменная относится к типу логических данных. Помните, что если к блоку подсоединена шина входных данных, а Блок «Переменная» настроен на режим "считать", новая информация, динамически поступающая через шину данных, будет иметь приоритет перед любым постоянным значением.



Примечание: если потребуется изменить тип данных или имя переменной, вернитесь к команде «Редактор переменных» в меню Правка, и внесите изменения в настройки.

## Совместное использование данных Программой и Моим Блоком

Чтобы сделать возможным совместное использование данных основной программой и Моим Блоком (т.е. двумя файлами с расширением .rbt), произведите два действия:

1. Определите переменную с одним и тем же именем и типом данных в основной программе и в Моём Блоке.
2. Используйте эту переменную в Блоке «Переменная» основной программы и в Моём Блоке.

Где бы эта Переменная ни использовалась, считывание/запись данных будет производиться в одних и тех же ячейках памяти NXT.

## Настройка концентратора данных Блока «Переменная»

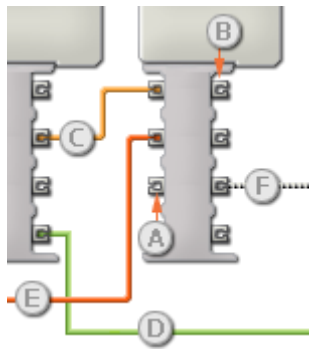
Блоком «Переменная» можно управлять динамически, подключая к его концентратору данных шины данных (от концентраторов данных других блоков).

Перетащив блок в рабочую область, откройте его концентратор данных, щёлкнув на кнопке в нижней левой части блока.



Шины данных, передающие входящие данные на блок, подключены к разъёмам, расположенным на левой стороне их концентраторов данных. Шины данных, передающие исходящие данные, подключены к разъёмам, расположенным справа.





- [A] Входной разъём
- [B] Выходной разъём
- [C] Числовая шина данных (желтая)
- [D] Логическая шина данных (зеленая)
- [E] Текстовая шина данных (оранжевая)
- [F] Повреждённая шина данных (серая)

## Передача данных от входного разъёма на выходной разъём

Если для входного разъёма имеется соответствующий выходной разъём (см. выше А), входящие данные будут проходить от входного разъёма на выходной разъём без изменений. В этом случае можно использовать выходной разъём, если входной разъём подключен к шине данных входа; подключение шины данных выхода к такому выходному разъёму без подключенной шины данных входа приведет к «повреждению» шины данных выхода (она станет серой).

## По шинам данных передаются данные различных типов

Через каждую шину данных осуществляется обмен данными различных типов между блоками. Например, если шина данных протянута с логического разъёма на концентратор данных блока, её можно подключить только к логическому разъёму на концентраторе данных другого блока. На приведенной ниже схеме показано, данные какого типа может принимать или передавать каждый из разъёмов.

## Цвет шин данных

Шины данных окрашены в различные цвета: шины для передачи числовых данных - желтые; шины, для передачи логических данных - зеленые; шины для передачи текстовых данных - оранжевые.

## «Повреждённые» шины данных

При попытке подключить шину данных к разъёму данных несоответствующего типа, шина будет повреждена (ее цвет изменится на серый). Вы не сможете загрузить программу, если шина данных повреждена.

Если щёлкнуть на повреждённой шине, то в небольшом окне контекстной справки в правом нижнем углу рабочей области появится информация о причине повреждения.

## Данные должны соответствовать возможному диапазону, в котором работает разъём

Если шина данных входа передает значение, не попадающее в допустимый диапазон разъёма, к которому она подключена, блок либо проигнорирует такие данные, либо приведёт их к значению, попадающему в этот диапазон. Если разъём допускает ограниченное количество значений (например, только 0, 1 или 2), он проигнорирует

входящий сигнал, если передаваемое значение не будет вписываться в допустимый диапазон.

Если разъем допускает более широкий диапазон значений (например, 0 – 100), любой входящий сигнал, не попадающий в допустимый диапазон, будет приведен к значению, соответствующее ему. Например, если разъем мощности блока «Движение» получает входящий сигнал со значением 150, блок преобразует входящее значение в 100 (т.е., в числовое значение, лежащее в пределах диапазона разъема мощности ).

## Блок «Константа»

Представим константу как место хранения значения в памяти NXT. Другие программные блоки могут считывать текущее значение константы, связываясь с этим блоком через шины данных.

Данный программный блок работает так же, как и Блок «Переменная», но созданные с его помощью константы являются глобальными (т.е. общими) для всех программ в вашей среде программирования NXT. Если задать константу используя команду Редактор констант в меню Правка, её можно будет использовать во всех программах. Константы, например, число Пи (3,1415), никогда не изменяются со временем, и являются математическими постоянными.

### Содержание

- [Настройки отображения](#)
- [Редактирование констант](#)
- [Настройка Блока «Константа»](#)
- [Совместное использование данных программой и Моим Блоком](#)
- [Настройка концентратора данных Блока «Константа»](#)

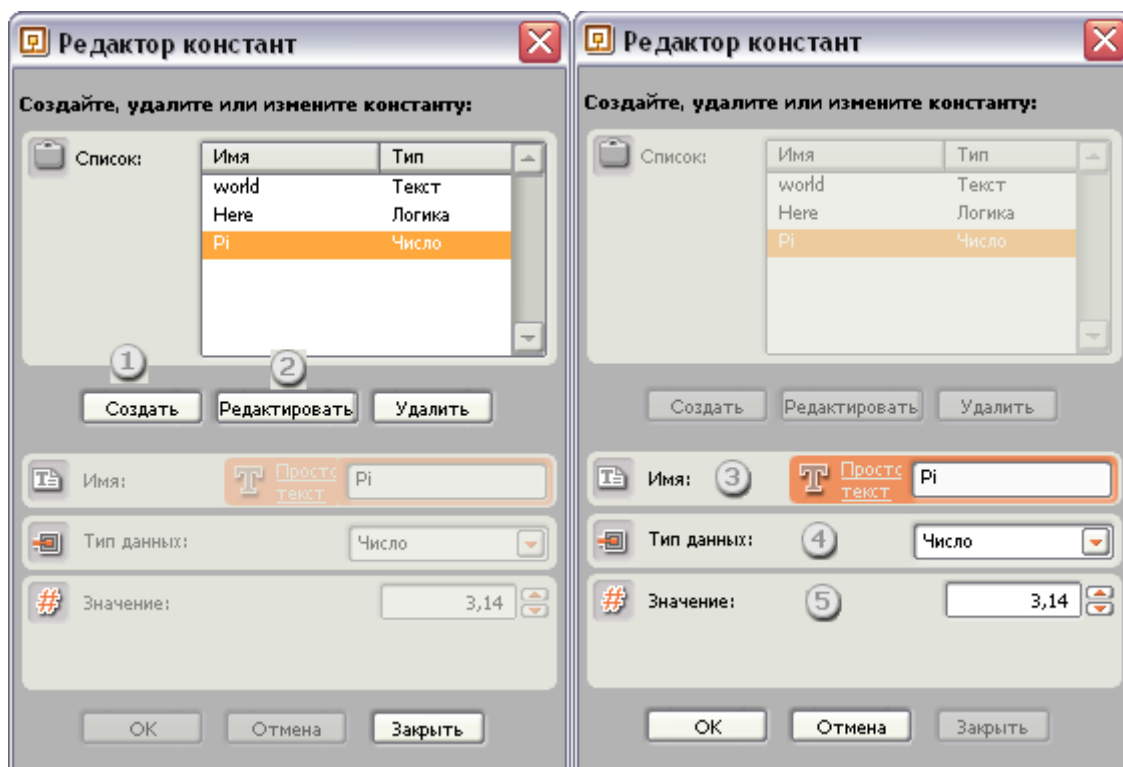
### Отображение настроек



1. Имя Блока «Константа»
2. Когда программный блок перетасчен в рабочую область, его концентратор данных открывается автоматически. Хотя бы одну шину данных следует перетащить с выходного разъема этого блока на концентратор данных другого блока. (Дополнительную информацию см. в разделе "Концентратор данных".)

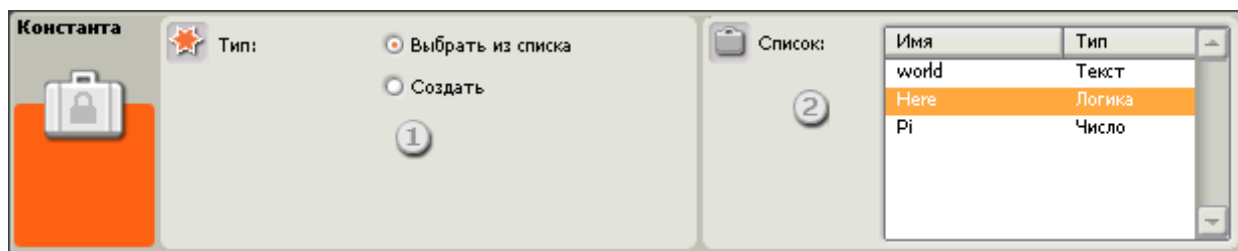
### Редактирование констант

Чтобы создать новую константу, сначала выберите команду Редактор констант в меню Правка.

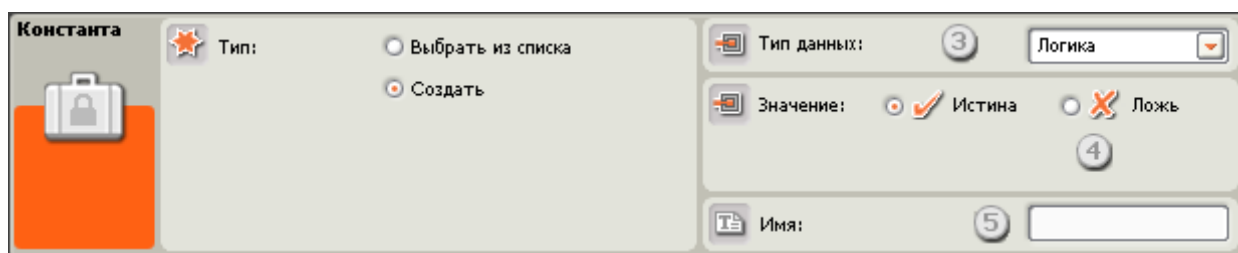


1. Для того чтобы создать новую константу, выберите Создать.
2. Для того чтобы изменить или удалить существующую константу, выделите ее в списке и выберите Редактировать или Удалить.
3. Присвойте новой константе короткое и понятное имя.
4. Задайте тип данных, содержащихся в константе (текст, число или логика).
5. В зависимости от выбранного типа данных задайте величину константы.

## Настройка Блока «Константа»



1. Выберите константу из списка существующих или создайте её сами.
2. Выберите константу из списка имеющихся констант. Для того чтобы создать новую константу и занести её в этот список, используйте команду Редактор констант из меню Правка.



3. Задайте тип данных, содержащихся в пользовательской константе (текст, число или логика).
4. Введите текстовое, числовое или логическое значение в зависимости от выбранного типа.
5. В соответствующее поле введите имя пользовательской константы.

# Совместное использование данных программой и Моим Блоком

Чтобы сделать возможным совместное использование данных основной программой и Моим Блоком (т.е. двумя файлами .rbt), необходимо выполнить следующее:

1. Определите константу с одним и тем же именем и типом данных в основной программе и в Моём Блоке.
2. Используйте эту константу в Блоке «Константа» в основной программе и в Моём Блоке.

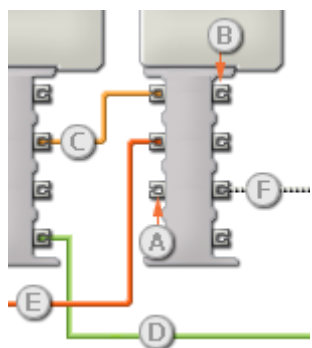
## Настройка концентратора данных Блока «Константа»

Блоком «Константа» можно управлять динамически, подключая к его концентратору данных шины данных (от концентраторов данных других блоков).

Перетаскив блок в рабочую область, откройте его концентратор данных, щёлкнув на кнопке в нижней левой части блока.



Шины данных, передающие входящие данные на блок, подключены к разъёмам, расположенным на левой стороне их концентраторов данных. Шины данных, передающие исходящие данные, подключены к разъёмам, расположенным справа.



## Передача данных от входного разъёма на выходной разъём

Если для входного разъёма имеется соответствующий выходной разъём (см. выше А), входящие данные будут проходить от входного разъёма на выходной разъём без изменений. В этом случае можно использовать выходной разъём, если входной разъём подключен к шине данных входа; подключение шины данных выхода к такому выходному разъёму без подключенной шины данных входа приведет к «повреждению» шины данных выхода (она станет серой).

## По шинам данных передаются данные различных типов

Через каждую шину данных осуществляется обмен данными различных типов между блоками. Например, если шина данных протянута с логического разъёма на концентратор данных блока, её можно подключить только к логическому разъёму на концентраторе данных другого блока. На приведенной ниже схеме показано, данные какого типа может принимать или передавать каждый из разъёмов.

## Цвет шин данных

Шины данных окрашены в различные цвета: шины для передачи числовых данных - желтые; шины, для передачи логических данных - зеленые; шины для передачи текстовых данных - оранжевые.

## Повреждённые шины данных

При попытке подключить шину данных к разъёму данных несоответствующего типа, шина будет повреждена (ее цвет изменится на серый). Вы не сможете загрузить программу, если шина данных повреждена.

Если щёлкнуть на повреждённой шине, то в небольшом окне контекстной справки в правом нижнем углу рабочей области появится информация о причине повреждения.

## Данные должны соответствовать возможному диапазону, в котором работает разъём

Если шина данных входа передает значение, не попадающее в допустимый диапазон разъёма, к которому она подключена, блок либо проигнорирует такие данные, либо приведёт их к значению, попадающему в этот диапазон. Если разъём допускает ограниченное количество значений (например, только 0, 1 или 2), он проигнорирует входящий сигнал, если передаваемое значение не будет вписываться в допустимый диапазон.

Если разъём допускает более широкий диапазон значений (например, 0 – 100), любой входящий сигнал, не попадающий в допустимый диапазон, будет приведен к значению, соответствующее ему. Например, если разъём мощности блока «Движение» получает входящий сигнал со значением 150, блок преобразует входящее значение в 100 (т.е., в числовое значение, лежащее в пределах диапазона разъёма мощности ).

Блоки группы Дополнения

- [Блок «Калибровка»](#)
- [Блок «Доступ к файлам»](#)
- [Блок «Не засыпать»](#)
- [Блок «Число в текст»](#)
- [Блок «Обнулить мотор»](#)
- [Блок «Текст»](#)
- [Блок «Bluetooth соединение»](#)
- [Блок «Начать регистрацию данных»](#)
- [Блок «Остановить регистрацию данных»](#)

## Блок «Калибровка»


Этот блок служит для калибровки минимального (0%) и максимального (100%) показаний датчика звука или датчика освещённости.

Чтобы полностью откалибровать датчик звука или датчик освещённости потребуется два Блока «Калибровка»: один для установки минимального значения для датчика и один для установки максимального значения. Не требуется производить калибровку обоих значений (минимального и максимального); достаточно задать лишь одно из них, другое значение будет установлено по умолчанию.

Полностью откалибровать датчик звука или датчик освещённости можно несколькими способами. 1. Установите в начале программы два Блока «Калибровка», а после каждого из них - Блок «Ожидание» (сконфигурированный на ожидание сигнала от датчика касания. Например, если при калибровке датчика освещённости, установить первый Блок «Калибровка» для минимального значения, а второй Блок «Калибровка» для максимального значения, процедура калибровки будет выглядеть следующим образом:

1. Поместите робота в наиболее темное место, в котором он может оказаться при выполнении программы, и нажмите на датчик касания. При этом датчик освещённости будет откалиброван на минимальное значение.
2. Затем поместите робота в наиболее освещённое место, в котором он может оказаться во время выполнения программы, и нажмите на датчик касания. Датчик освещённости будет откалиброван на максимальное значение, и робот продолжит выполнение программы.

Датчик также можно откалибровать, вставив Блоки «Калибровка» в определенные места программы. Например, если робот с датчиком освещённости должен проехать под столом (и это будет самым темным местом, в котором появится робот), Блок «Калибровка» (настроенный для калибровки минимального показания датчика освещённости) можно вставить в программу так, чтобы он активировался во время проезда робота под столом.

 **Примечание:** калибруется датчик, а не порт. Например, если этот блок используется для считывания показаний датчика освещённости на порте 2, и это значение регистрируется как максимальное, то ВСЕ датчики освещённости будут считать это значение максимальным (независимо от того, подключены ли они к порту 2, или нет).

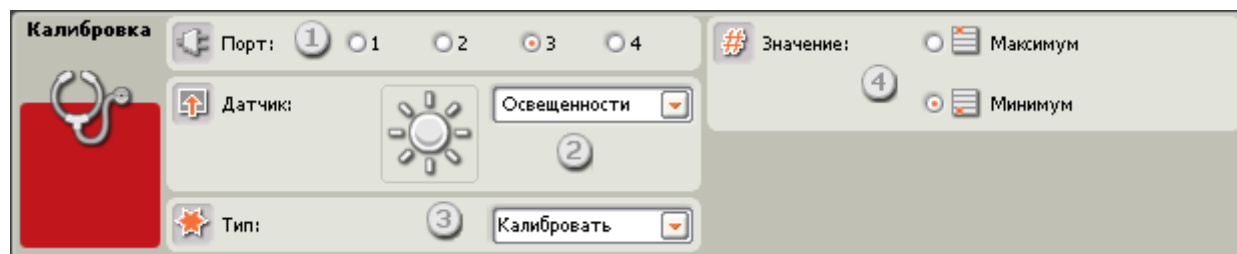
Важно: помните, что для калибровки датчиков освещённости и звука можно также использовать и функцию "Калибровка датчиков" из меню "Инструменты".  
Дополнительную информацию см. в разделе "Калибровать датчики".

## Отображение параметров настройки блока



1. После того как Программный блок переташен в рабочую область, его концентратор данных открывается автоматически. При помощи шин данных можно динамически настроить множество параметров. (Дополнительную информацию см. в разделе "Концентратор данных".)

## Настройка Блока «Калибровка»



1. Выберите порт, к которому подключен датчик.
2. Выберите объект калибровки - датчик звука или датчик освещённости. (Если вы калибруете более раннюю модель датчика освещённости MINDSTORMS с переходным кабелем, выберите “Датчик освещённости\*”).
3. При помощи этого списка выберите действие - калибровать на новое значение или удалить ранее установленное значение.
4. Кнопками выбора установите, максимальное или минимальное значение должен получить датчик.

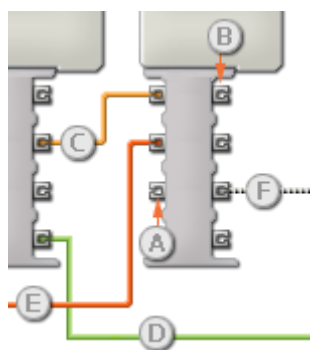
## Настройка концентратора данных Блока «Калибровка»

Блоком «Калибровка» можно управлять динамически, протягивая шины данных (от концентраторов данных других блоков) к концентратору данных Блока «Калибровка».

Перетащив блок в рабочую область, откройте его концентратор данных, щёлкнув на кнопке в нижней левой части блока.



Шины данных, передающие входящие данные на блок, подключены к разъёмам, расположенным на левой стороне их концентраторов данных. Шины данных, передающие исходящие данные, подключены к разъёмам, расположенным справа.



- [A] Входной разъём
- [B] Выходной разъём
- [C] Числовая шина данных (желтая)
- [D] Логическая шина данных (зеленая)
- [E] Текстовая шина данных (оранжевая)
- [F] Повреждённая шина данных (серая)

### Передача данных от входного разъёма на выходной разъём

Если для входного разъёма имеется соответствующий выходной разъём (см. выше A), входящие данные будут проходить от входного разъёма на выходной разъём без изменений. В этом случае можно использовать только выходной разъём, если входной разъём подключен к шине данных входа; подключение шины данных выхода к такому выходному разъёму без подключенной шины данных входа приведет к «повреждению» шины данных выхода (она станет серой).

### По шинам данных передаются данные различных типов

Через каждую шину данных осуществляется обмен данными различных типов между блоками. Например, если шина данных протянута с логического разъёма на концентратор данных блока, её можно подключить только к логическому разъёму на концентраторе данных другого блока. На приведенной ниже схеме показано, данные какого типа может принимать или передавать каждый из разъёмов.

## **Цвет шин данных**

Шины данных окрашены в различные цвета: шины для передачи числовых данных - желтые; шины, для передачи логических данных - зеленые; шины для передачи текстовых данных - оранжевые.

## **«Повреждённые» шины данных**

При попытке подключить шину данных к разъёму данных несоответствующего типа, шина будет повреждена (ее цвет изменится на серый). Вы не сможете загрузить программу, если шина данных повреждена.

Если щёлкнуть на повреждённой шине, то в небольшом окне контекстной справки в правом нижнем углу рабочей области появится информация о причине повреждения.

## **Данные должны соответствовать возможному диапазону, в котором работает разъём**

Если шина данных входа передает значение, не попадающее в допустимый диапазон разъёма, к которому она подключена, блок либо проигнорирует такие данные, либо приведёт их к значению, попадающему в этот диапазон. Если разъём допускает ограниченное количество значений (например, только 0, 1 или 2), он проигнорирует входящий сигнал, если передаваемое значение не будет вписываться в допустимый диапазон.

Если разъём допускает более широкий диапазон значений (например, 0 – 100), любой входящий сигнал, не попадающий в допустимый диапазон, будет приведен к значению, соответствующее ему. Например, если разъём мощности блока «Движение» получает входящий сигнал со значением 150, блок преобразует входящее значение в 100 (т.е., в числовое значение, лежащее в пределах диапазона разъёма мощности ).

## **Блок «Файл»**

Благодаря этому блоку можно сохранять данные, поступающие с робота, в файлах NXT. После сохранения данных в файл необходимо использовать другой блок «Файл», чтобы закрыть этот файл, прежде чем прочесть его или удалить (для удаления потребуется третий блок «Файл»).

Например, чтобы прочитать только что записанную в файл информацию, этот файл сначала следует закрыть. Для этого потребуются три Блока «Файл»: первый (с функцией "Записать", заданной на его панели настроек) будет записывать данные в файл; второй блок, вставленный в программу немного дальше, будет закрывать этот файл; третий Блок «Файл» (с функцией "Прочитать" заданной на его панели настроек) будет считывать из файла информацию. Эти три блока могут следовать друг за другом или находиться в разных местах программы.

При записи в уже существующий файл, данные будут добавлены в его конец; записанные ранее данные удалены не будут. Чтобы перезаписать файл, используйте

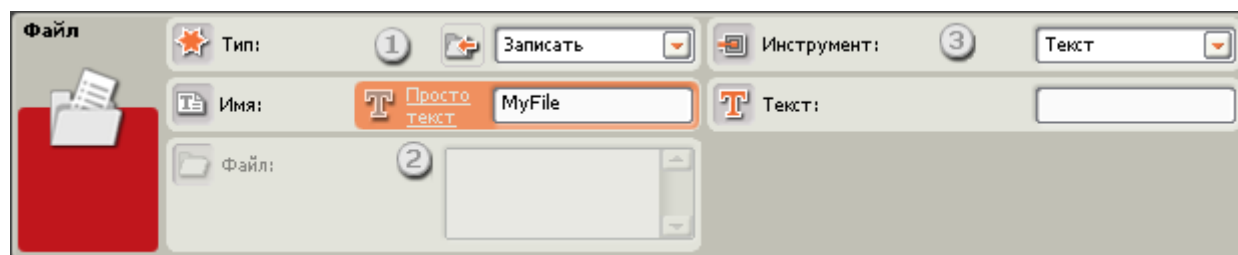


Блок «Файл», настроенный на удаление. Затем потребуется ещё один Блок «Файл», чтобы записать новый файл с таким же именем.



1. Эта пиктограмма показывает настройку Блока «Файл» - Записать, Прочсть, Закреть или Удалить файл.

## Настройка Блока «Файл»



1. В открывающемся списке выберите операцию, которую необходимо произвести с файлом - Записать, Прочсть, Закреть или Удалить.
2. Введите название файла (при первоначальной записи в него) или укажите название существующего файла, введя его в поле или выбрав в открывающемся списке.
3. Выберите формат данных - Текст или Число.

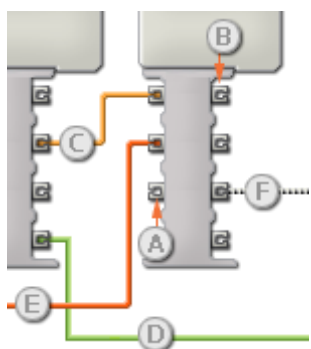
## Настройка концентратора данных Блока «Файл»

Блоком «Файл» можно управлять динамически, подключая шины данных (от концентраторов данных других блоков) к концентратору данных Блока «Файл».

Перетащите блок в рабочую область и откройте его концентратор данных, щёлкнув на кнопке в нижней левой части блока.



Шины данных, передающие входящие данные на блок, подключены к разъёмам, расположенным на левой стороне их концентраторов данных. Шины данных, передающие исходящие данные, подключены к разъёмам, расположенным справа.



- [A] Входной разъём
- [B] Выходной разъём
- [C] Числовая шина данных (желтая)
- [D] Логическая шина данных (зеленая)
- [E] Текстовая шина данных (оранжевая)
- [F] Повреждённая шина данных (серая)

## **Передача данных от входного разъёма на выходной разъём**

Если для входного разъёма имеется соответствующий выходной разъём (см. выше А), входящие данные будут проходить от входного разъёма на выходной разъём без изменений. В этом случае можно использовать выходной разъём, если входной разъём подключен к шине данных входа; подключение шины данных выхода к такому выходному разъёму без подключенной шины данных входа приведет к «неисправности» шины данных выхода (она будет окрашена в серый цвет).

## **По шинам данных передаются данные различных типов**

Через каждую шину данных осуществляется обмен данными различных типов между блоками. Например, если шина данных протянута с логического разъёма на концентратор данных блока, её можно подключить только к логическому разъёму на концентраторе данных другого блока. На приведенной ниже схеме показано, данные какого типа может принимать или передавать каждый из разъёмов.

## **Цвет шин данных**

Шины данных окрашены в различные цвета: шины для передачи числовых данных - желтые; шины, для передачи логических данных - зеленые; шины для передачи текстовых данных - оранжевые.

## **«Повреждённые» шины данных**

При попытке подключить шину данных к разъёму данных несоответствующего типа, шина будет повреждена (ее цвет изменится на серый). Вы не сможете загрузить программу, если шина данных повреждена.

Если щёлкнуть на повреждённой шине, то в небольшом окне контекстной справки в правом нижнем углу рабочей области появится информация о причине повреждения.

## **Данные должны соответствовать возможному диапазону, в котором работает разъём**

Если шина данных входа передает значение, не попадающее в допустимый диапазон разъёма, к которому она подключена, блок либо проигнорирует такие данные, либо приведёт их к значению, попадающему в этот диапазон. Если разъём допускает ограниченное количество значений (например, только 0, 1 или 2), он проигнорирует входящий сигнал, если передаваемое значение не будет вписываться в допустимый диапазон.

Если разъём допускает более широкий диапазон значений (например, 0 – 100), любой входящий сигнал, не попадающий в допустимый диапазон, будет приведен к значению, соответствующее ему. Например, если разъём мощности блока «Движение» получает входящий сигнал со значением 150, блок преобразует входящее значение в 100 (т.е., в числовое значение, лежащее в пределах диапазона разъёма мощности ).

# Блок «Не засыпать»

Этот блок не допускает переключения NXT в спящий режим. Добавьте этот блок, если необходимо, чтобы программа была активна дольше заданного периода перехода NXT в "спящий режим" (настраивается через меню NXT).



## Настройка Блока «Не засыпать»



У Блока «Не засыпать» нет настраиваемых параметров.

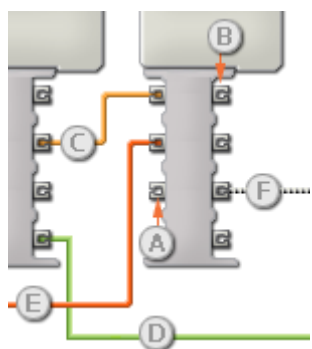
## Настройка концентратора данных Блока «Не засыпать»

Блоком «Не засыпать» можно управлять динамически, подключая шины данных (от концентраторов данных других блоков) к концентратору данных Блока «Не засыпать».

Перетаскив блок в рабочую область, откройте его концентратор данных, щёлкнув на кнопке в нижней левой части блока.



Шины данных, передающие входящие данные на блок, подключены к разъёмам, расположенным на левой стороне их концентраторов данных. Шины данных, передающие исходящие данные, подключены к разъёмам, расположенным справа.



- [A] Входной разъём
- [B] Выходной разъём
- [C] Числовая шина данных (желтая)
- [D] Логическая шина данных (зеленая)
- [E] Текстовая шина данных (оранжевая)
- [F] Повреждённая шина данных (серая)

## **Передача данных от входного разъёма на выходной разъём**

Если для входного разъёма имеется соответствующий выходной разъём (см. выше A), входящие данные будут проходить от входного разъёма на выходной разъём без изменений. В этом случае можно использовать выходной разъём, если входной разъём подключен к шине данных входа; подключение шины данных выхода к такому выходному разъёму без подключенной шины данных входа приведет к «повреждению» шины данных выхода (она станет серой).

## **По шинам данных передаются данные различных типов**

Через каждую шину данных осуществляется обмен данными различных типов между блоками. Например, если шина данных протянута с логического разъёма на концентратор данных блока, её можно подключить только к логическому разъёму на концентраторе данных другого блока. На приведенной ниже схеме показано, данные какого типа может принимать или передавать каждый из разъёмов.

## **Цвет шин данных**

Шины данных окрашены в различные цвета: шины для передачи числовых данных - желтые; шины, для передачи логических данных - зеленые; шины для передачи текстовых данных - оранжевые.

## **«Повреждённые» шины данных**

При попытке подключить шину данных к разъёму данных несоответствующего типа, шина будет повреждена (ее цвет изменится на серый). Вы не сможете загрузить программу, если шина данных повреждена.

Если щёлкнуть на повреждённой шине, то в небольшом окне контекстной справки в правом нижнем углу рабочей области появится информация о причине повреждения.

## **Данные должны соответствовать возможному диапазону, в котором работает разъём**

Если шина данных входа передает значение, не попадающее в допустимый диапазон разъёма, к которому она подключена, блок либо проигнорирует такие данные, либо приведёт их к значению, попадающему в этот диапазон. Если разъём допускает ограниченное количество значений (например, только 0, 1 или 2), он проигнорирует входящий сигнал, если передаваемое значение не будет вписываться в допустимый диапазон.

Если разъём допускает более широкий диапазон значений (например, 0 – 100), любой входящий сигнал, не попадающий в допустимый диапазон, будет приведен к значению, соответствующее ему. Например, если разъём мощности блока «Движение» получает входящий сигнал со значением 150, блок преобразует входящее значение в 100 (т.е., в числовое значение, лежащее в пределах диапазона разъёма мощности ).

# Блок преобразования числовых данных в текстовые

Этот блок регистрирует числовое значение (подобно тому, как данные регистрируются при помощи датчика) и преобразует это значение в текстовый формат, который может быть отображен на экране NXT. Числовое значение может быть введено путем набора или динамически, при помощи шины данных.

Текстом является группа символов любой длины, набираемая Вами при помощи клавиатуры. Все это - примеры текста: abcd23, собака, Джон!. Текст важен, поскольку он может быть отображен на экране NXT.

Числа, сгенерированные NXT при запуске вашей программы, не являются текстом, они не могут быть отображены непосредственно на экране NXT. Вы должны в начале преобразовать их в текст при помощи блока преобразования числовых данных в текстовые.

При преобразовании числовых данных в текстовые, Вы можете добавить их в предложение, которое будет отображено на экране NXT. При использовании другого блока, называемого Текстовым блоком, вы можете составлять такие предложения, как: “Мотор сделал 6 оборотов” или “Показания датчика освещенности 35%”, которые могут отображаться на экране NXT. В данных предложениях числа 6 и 35 были сгенерированы NXT (а не введены) и перед тем как добавиться к другим элементам текста должны были быть преобразованы блоком преобразования числовых данных в текстовые. (См. разделу "Текстовый блок".)

Числовое значение может быть введено в блок преобразования числовых данных в текстовые путем набора или динамически, при помощи шины данных. Выводимый текст посылается только через шину данных. (Чтобы получить дополнительную информацию, см. ниже раздел "Концентратор данных".)



1. Концентратор данных откроется автоматически, когда блок будет присоединен к коммутатору последовательности действий. Числовое значение может вводиться динамически при помощи шины данных через входной разъем на левой стороне концентратора данных. Выводимый текст будет посылаться через выходной разъем справа внизу.

При необходимости числовой выходной разъем на правой стороне концентратора данных может использоваться для передачи любых входных числовых данных на другие блоки.

## Конфигурация блока преобразования числовых данных в текстовые



1. Числовое значение может быть введено путем набора или динамически, при помощи шины данных.

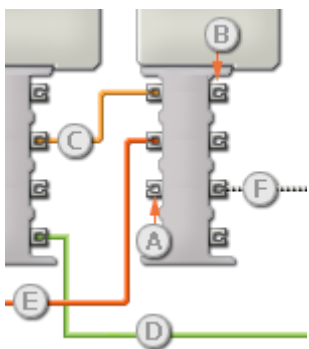
## Конфигурация концентратора данных блока преобразования числовых данных в текстовые

Вы можете управлять блоком преобразования числовых данных в текстовые динамически, подключая шины данных (от концентраторов данных других блоков) к концентратору данных блока преобразования числовых данных в текстовые.

Поместив блок в рабочую область, откройте концентратор данных блока, щелкнув по кнопке в нижней левой части блока.



Шины данных, передающие входящие данные на блок, подключены к разъемам, находящимся на левой стороне их концентраторов данных. Шины данных, передающие исходящие данные, подключены к разъемам, находящимся на правой стороне.



- [A] Входной разъем
- [B] Выходной разъем
- [C] Числовая шина данных (желтая)
- [D] Логическая шина данных (зеленая)
- [E] Текстовая шина данных (оранжевая)
- [F] Поврежденная шина данных (серая)

### Передача данных от входного разъема на выходной разъем

Если для входного разъема есть соответствующий выходной разъем (см. выше A), входящие данные будут проходить от входного разъема на выходной разъем без изменений. В этом случае вы сможете использовать выходной разъем, если входной разъем подключен к шине данных входа; подключение шины данных выхода к такому выходному разъему без подключенной шины данных входа приведет к «неисправности» шины данных выхода (она будет окрашена в серый цвет).

## **Шины данных осуществляют передачу специфических типов данных**

Каждая шина данных осуществляет обмен специфическими типами данных между блоками. Например, если шина данных перетянута с логического разъема на концентратор данным блока, ее можно подключить только к логическому разъему на концентраторе данных другого блока. На приведенной ниже схеме показано, какой вид данных может принимать или передавать каждый из разъемов.

### **Окраски шин данных**

Шины данных различаются особыми окрасками: шины, передающие числовые данные, имеют желтый цвет; шины, передающие логические данные, имеют зеленый цвет; шины, передающие текстовые данные, имеют оранжевый цвет.

### **«Поврежденные» шины данных**

Если вы пытаетесь подключить шину данных к разъему несоответствующего типа данных, шина будет повреждена (ее цвет изменится на серый). Вы не сможете загрузить вашу программу, если шина данных повреждена.

Если вы щелкнете по поврежденной шине, то в небольшом окне контекстной справки в правом нижнем углу рабочей области вы сможете прочесть информацию о причине повреждения.

### **Данные должны соответствовать возможному диапазону, в котором работает разъем**

Если шина данных входа передает значение, не попадающее в допустимый диапазон разъема, к которому она подключена, блок либо проигнорирует такие данные, либо трансформирует данные, попадающие в этот диапазон. Для разъемов, допускающих небольшое количество значений (например, только 0, 1 или 2), разъем проигнорирует входящий сигнал, если передаваемое значение не будет вписываться в диапазон разъема.

Для разъемов, допускающих более широкий диапазон значений (например, 0 – 100), разъем усилит любой входящий сигнал, если он не соответствует диапазону. Например, если разъем питания программного блока перемещения получает входящий сигнал со значением 150, блок преобразует входящее значение в 100 (т.е., в числовое значение, лежащее в пределах диапазона разъема мощности).

## **Программный блок сброса электродвигателя**

Интерактивные серводвигатели обладают механизмом автоматического исправления ошибок, что позволяет Вашему роботу передвигаться очень точно. Однако, иногда вы можете захотеть отключить эту функцию - программный блок сброса электродвигателя позволит вам это сделать.



1. Буква в верхнем правом углу блока показывает, на какого из портов NXT будет влиять программный блок сброса электродвигателя.

## Конфигурация программного блока сброса электродвигателя



1. Выберите, на какого из портов NXT будет влиять программный блок сброса электродвигателя.

## Как использовать программный блок сброса электродвигателя

Если вы используете один программный блок перемещения (с параметром Следующего действия, установленного на Торможение), чтобы привести вашего робота в движение на 8 оборотов вперед, перед остановкой он осуществит ровно 8 оборотов.

Вы можете сделать то же самое с 4-мя программными блоками перемещения, с каждым, установленным на 2 оборота. При помощи параметра Следующего действия установите катание на первые 3 блока, механизм автоматического исправления ошибок обеспечит продвижение робота ровно на 8 оборотов.

Предполагая, что движение будет осуществляться по ровной поверхности или даже по поверхности с небольшим наклоном, фактическое расстояние, пройденное за время работы каждого блока, в данном примере может быть следующим:

Блок	Заданная длительность	Фактическая длительность
Передвижение №1, катиться	2 вращений	2,2 вращений
Ожидание, 1 секунда		
Передвижение №2, катиться	2 вращений	2,3 вращений
Ожидание, 1 секунда		
Передвижение №3, катиться	2 вращений	2,1 вращений



Ожидание, 1  
секунда

Передвижение  
№4,  
торможение

2 вращений

1,5 вращений

ИТОГО

8,0 вращений



Примечание: Если вы устанавливаете параметр Следующего действия как Торможение для всех программных блоков перемещения, ваш робот будет передвигаться более точно в соответствии с каждым шагом. Ошибки возникают только в случае, когда для Следующего действия установлен параметр катиться.

Если вы хотите, чтобы ваш робот переместился на 2 оборота для каждого программного блока перемещения, и при этом хотите позволить колесам свободно вращаться катиться без воздействия на следующее движение, вам придется задействовать программный блок сброса электродвигателя. Это действие отключит автоматическое исправление ошибок.

При использовании программных блоков сброса электродвигателя пример, приведенный выше, будет изменен, как описано ниже:

Блок	Заданная длительность	Фактическая длительность
Передвижение №1, катиться	2 вращений	2,2 вращений
Ожидание, 1 секунда		
Обнулить Мотор		
Передвижение №2, катиться	2 вращений	2,3 вращений
Ожидание, 1 секунда		
Обнулить Мотор		
Передвижение №3, катиться	2 вращений	2,1 вращений
Ожидание, 1 секунда		
Обнулить Мотор		
Передвижение №4, торможение	2 вращений	2,0 вращений



ИТОГО

8,5 вращений

Примечание: Длительность ошибки будет разной в зависимости от трения и/или угла наклона.



Совет: Вы также можете сбросить движение электродвигателя и программного блока перемещения, установив значение Стоп для параметра Направление.

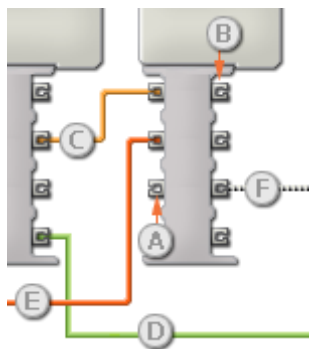
## Конфигурация концентратора данных программного блока сброса электродвигателя

Вы можете управлять программным блоком сброса электродвигателя динамически, подключая шины данных (от концентраторов данных других блоков) к концентратору данных программного блока сброса электродвигателя.

Поместив блок в рабочую область, откройте концентратор данных блока, щелкнув по кнопке в нижней левой части блока.



Шины данных, передающие входящие данные на блок, подключены к разъемам, находящимся на левой стороне их концентраторов данных. Шины данных, передающие исходящие данные, подключены к разъемам, находящимся на правой стороне.



- [A] Входной разъем
- [B] Выходной разъем
- [C] Числовая шина данных (желтая)
- [D] Логическая шина данных (зеленая)
- [E] Текстовая шина данных (оранжевая)
- [F] Поврежденная шина данных (серая)

### Передача данных от входного разъема на выходной разъем

Если для входного разъема есть соответствующий выходной разъем (см. выше A), входящие данные будут проходить от входного разъема на выходной разъем без изменений. В этом случае вы сможете использовать выходной разъем, если входной разъем подключен к шине данных входа; подключение шины данных выхода к такому выходному разъему без подключенной шины данных входа приведет к «неисправности» шины данных выхода (она будет окрашена в серый цвет).

### Шины данных осуществляют передачу специфических типов данных

Каждая шина данных осуществляет обмен специфическими типами данных между блоками. Например, если шина данных перетянута с логического разъема на концентратор данным блока, ее можно подключить только к логическому разъему на концентраторе данных другого блока. На приведенной ниже схеме показано, какой вид данных может принимать или передавать каждый из разъемов.

## Окраски шин данных

Шины данных различаются особыми окрасками: шины, передающие числовые данные, имеют желтый цвет; шины, передающие логические данные, имеют зеленый цвет; шины, передающие текстовые данные, имеют оранжевый цвет.

## «Поврежденные» шины данных

Если вы пытаетесь подключить шину данных к разъему несоответствующего типа данных, шина будет повреждена (ее цвет изменится на серый). Вы не сможете загрузить вашу программу, если шина данных повреждена.

Если вы щелкнете по поврежденной шине, то в небольшом окне контекстной справки в правом нижнем углу рабочей области вы сможете прочесть информацию о причине повреждения.

## Данные должны соответствовать возможному диапазону, в котором работает разъем

Если шина данных входа передает значение, не попадающее в допустимый диапазон разъема, к которому она подключена, блок либо проигнорирует такие данные, либо трансформирует данные, попадающие в этот диапазон. Для разъемов, допускающих небольшое количество значений (например, только 0, 1 или 2), разъем проигнорирует входящий сигнал, если передаваемое значение не будет вписываться в диапазон разъема.

Для разъемов, допускающих более широкий диапазон значений (например, 0 – 100), разъем усилит любой входящий сигнал, если он не соответствует диапазону. Например, если разъем питания программного блока перемещения получает входящий сигнал со значением 150, блок преобразует входящее значение в 100 (т.е., в числовое значение, лежащее в пределах диапазона разъема мощности).

## Программный текстовый блок

Этот блок может объединять символы в группы, образуя текст. Ниже приведены примеры текста: abcd23, Dog, john!. Текст может содержать цифры и даже специальные знаки, такие, как символы и знаки препинания. Текст важен, поскольку он может быть отображен на экране NXT.

Предложение, показанное здесь, является длинным фрагментом текста. Отдельные слова являются более мелкими элементами текста. Программный текстовый блок может объединить три более мелких фрагмента текста и сформировать более крупный фрагмент, похожий на предложение.

Если сигнал на входе А является элементом текста "Эта", а сигналы В и С являются текстовыми элементами " программа - " и "чудо!", полученный объединенный текст будет выглядеть так: "Эта программа - чудо!". Учтите, что текстовый элемент В (" программа ") содержит два пробела, что имеет немаловажное значение при формировании правильного предложения.

Числа, которые генерирует устройство NXT не являются обычными текстовыми элементами; их нельзя отобразить непосредственно на экране устройства NXT. Если вы используете другой программный блок, называемый "Блок преобразования числовых данных в текстовые", вы сможете конвертировать числовые значения в текст. Это может позволить вам использовать Программный текстовый блок для создания более крупных фрагментов текста, включая числовые значения, считанные с датчиков:

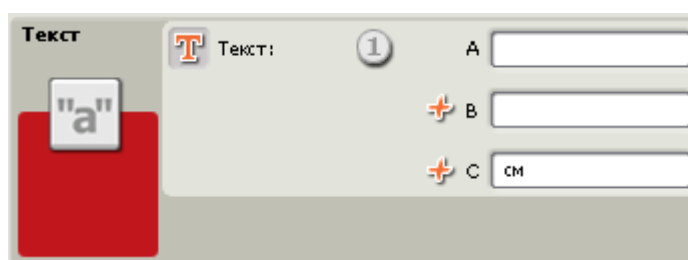
“Электродвигатель выполнил 6 оборотов” или “Показания датчика освещенности 35%.” Оба приведенный в качестве примера текстовых фрагмента можно отобразить на экране устройства NXT.

Три вводимых текстовых фрагмента могут быть введены путем набора или динамически, при помощи шин данных. Выводимый текст посылается через шину данных.



1. Концентратор данных откроется автоматически, когда блок будет присоединен к коммутатору последовательности действий. Три вводимых текстовых фрагмента могут быть загружены динамически при помощи шин данных, подключенных к разъемам на левой стороне концентратора данных. Выводимый текст будет направлен от разъема правой верхней части. Три других выходных разъема справа будут передавать входящую текстовую информацию к другим программным блокам, если это необходимо.

## Настройка Программного текстового блока



1. Три вводимых текстовых фрагмента (A, B, & C) могут быть введены путем набора или динамически, при помощи шин данных. Не забывайте, что пробелы имеют большое значение и будут применены в выходящем тексте только в том случае, если они были во входящем тексте.

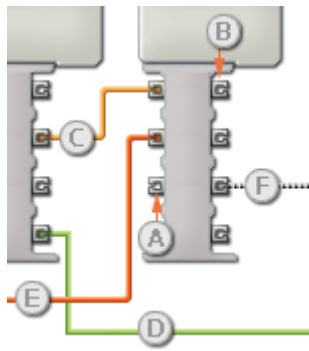
## Настройка концентратора данных Программного текстового блока

Вы можете управлять Программным текстовым блоком динамически, подключая шины данных (от концентраторов данных других блоков) к концентратору данных Программного текстового блока.

Поместив блок в рабочую область, откройте концентратор данных блока, щелкнув по кнопке в нижней левой части блока.



Шины данных, передающие входящие данные на блок, подключены к разъемам, находящимся на левой стороне их концентраторов данных. Шины данных, передающие исходящие данные, подключены к разъемам, находящимся на правой стороне.



- [A] Входной разъем
- [B] Выходной разъем
- [C] Числовая шина данных (желтая)
- [D] Логическая шина данных (зеленая)
- [E] Текстовая шина данных (оранжевая)
- [F] Поврежденная шина данных (серая)

## Передача данных от входного разъема на выходной разъем

Если для входного разъема есть соответствующий выходной разъем (см. выше A), входящие данные будут проходить от входного разъема на выходной разъем без изменений. В этом случае вы сможете использовать выходной разъем, если входной разъем подключен к шине данных входа; подключение шины данных выхода к такому выходному разъему без подключенной шины данных входа приведет к «неисправности» шины данных выхода (она будет окрашена в серый цвет).

## Шины данных осуществляют передачу специфических типов данных

Каждая шина данных осуществляет обмен специфическими типами данных между блоками. Например, если шина данных перетянута с логического разъема на концентратор данных блока, ее можно подключить только к логическому разъему на концентраторе данных другого блока. На приведенной ниже схеме показано, какой вид данных может принимать или передавать каждый из разъемов.

## Окраски шин данных

Шины данных различаются особыми окрасками: шины, передающие числовые данные, имеют желтый цвет; шины, передающие логические данные, имеют зеленый цвет; шины, передающие текстовые данные, имеют оранжевый цвет.

## «Поврежденные» шины данных

Если вы пытаетесь подключить шину данных к разъему несоответствующего типа данных, шина будет повреждена (ее цвет изменится на серый). Вы не сможете загрузить вашу программу, если шина данных повреждена.

Если вы щелкнете по поврежденной шине, то в небольшом окне контекстной справки в правом нижнем углу рабочей области вы сможете прочесть информацию о причине повреждения.

## Данные должны соответствовать возможному диапазону, в котором работает разъем

Если шина данных входа передает значение, не попадающее в допустимый диапазон разъема, к которому она подключена, блок либо проигнорирует такие данные, либо трансформирует данные, попадающие в этот диапазон. Для разъемов, допускающих

небольшое количество значений (например, только 0, 1 или 2), разъем проигнорирует входящий сигнал, если передаваемое значение не будет вписываться в диапазон разъема.

Для разъемов, допускающих более широкий диапазон значений (например, 0 – 100), разъем усилит любой входящий сигнал, если он не соответствует диапазону. Например, если разъем питания программного блока перемещения получает входящий сигнал со значением 150, блок преобразует входящее значение в 100 (т.е., в числовое значение, лежащее в пределах диапазона разъема мощности).

## Блок «Bluetooth соединение»

Блок соединения через Bluetooth используется для соединения с другим устройством Bluetooth, например с другим устройством NXT или компьютером, с которым соединение уже устанавливалось (выполнялось "рукопожатие"), и которое уже имеется в списке контактов.

Более подробная информация по настройке устройств NXT для работы с беспроводной связью приведена в разделе [Блок «Отправить сообщение»](#).

Кроме того, Блок «Bluetooth соединение» позволяет включать и выключать функцию Bluetooth.

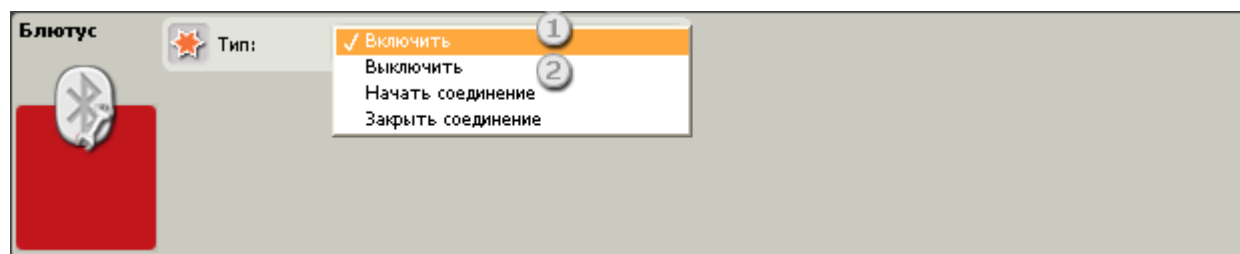
### Содержание

- [Включение/выключение](#)
- [Установка соединения](#)
- [Закрытие соединения](#)
- [Настройка концентратора данных Блока «Bluetooth соединение»](#)



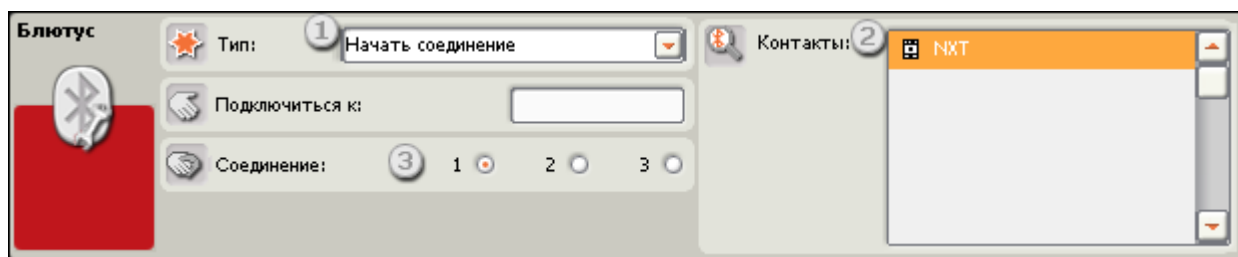
1. Эта пиктограмма показывает, на какой тип операции настроен Блок: Включить, Выключить, Установить соединение или Закрыть соединение.

### Включение/Выключение



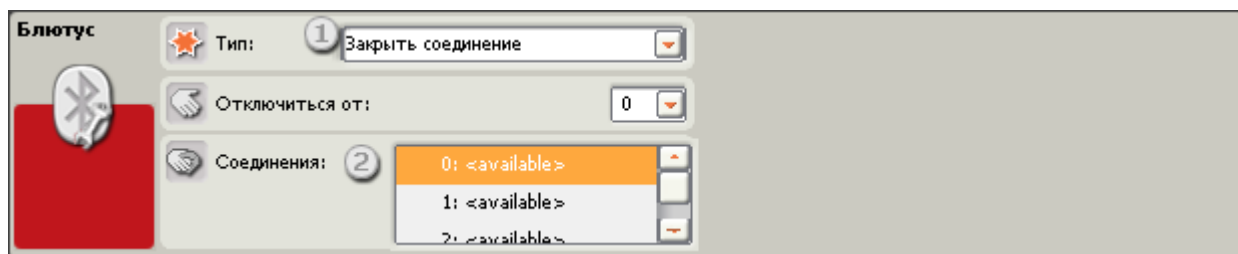
1. Для того чтобы включить функцию Bluetooth, выберите пункт меню Включить
2. Для того чтобы выключить функцию Bluetooth, выберите пункт меню Выключить

### Установка соединения



1. Для того чтобы установить соединение с другим устройством Bluetooth, выберите пункт меню Начать соединение.
2. Если устройство NXT подключено к компьютеру и включено, список контактов автоматически заполняется контактами, полученными от этого устройства. Если имя в списке контактов отсутствует, его можно ввести с клавиатуры в соответствующее поле ввода.
3. Выберите номер соединения от 1 до 3 (Канал 0 зарезервирован для устройства управления элементами NXT). Соединение по каналу 0 должно устанавливаться другим устройством, например, компьютером).

## Заккрытие соединения



1. Для того чтобы завершить соединение, выберите пункт меню Закрыть соединение.
2. Выберите номер соединения, которое следует завершить, в ниспадающем меню или выберите имя устройства Bluetooth в списке соединений.

Примечание: Если через канал 0 устройство NXT подключено к компьютеру, и это соединение будет закрыто, для продолжения работы с Bluetooth в целях загрузки программ будет необходимо повторно установить соединения при помощи окна NXT.

## Настройка концентратора данных

Возможно динамическое управление Блоком соединения через Bluetooth путем подключения шин данных (от концентраторов данных других Блоков) к концентратору данных данного Блока.

После перемещения Блока в рабочую область, откройте его концентратор данных, нажав на кнопку, расположенную в нижней левой части Блока.



Шины данных, передающие входящие данные на Блок, подключены к разъемам, находящимся на левой стороне их концентраторов данных. Шины данных, передающие исходящие данные, подключены к разъемам, находящимся на правой стороне.

## Передача данных от входного разъема на выходной разъем

Если для входного разъема есть соответствующий выходной разъем (см. выше А), входящие данные будут проходить от входного разъема на выходной разъем без изменений. В этом случае вы сможете использовать выходной разъем, если входной разъем подключен к шине данных входа; подключение шины данных выхода к такому выходному разъему без подключенной шины данных входа приведет к "неисправности" шины данных выхода (она будет окрашена в серый цвет).

## **Шины данных осуществляют передачу специфических типов данных**

Каждая шина данных осуществляет обмен специфическими типами данных между Блоками. Например, если шина данных перетянута с логического разъема на концентратор данных Блока, ее можно подключить только к логическому разъему на концентраторе данных другого Блока. На приведенной ниже схеме показано, какой вид данных может принимать или передавать каждый из разъемов.

## **Окраски шин данных**

Шины данных различаются особыми окрасками: шины, передающие числовые данные, имеют желтый цвет; шины, передающие логические данные, имеют зеленый цвет; шины, передающие текстовые данные, имеют оранжевый цвет.

## **Поврежденные шины данных**

Если вы пытаетесь подключить шину данных к разъему несоответствующего типа данных, шина будет повреждена (ее цвет изменится на серый). Вы не сможете загрузить вашу программу, если шина данных повреждена.

Если вы щелкнете по поврежденной шине, то в небольшом окне контекстной справки в правом нижнем углу рабочей области вы сможете прочесть информацию о причине повреждения.

## **Данные должны соответствовать возможному диапазону, в котором работает разъем**

Если шина данных входа передает значение, не попадающее в допустимый диапазон разъема, к которому она подключена, Блок либо проигнорирует такие данные, либо трансформирует данные, попадающие в этот диапазон. Для разъемов, допускающих небольшое количество значений (например, только 0, 1 или 2), разъем проигнорирует входящий сигнал, если передаваемое значение не будет вписываться в диапазон разъема.

Для разъемов, допускающих более широкий диапазон значений (например, 0 – 100), разъем усилит любой входящий сигнал, если он не соответствует диапазону. Например, если разъем питания Блока перемещения получает входящий сигнал со значением 150, Блок преобразует входящее значение в 100 (т.е., в числовое значение, лежащее в пределах диапазона разъема мощности).

# **Блок «Начать регистрацию данных»**

Этот блок можно использовать для перевода NXT в режим сбора данных от датчиков и сохранения их в файле своей памяти.



## Содержание

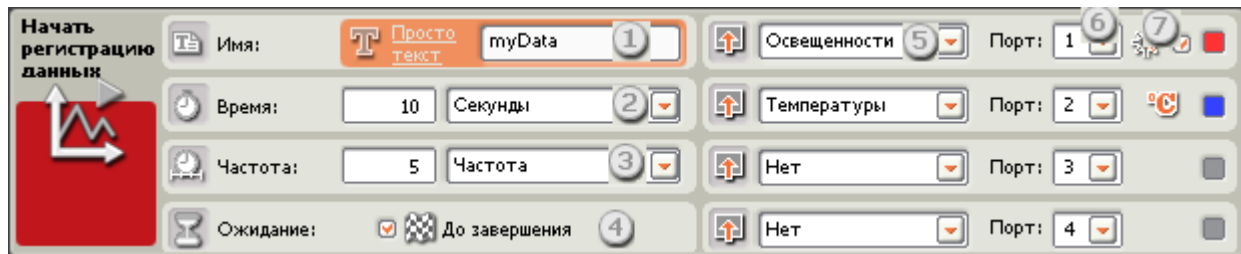
- [Настройки отображения](#)
- [Настройка Блока «Начать регистрацию данных»](#)
- [Файлы экспериментов на NXT](#)
- [Настройка концентратора данных Блока «Начать регистрацию данных»](#)

## Отображение параметров настройки



1. Имя файла экспериментов, в который будут записываться данные.
2. Эта пиктограмма показывает выбранную продолжительность эксперимента (секунды, минуты, Без ограничения или одно измерение). Пиктограммы для секунд и минут одинаковы.

## Настройка Блока «Начать регистрацию данных»



1. Введите имя файла эксперимента в поле ввода. По умолчанию используется имя myData
2. Выберите продолжительность эксперимента в секундах или минутах и укажите это значение в соответствующем поле ввода слева. Если эксперимент должен выполняться непрерывно до тех пор, пока не будет остановлен соответствующим программным блоком, выберите Ограничения нет. Если в ходе эксперимента данные должны быть получены только один раз, выберите Единичное измерение.
3. Выберите частоту, с которой будут производиться измерения. Это может быть количество измерений в секунду, или интервал времени между замерах. Для этого введите необходимое значение в соответствующее поле слева.
4. Если установлен флажок «До завершения» (по умолчанию он установлен), то выполнение программы продолжится только после того, как Блок «Начать регистрацию данных» полностью прекратит работу. Если снять флажок «До завершения», то другие блоки в программе могут выполняться, пока Блок «Начать регистрацию данных» завершает работу. Например, Блок «Звук», установленный после Блока «Начать регистрацию данных», может начать воспроизводить звуковой файл, пока Блок «Начать регистрацию данных» продолжает работать.
5. В открывающемся списке «Датчик» выберите датчик, который будет регистрировать данные.
6. В открывающемся списке «Порт» выберите порт NXT, к которому подключен датчик.
7. Щёлкните на нужной пиктограмме, чтобы выбрать для каждого датчика единицу измерения.

## Файлы экспериментов NXT

Блок «Начать регистрацию данных» создает в NXT файл эксперимента. Чтобы получить доступ к этому файлу, откройте окно NXT, выберите в списке слева категорию файлов «Прочие», после чего укажите файл эксперимента, который необходимо передать, и щёлкните «Передать». Затем выберите папку, куда будет скопирован этот файл эксперимента на компьютере.

После того как файлы экспериментов будут скопированы на компьютер, их можно открыть в текстовом редакторе или электронной таблице.

Для создания нескольких файлов экспериментов можно использовать несколько Блоков «Начать регистрацию данных». Название блока определяет, какой файл NXT он будет использовать. Если одно и то же имя используется несколькими Блоками «Начать регистрацию данных», данные будут записываться в один и тот же файл эксперимента. Если Блоки «Начать регистрацию данных» имеют различные имена, то для регистрации данных будут созданы разные файлы экспериментов.

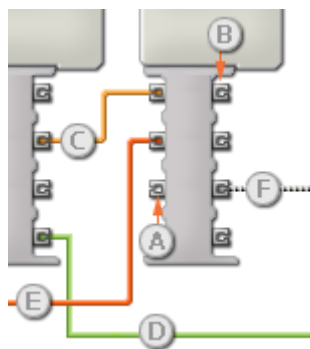
## Настройка концентратора данных Блока «Начать регистрацию данных»

Блоком «Начать регистрацию данных» можно управлять динамически, подключая шины данных (от концентраторов данных других блоков) к его концентратору данных.

Перетащите Блок «Начать регистрацию данных» в рабочую область и откройте его концентратор данных, щёлкнув на кнопке в нижней левой части блока.



Шины данных, по которым данные поступают на блок, подключены к разъёмам, расположенным на левой стороне их концентраторов данных. Шины данных, передающие исходящие данные, подключены к разъёмам, расположенным справа.



- [A] Входной разъём
- [B] Выходной разъём
- [C] Числовая шина данных (желтая)
- [D] Логическая шина данных (зеленая)
- [E] Текстовая шина данных (оранжевая)
- [F] Повреждённая шина данных (серая)

### Передача данных от входного разъёма на выходной разъём

Если для входного разъёма имеется соответствующий выходной разъём (см. выше A), входящие данные будут проходить от входного разъёма на выходной разъём без изменений. В этом случае можно использовать только выходной разъём, если входной разъём подключен к шине данных входа; подключение шины данных выхода к такому выходному разъёму без подключенной шины данных входа приведет к «повреждению» шины данных выхода (она станет серой).

### По шинам данных передаются данные различных типов

Через каждую шину данных осуществляется обмен данными различных типов между блоками. Например, если шина данных протянута с логического разъёма на концентратор данных блока, её можно подключить только к логическому разъёму на концентраторе данных другого блока. На приведенной ниже схеме показано, данные какого типа может принимать или передавать каждый из разъёмов.

## Цвет шин данных

Шины данных окрашены в различные цвета: шины для передачи числовых данных - желтые; шины, для передачи логических данных - зеленые; шины для передачи текстовых данных - оранжевые.

## Повреждённые шины данных

При попытке подключить шину данных к разъёму данных несоответствующего типа, шина будет повреждена (ее цвет изменится на серый). Вы не сможете загрузить программу, если шина данных повреждена.

Если щёлкнуть на повреждённой шине, то в небольшом окне контекстной справки в правом нижнем углу рабочей области появится информация о причине повреждения.

## Данные должны соответствовать возможному диапазону, в котором работает разъём

Если шина данных входа передает значение, не попадающее в допустимый диапазон разъёма, к которому она подключена, блок либо проигнорирует такие данные, либо приведёт их к значению, попадающему в этот диапазон. Если разъём допускает ограниченное количество значений (например, только 0, 1 или 2), он проигнорирует входящий сигнал, если передаваемое значение не будет вписываться в допустимый диапазон.

Если разъём допускает более широкий диапазон значений (например, 0 – 100), любой входящий сигнал, не попадающий в допустимый диапазон, будет приведен к значению, соответствующее ему. Например, если разъём мощности блока «Движение» получает входящий сигнал со значением 150, блок преобразует входящее значение в 100 (т.е., в числовое значение, лежащее в пределах диапазона разъёма мощности).

# Блок «Остановить регистрацию данных»

Этот блок останавливает работу Блока «Начать регистрацию данных», настроенного на неограниченный по времени сбор данных («Без ограничения»).

## Содержание

- [Настройки отображения](#)
- [Настройка Блока «Остановить регистрацию данных»](#)
- [Настройка концентратора данных Блока «Остановить регистрацию данных»](#)

## Настройки отображения



1. Показано имя Блока «Начать регистрацию данных», работа которого должна быть остановлена.

## Настройка Блока «Остановить регистрацию данных»



1. В открывающемся списке выберите имя Блока «Начать регистрацию данных», который нужно остановить.

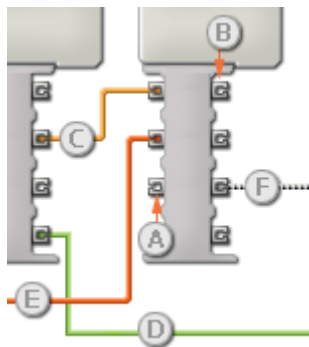
## Настройка концентратора данных Блока «Остановить регистрацию данных»

Блоком «Остановить регистрацию данных» можно управлять динамически, подключая шины данных (от концентраторов данных других блоков) к его концентратору данных.

Перетащите Блок «Остановить регистрацию данных» в рабочую область и откройте его концентратор данных, щёлкнув на кнопке в нижней левой части блока.



Шины данных, по которым данные поступают на блок, подключены к разъёмам, расположенным на левой стороне их концентраторов данных. Шины данных, передающие исходящие данные, подключены к разъёмам, расположенным справа.



- [A] Входной разъём
- [B] Выходной разъём
- [C] Числовая шина данных (желтая)
- [D] Логическая шина данных (зеленая)
- [E] Текстовая шина данных (оранжевая)
- [F] Повреждённая шина данных (серая)

### Передача данных от входного разъёма на выходной разъём

Если для входного разъёма имеется соответствующий выходной разъём (см. выше А), входящие данные будут проходить от входного разъёма на выходной разъём без изменений. В этом случае можно использовать только выходной разъём, если входной

разъём подключен к шине данных входа; подключение шины данных выхода к такому выходному разъёму без подключенной шины данных входа приведет к «повреждению» шины данных выхода (она станет серой).

## **По шинам данных передаются данные различных типов**

Через каждую шину данных осуществляется обмен данными различных типов между блоками. Например, если шина данных протянута с логического разъёма на концентратор данных блока, её можно подключить только к логическому разъёму на концентраторе данных другого блока. На приведенной ниже схеме показано, данные какого типа может принимать или передавать каждый из разъёмов.

## **Цвет шин данных**

Шины данных окрашены в различные цвета: шины для передачи числовых данных - желтые; шины, для передачи логических данных - зеленые; шины для передачи текстовых данных - оранжевые.

## **Повреждённые шины данных**

При попытке подключить шину данных к разъёму данных несоответствующего типа, шина будет повреждена (ее цвет изменится на серый). Вы не сможете загрузить программу, если шина данных повреждена.

Если щёлкнуть на повреждённой шине, то в небольшом окне контекстной справки в правом нижнем углу рабочей области появится информация о причине повреждения.

## **Данные должны соответствовать возможному диапазону, в котором работает разъём**

Если шина данных входа передает значение, не попадающее в допустимый диапазон разъёма, к которому она подключена, блок либо проигнорирует такие данные, либо приведёт их к значению, попадающему в этот диапазон. Если разъём допускает ограниченное количество значений (например, только 0, 1 или 2), он проигнорирует входящий сигнал, если передаваемое значение не будет вписываться в допустимый диапазон.

Если разъём допускает более широкий диапазон значений (например, 0 – 100), любой входящий сигнал, не попадающий в допустимый диапазон, будет приведен к значению, соответствующее ему. Например, если разъём мощности блока «Движение» получает входящий сигнал со значением 150, блок преобразует входящее значение в 100 (т.е., в числовое значение, лежащее в пределах диапазона разъёма мощности).

## **На этой схеме показаны различные характеристики разъёмов на концентраторе данных Блока «Остановить регистрацию данных»:**



## Пользовательские Блоки

- [«Мой Блок»](#)

### «Мои Блоки»

Конструктор блоков позволяет выделить несколько блоков в рабочей области и сгруппировать их в отдельный Блок со своей пиктограммой. Например, в Блок под названием «Движение робота» могут входить Блок «Движение» (для управления колесами робота) и Блок «Звук» (воспроизводящий звуковой файл, когда робот перемещается на определенное расстояние).

При создании новой программы для того же робота достаточно перетащить блок «Движение робота» из Пользовательской палитры: эта часть программы будет готова, поскольку все параметры уже заданы в Блоке.

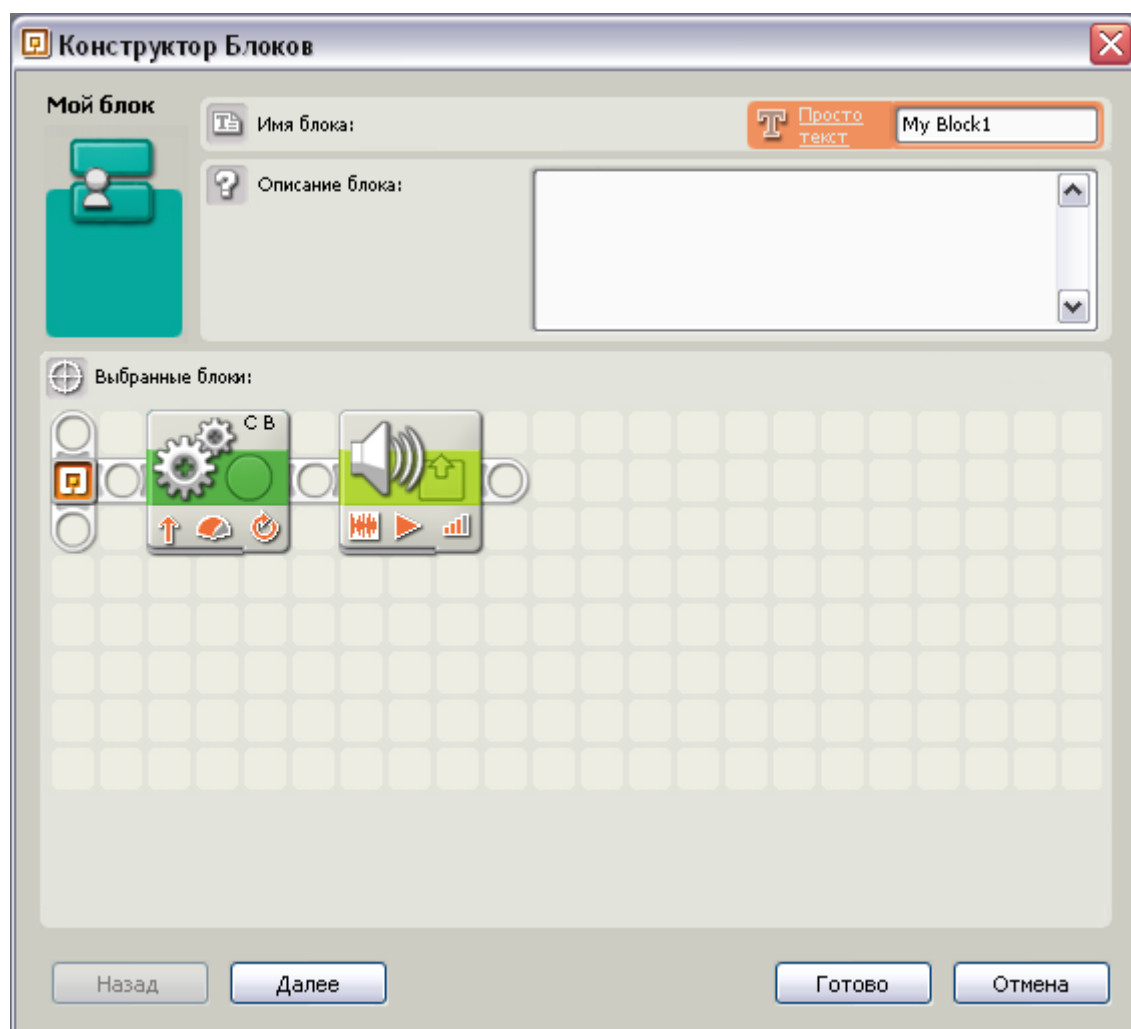
Со временем вы можете создать библиотеку Блоков, которые можно использовать в других программах или обмениваться с другими пользователями MINDSTORMS.

### Создание Блока

1. Сначала выберите блоки, которые можно последовательно (логически) сгруппировать. В предыдущем примере показано, что в Блок можно объединить блоки, управляющие движением робота.
2. Выделите блоки, которые нужно объединить в Блок: установите указатель мыши в угол воображаемого прямоугольника, который должен охватить все эти блоки; удерживая кнопку мыши, переместите указатель к противоположному углу воображаемого прямоугольника. Отпустите кнопку мыши: все блоки нового Блока будут выбраны. При этом нужно учитывать, что если даже часть какого-либо блока попадёт в прямоугольник, этот блок будет включён в Блок.

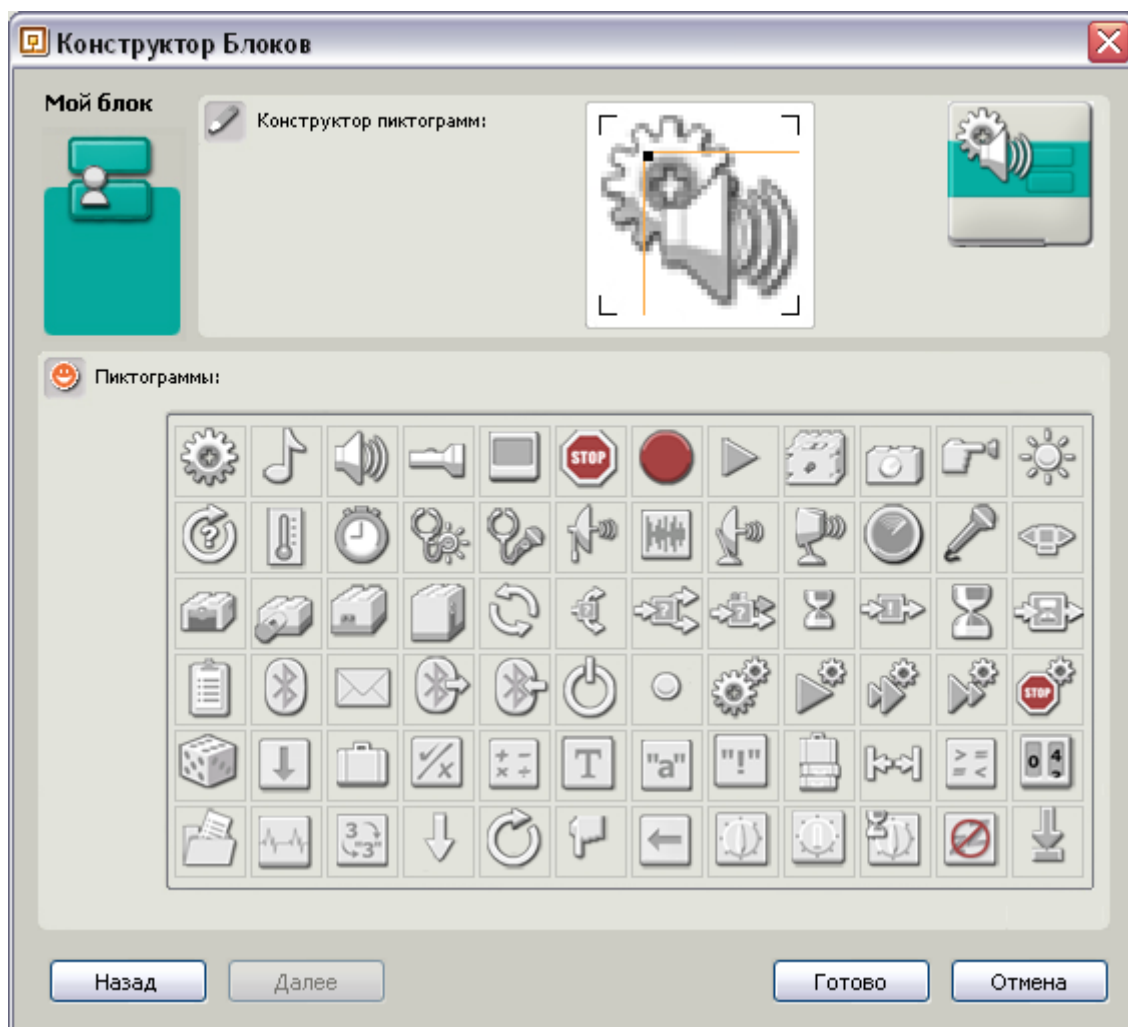


3. Выбрав блоки, откройте в верхней части окна NXT меню «Правка» и выберите команду «Создать Блок». Откроется первый экран «Конструктор Блоков» (показан ниже).

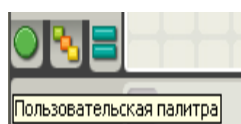


4. Присвойте новому Блоку имя и кратко опишите его функции. Пример. «этот блок управляет моторами TriBot и выдаёт звуковой сигнал, когда робот перемещается на один метр». Щёлкните «Далее» для перехода к следующему шагу.





5. Теперь для нового Блока нужно создать значок. Мышью перетащите один-два значка в поле пиктограммы нового блока. Клавишами навигации на клавиатуре установите пиктограммы точно в нужное место. Справа показано, как выглядит в данный момент пиктограмма вашего блока. Щёлкните «Готово». Новый Блок появится в текущей программе и Моей палитре (чтобы открыть её щёлкните на крайней правой кнопке в нижней части окна палитры). (См. рисунок ниже.)



Примечание: учитывайте, что если любые входные или выходные шины данных пересекают рамку выделения (охватывающую блоки, объединённые в Мой Блок), то будут созданы входные и выходные разъёмы концентратора данных нового Блока.

Чтобы изменить конфигурацию доступных разъёмов Блока, нужно снова создать Блок, но уже с корректным расположением разъёмов и соответствующими шинами данных, выходящими из выбранной области.

## РЕДАКТИРОВАНИЕ УЖЕ СОЗДАННОГО БЛОКА

Чтобы изменить функции Блока, отредактируйте содержащиеся в нем блоки, дважды щёлкнув на Блоке, или выбрав в меню «Правка» команду «Редактировать Мой блок». Если требуется изменить только пиктограмму элемента Мой Блок, выберите блок и выберите в меню «Правка» команду «Изменить значок Блока» — при этом запустится «Конструктор блоков», в котором можно внести изменения.

## УПРАВЛЕНИЕ ПОЛЬЗОВАТЕЛЬСКОЙ ПАЛИТРОЙ

Для добавления и удаления Блоков в Пользовательской палитре, в меню «Правка» выберите команду «Управление Палитрами». Чтобы отправить другу какой-либо «Блок», создайте его копию и вложите её в электронное письмо. Если вы получите Блок от друга, то зайдите в «Управление Палитрами» и перетащите этот «Мой Блок» в папку «Мои Блоки».

В Пользовательской палитре можно создавать новые дополнительные палитры. Для этого в папке «Блок» нужно создать новую папку. Чтобы изменить вид каждой дополнительной палитры, поместите файлы PNG (Portable Network Graphics) размером 45x45 пикселей (PaletteIcon.png и PaletteIconHilight.png) в папку дополнительной палитры. Для каждой дополнительной пользовательской палитры можно задать подсказку, для этого вставьте текстовый файл PaletteIcon.txt (с текстом подсказки) в эту же папку.

### Регистрация данных

- [Анализ данных](#)
- [Панель параметров эксперимента](#)
- [Пульт регистрации данных](#)
- [Таблица данных](#)
- [График](#)
- [Управление файлами экспериментов](#)
- [Создание прогноза](#)
- [Удаление данных](#)

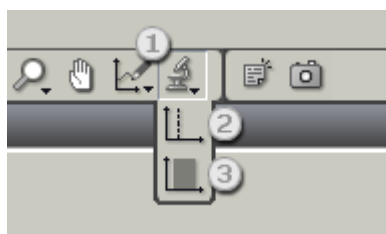
## АНАЛИЗ ДАННЫХ

Функция Анализ данных позволяет исследовать данные в отдельной точке или на определенном участке оси X, включая поиск минимума и максимума, а также линейную аппроксимацию или уравнение заданной линии.

### СОДЕРЖАНИЕ

- [Анализ данных](#)
- [Анализ в точке](#)
- [Анализ участка](#)
- [Совет: перемещение плавающих окон](#)
- [Линейная аппроксимация](#)
- [Совет: как скрыть линии графика при отображении линейной аппроксимации](#)

### СРЕДСТВА АНАЛИЗА



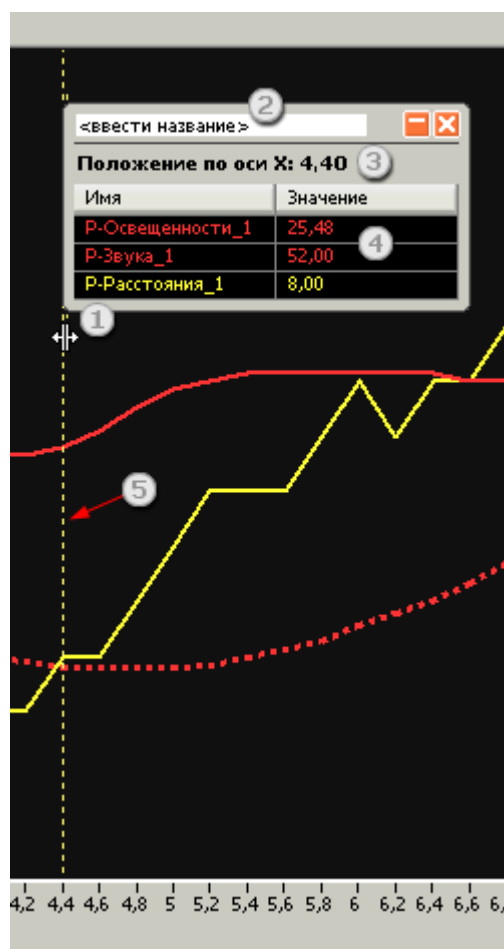
1. Средства анализа

2. Анализ в точке

3. Анализ участка

### АНАЛИЗ ТОЧКИ

Анализ точки - это инструмент исследования данных в определенной точке оси X (оси времени).



1. Переместите курсор: функция становится активной только тогда, когда курсор находится над анализируемым графиком. Переместите курсор на требуемую точку графика, в которой необходимо произвести анализ.

2. Введите Заголовок: чтобы изменить заголовок, заданный по умолчанию, щёлкните на поле ввода «ввести название» и наберите на клавиатуре соответствующий текст.

3. Ось X

4. Значение по оси Y - соответствующее текущему значению точки на оси X для каждого графика, включая прогноз.

5. Текущее положение на оси X.

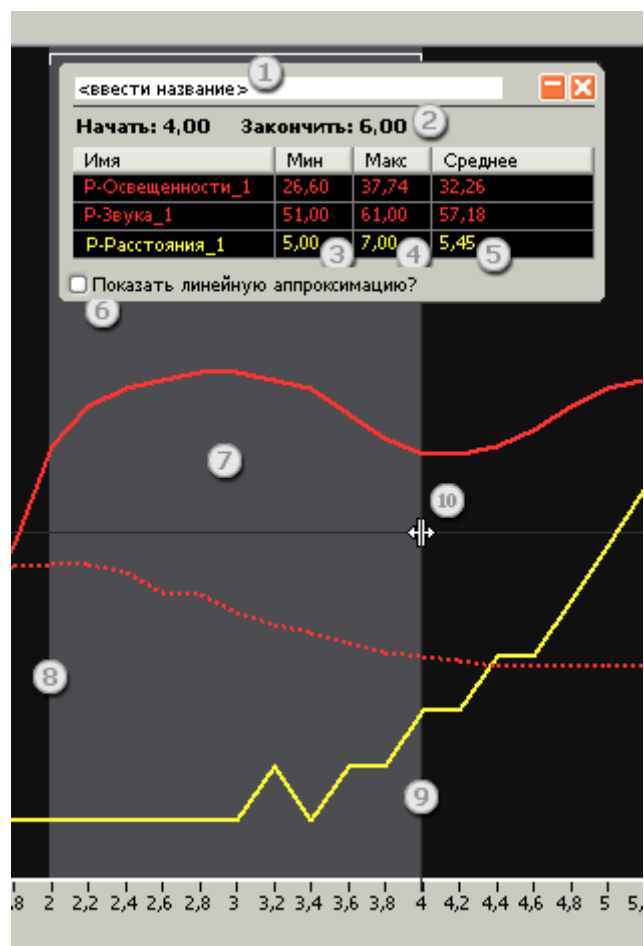
Положение на оси X и значения по оси Y автоматически изменяются при перемещении оси X по графику.

Вы можете создать несколько окон анализа точки. Каждый курсор анализа имеет свой цвет.



## Анализ участка

«Анализ участка» применяют для определённых промежутков времени или участков оси X графика.



1. Введите Заголовок: чтобы изменить заголовок, заданный по умолчанию, щёлкните на поле ввода «ввести название» и наберите на клавиатуре соответствующий текст
2. Граничные значения участка.
3. Минимум: это минимальное значение для каждого прогноза и набора данных в пределах между начальным (левым) и конечным (правым) положениями
4. Максимум: это максимальное значение для каждого прогноза и набора данных в пределах между начальным (левым) и конечным (правым) положениями
5. Среднее: это среднее значение для каждого прогноза и набора данных в пределах между начальным (левым) и конечным (правым) положениями
6. Показать линейную аппроксимацию : см. [ниже](#)
7. Анализируемый участок между начальным (левым) и конечным (правым) положениями выделен светло-серым цветом
8. Левый предел анализируемого участка оси X
9. Правый предел анализируемого участка оси X
10. Переместите границу участка: функция активизируется только тогда, когда курсор находится на границе участка. Переместите границы на точки графика, между которыми необходимо произвести анализ.

Граничные значения участка автоматически изменяются при перемещении курсора по графику.

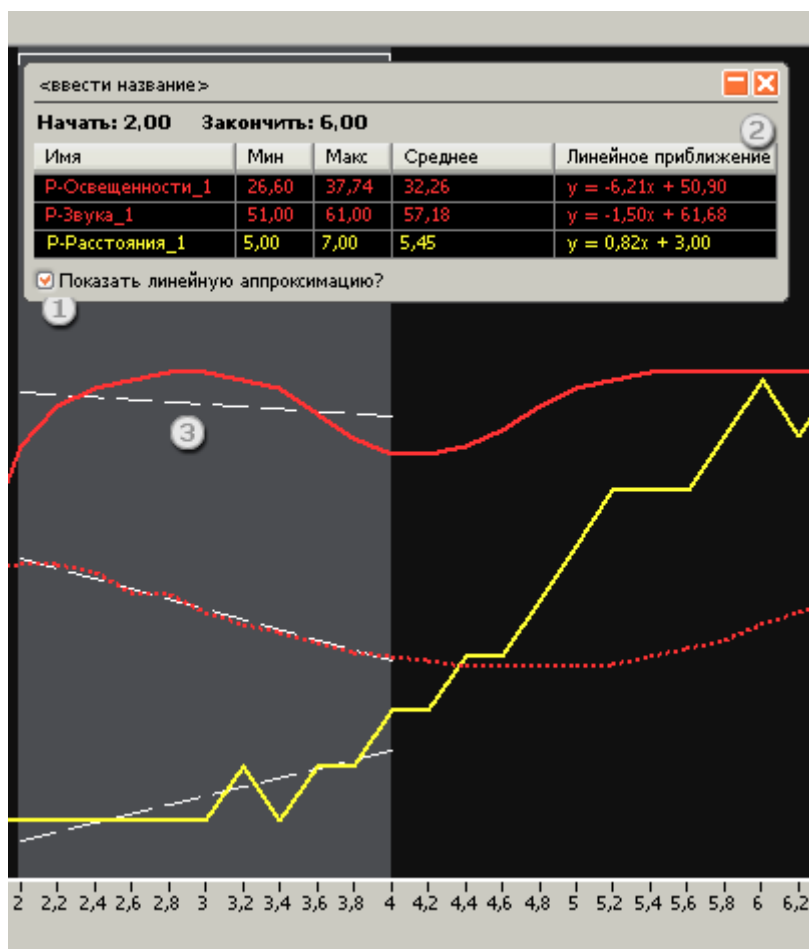
Вы можете создать несколько окон анализа участка.



#### Совет: перемещение плавающих окон

Щёлкните на любом месте окна и, удерживая кнопку мыши, перетащите окно в любое место на графике. Плавающее окно вернется в исходное положение относительно оси X, если щёлкнуть на курсоре.

## Линейная аппроксимация



1. Показать линейную аппроксимацию: нажмите эту кнопку, чтобы отобразить значения линейной аппроксимации
2. Линейная аппроксимация: представляет собой уравнение ( $y=mx+b$ ), описывающее исследуемый участок
3. График линейной аппроксимации: график уравнения линейной аппроксимации, показанной в плавающем окне.

Установите флажок Отобразить линейную аппроксимацию, чтобы показать уравнение и белую пунктирную прямую, соответствующие анализируемому участку графика. Уравнение выводится автоматически.

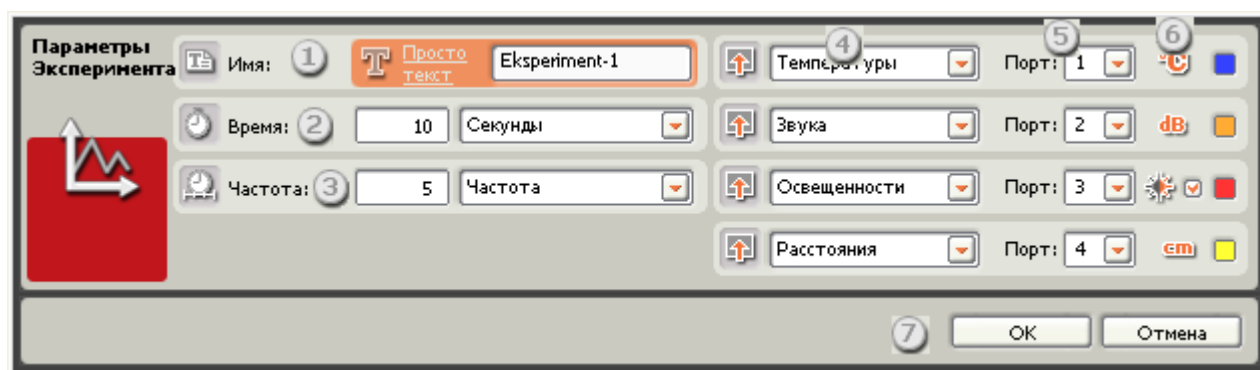


**Совет: как скрыть линии графика при отображении линейной аппроксимации**

Те или иные линии графика можно Показать/Скрыть при помощи пиктограммы с изображением глаза в разделе Таблица Данных. Если линия графика скрыта, из информационных полей также исчезают и соответствующие им значения. Таким образом, скрыв ненужные линии, можно отобразить и просмотреть линейную аппроксимацию для анализируемых линий графиков. Это может быть полезно также при создании снимков окна эксперимента, где график будет легко читаться.

## Окно «Параметры эксперимента»

Окно «Параметры эксперимента» предназначено для задания или изменения параметров новых или существующих экспериментов. Для каждого эксперимента должны быть заданы название, продолжительность, частота замеров и, как минимум, один датчик.



## Настройка параметров эксперимента осуществляется за 8 шагов

Шаг 1: В поле Имя введите название эксперимента или измените существующее.

Шаг 2: В поле Время укажите, в течение какого времени будет осуществляться сбор данных. Выберите секунды (по умолчанию) или минуты.

Шаг 3: В поле Частота укажите, через какой промежуток времени будет осуществляться регистрация данных. Выберите Частоту (количество измерений в секунду) или Секунд между замерами.

Шаг 4: В полях Датчик, выберите датчик или датчики, которые будут использованы.

Шаг 5: В открывающемся списке Порт выберите порты NXT, соответствующие заданным датчикам.

Шаг 6: Щёлкните на пиктограмме, соответствующей нужной единице измерения, чтобы выбрать её. Таким же образом можно сменять единицы измерения.

Шаг 7: Для продолжения щёлкните ОК. Чтобы закрыть Окно «Параметры эксперимента», щёлкните Отмена.

## Содержание

- [Создание нового эксперимента](#)
- [Изменение параметров эксперимента](#)
- [Время](#)
- [Частота замеров](#)
- [Совет: как выбрать оптимальную частоту замеров](#)
- [Выбор датчиков](#)
- [Выбор портов](#)
- [Выбор единиц измерения](#)

## Создание нового эксперимента



Создать новый эксперимент можно тремя способами:

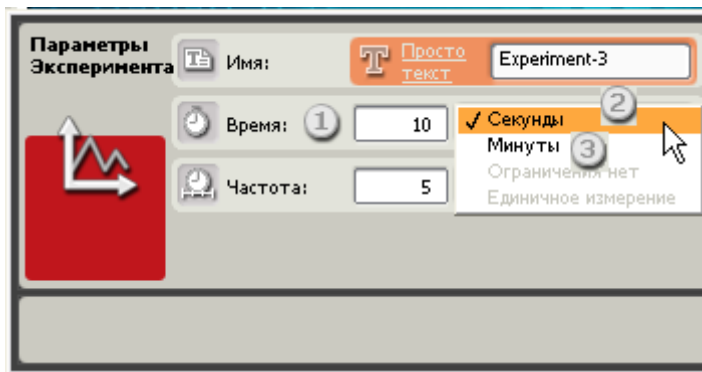
1. В стартовом окне программы, в секции Новый эксперимент, нажать кнопку Далее.
2. Нажать кнопку Новый эксперимент на панели инструментов
3. Выбрать команду Новый в меню Файл

## Изменение параметров эксперимента



Для изменения параметров эксперимента откройте его вкладку, а затем щелкните на пиктограмме Параметры Эксперимента. Откроется Окно "Параметры эксперимента".

Время

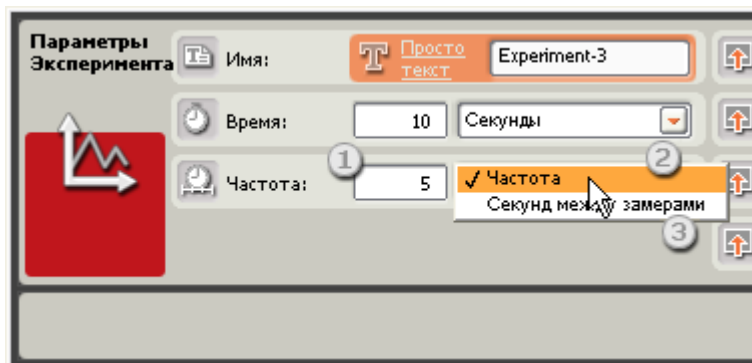


Параметром Время задаётся интервал времени, в течение которого будет проводиться эксперимент. Значения времени откладываются по оси X.

1. Время (число): Введите числовое значение продолжительности эксперимента, которое будет отложено по оси X.
2. Секунды: для кратковременных экспериментов выберите Секунды.
3. Минуты: для более продолжительных экспериментов выберите Минуты.

Не путайте Время и Частоту. Время определяет общую продолжительность эксперимента, то есть промежуток времени, в течение которого NXT будет производить сбор данных. Параметр Частота определяет количество измерений в единицу времени.

### Частота замеров



1. Частота замеров (число): Введите значение. Максимальная частота замеров зависит от типа и количества выбранных датчиков.



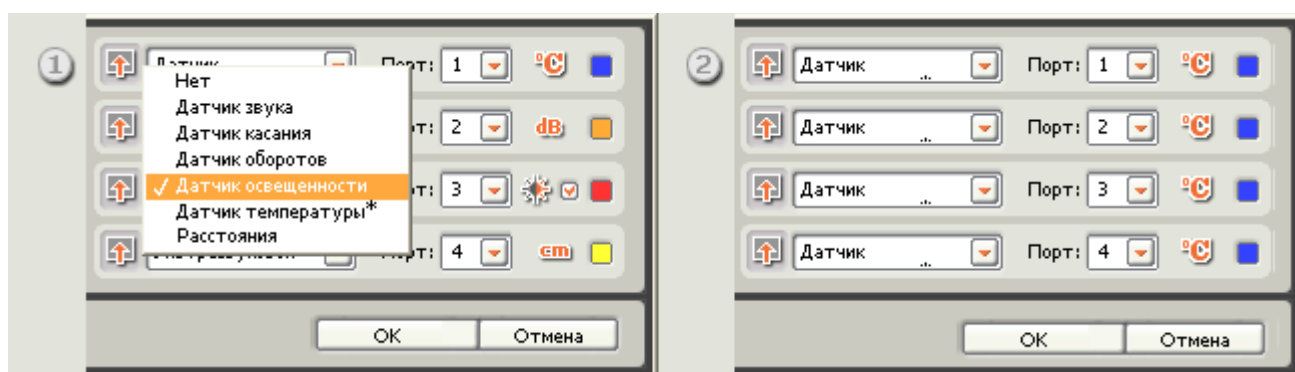
2. Частота: Удобнее для определения частоты замеров кратковременных экспериментов, протекающих в течение секунд или минут.

3. Секунд между замерами: Удобнее для продолжительных экспериментов, идущих в течение часов или дней.

 Совет: как выбрать оптимальную частоту замеров

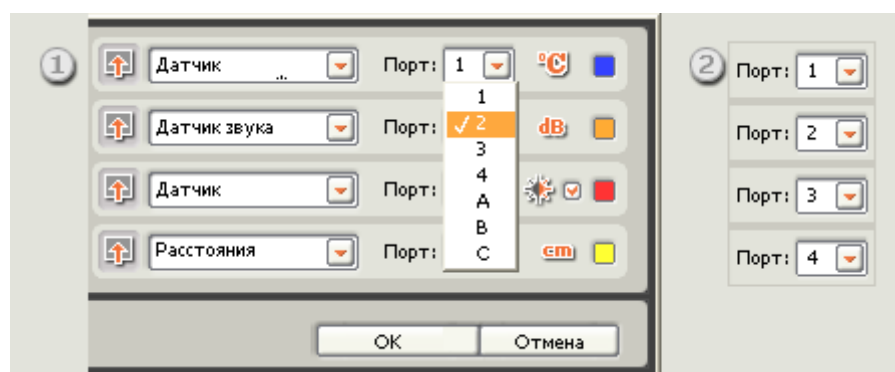
Просто представьте, какой объем данных будет получен, если эксперимент продлится 1 час при частоте замеров 1 раз в секунду. Получится 3600 замеров! Или как мало данных будет собрано, если эксперимент продлится 10 секунд при пятисекундном интервале между замерах. Всего три замера: на 0, 5 и 10 секундах!

## Выбор датчиков



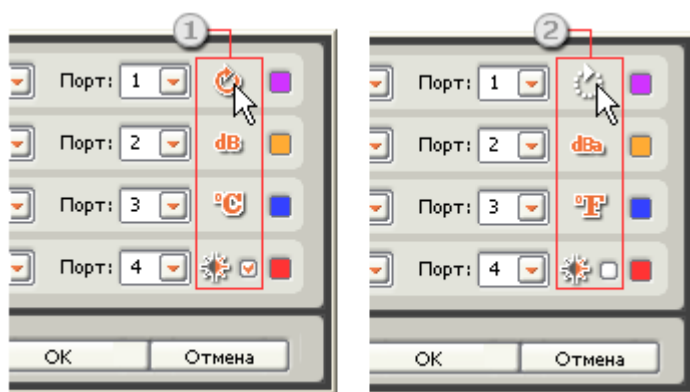
Можно выбрать от одного до шести датчиков (см. рис. 1). Датчики одного типа могут назначаться и на два, и даже на четыре порта (см. рис. 2).

## Выбор портов



В открывающемся списке Порт можно выбрать какое-либо число или букву, соответствующие порту датчика (см. рис. 1). Помните, что в Программе регистрации данных порты для подключения датчиков по умолчанию пронумерованы от 1 до 4 (см. рис. 2).

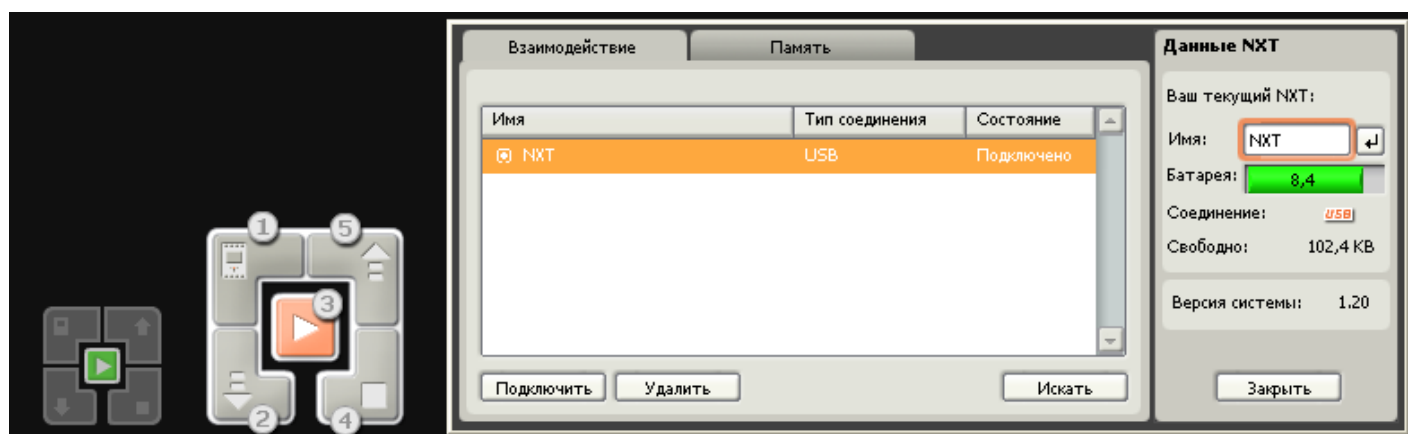
## Выбор единиц измерения



Щёлкая на соответствующих пиктограммах (см. рис. 1 и 2) можно менять единицы измерения. Единицу измерения также можно сменить в Таблице данных. Обратите внимание, что не все датчики имеют выбор единиц измерения.

## ПУЛЬТ РЕГИСТРАЦИИ ДАННЫХ

Пульт Регистрации данных представляет собой интерфейс между программой регистрации данных и NXT. Пульт Регистрации данных позволяет открывать окно NXT, загружать и получать от NXT файлы экспериментов, загружать, запускать и останавливать активные эксперименты.



Пульт остается неактивным (см. рис. 1) до тех пор, пока на него не будет наведен указатель мыши. После этого вместо небольшой серой пиктограммы отображается большая, полностью интерактивная версия пульта (см. рис. 2). Пульт Регистрации данных позволяет выполнять следующие пять функций:

1. **Окно NXT:** Пульт Регистрации данных открывает окно NXT (см. рис. 3), которое позволяет подключить компьютер к NXT, просмотреть важную информацию о нем, а также управлять его файлами. Если память NXT при регистрации данных полностью заполняется, можно удалить файлы экспериментов на вкладке Память. Более полная информация об окне NXT приведена в Разделе [Файлы и память NXT](#).
2. **Загрузка в NXT:** Пульт Регистрации данных загружает программы в NXT, что позволяет использовать его удаленно при проведении экспериментов и регистрации данных в соответствии с программой.
3. **Загрузка и запуск:** с Пульта Регистрации данных можно загружать и запускать выбранные эксперименты.
4. **Останов:** Пульт Регистрации данных позволяет остановить эксперимент, запущенный на NXT.

5. Передача на компьютер: Пульт Регистрации данных позволяет предавать файлы экспериментов с NXT на компьютер. Файлы экспериментов на NXT могут быть получены в ходе регистрации данных экспериментов, либо из любой программы, выполняемой устройством. Эксперименты можно проводить и регистрировать их данные удаленно: подключать NXT к компьютеру даже после завершения работы.

Более подробная информация о файлах экспериментов приведена в разделе [Управление файлами экспериментов](#).

## Таблица данных

### Содержание

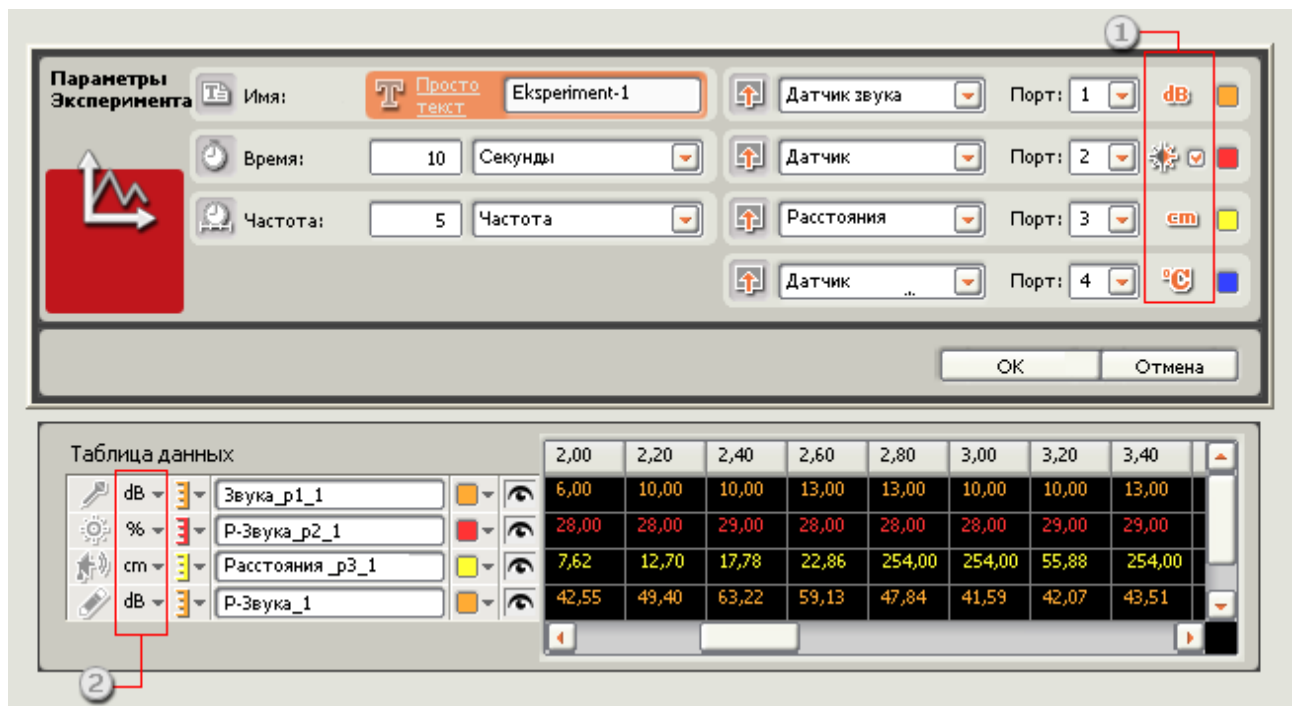
- [Таблица данных](#)
- [Таблица свойств датчиков](#)
- [Изменение единиц измерения](#)
- [Различие между дБ и дБА](#)
- [В чем заключается различие между оборотами и градусами?](#)
- [Перенесение набора данных на ось](#)
- [Изменение названия линии графика](#)
- [Выбор цвета и стиля линий графика](#)
- [Скрытие/Отображение линий графика](#)
- [Редактирование значений прогноза](#)

## Таблица данных



1. Пиктограмма датчика
2. Единица измерения для оси Y
3. Перенос набора данных на ось Y
4. Название линии графика
5. Цвет и стиль линий графиков
6. Скрытие/отображение линий графиков
7. Значения по оси Y
8. Значения по оси X

## Изменение единиц измерения



Изменить единицы измерения можно одним из двух способов:

1. В окне параметров эксперимента: переключение между доступными единицами измерения производится щелчком в поле Единица измерения.
2. В таблице данных: единицу измерения выбирают в открывающемся списке.

При изменении единицы измерения шкала оси Y автоматически изменяется.

## Различие между дБ и дБА



**дБ:** При регистрации громкости звука в децибелах, датчик не калибруется в соответствии с диапазоном восприятия человеческого уха. Поэтому могут быть зарегистрированы звуки, слишком громкие или тихие для человеческого уха.

**дБА:** При измерении громкости звука в децибелах, регистрируемых вне диапазона восприятия человеческого уха, чувствительность датчика регулируется и адаптируется к этому диапазону. Другими словами, регистрируются шумы, которые может услышать ухо человека.

## В чем заключается различие между оборотами и градусами?

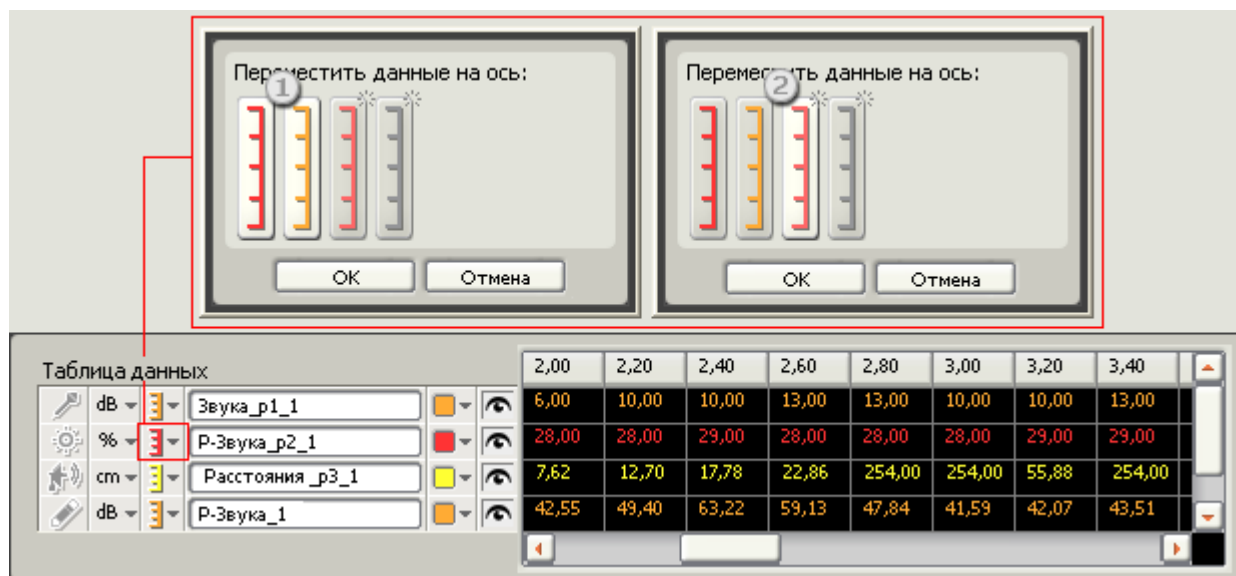


(Об.) - сокращённо от "оборот". Величина, выраженная в оборотах, представляет собой количество полных оборотов, совершенных валом мотора NXT при вращении вперед или назад. Один оборот соответствует длине окружности колеса.

Измерение в оборотах больше подходит для экспериментов, в которых мотор NXT совершает большое количество оборотов.

(°) - обозначение градуса. Величина, выраженная в градусах, представляет собой количество градусов, на которые повернулся вал мотора NXT при вращении вперед или назад. Полный оборот соответствует 360 градусам. Измерение в градусах больше подходит для экспериментов, в которых вал мотора NXT совершает менее двух полных оборотов, а также при необходимости точных измерений.

## Перенос набора данных на ось



1. Существующая ось Y: цвета соответствуют цветам осей Y, существующим на графике.

2. Новая ось Y: перенесите существующий набор данных на новую ось Y, отмеченную звездочкой в верхнем левом углу.

Выбрать линию на графике можно одним из трёх способов:

- щёлкнуть на линии графика
- щёлкнуть на оси Y
- щёлкнуть на названии или строке в таблице данных, соответствующих графику

Наборы существующих или прогнозируемых данных можно переносить с нескольких осей Y на одну. Также можно переносить такие данные с одной оси Y на разные, уже имеющиеся или новые оси Y.

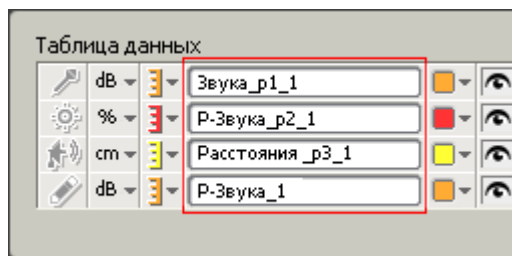
Перенос данных может оказаться полезным в следующих случаях:

- Когда две линии графика отображают сильно различающиеся значения: например, очень большие и слишком малые.
- Когда один график отображает равные или почти равные значения, а другой - совершенно разные.
- Когда необходимо отделить линии графика друг от друга для оптимизации работы функции Автомасштабирования.
- Когда необходимо отобразить показания однотипных датчиков на одной оси Y. Например, при одновременном использовании нескольких датчиков одного и того же типа, программа регистрации данных автоматически создает для каждого из них ось Y. Поэтому, если в эксперименте используется три датчика освещённости, будет создано три оси Y. Чтобы объединить их в одну нужно три раза повторить процедуру указания другой оси.
- Когда необходимо переместить одну из нескольких линий графика, построенных по показаниям датчиков одного и того же типа, и нанесенных на одну и ту же ось Y, на новую ось. Например, на одну и ту же ось Y нанесены три набора прогнозируемых данных для датчика оборотов, и в качестве единицы измерения выбраны обороты. Один из прогнозов можно переместить на новую ось Y и

изменить единицу измерения на градусы. При этом единица измерения двух остальных прогнозов не изменится.

- Когда необходимо на бумаге вручную отобразить все графики в одной системе координат.

## Изменение названия линии графика

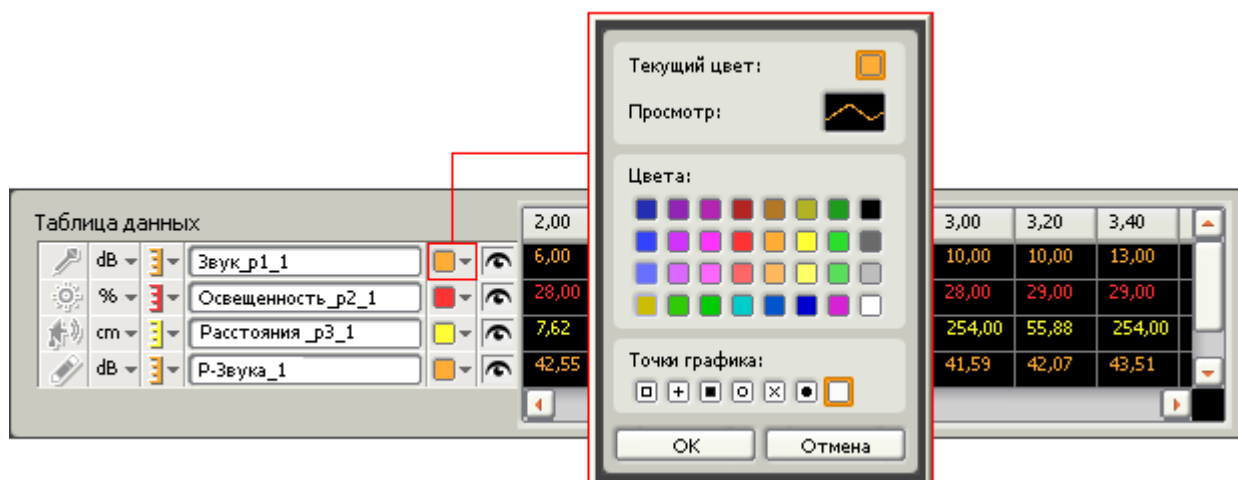


Название любой линии графика можно изменить в соответствующем поле таблицы данных. Для этого просто введите туда нужное название.

В программе регистрации данных по умолчанию используется интуитивно понятная система названий:

- Прогнозы: "Р-[имя датчика]\_#". "Р" информирует о том, что это - прогноз, а "#" представляет собой порядковый номер прогноза
- Наборы данных: "[ имя название датчика]\_р#\_#". "р#" указывает на номер порта, к которому подключен датчик, а "#" - порядковый номер эксперимента

## Выбор цвета и стиля линии графика



Для каждого набора данных или прогноза можно выбрать различные цвета и стили линий графиков.

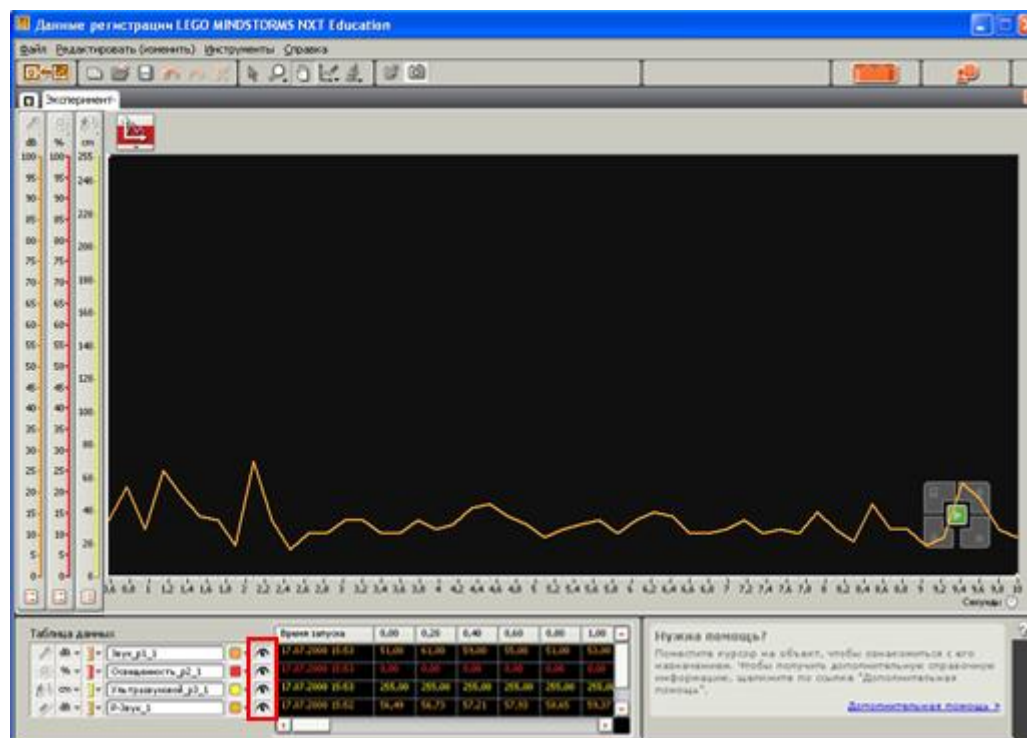
По умолчанию прогнозы всегда отображаются пунктирными линиями, а наборы данных - сплошными. Линии обоих типов также можно отображать используя различные значки.

По умолчанию для отображения прогнозов и наборов данных, полученных от одного и того же датчика, используются линии одинакового цвета. Чтобы лучше различать такие прогнозы и наборы данных, цвета соответствующих линий можно изменять.

При печати в монохромном или черно-белом режиме все линии будут одинакового цвета. Поэтому использование различных стилей для отображения наборов данных от разных датчиков сделает распечатку более понятной.

Цвет линий автоматически становится более бледным по мере регистрации новых данных и построения графиков новых прогнозов. Чем ярче цвет линии, тем более свежим данным или прогнозам она соответствует. При достаточно большом количестве данных, более ранние данные или прогнозы начинают отображаться серым цветом.

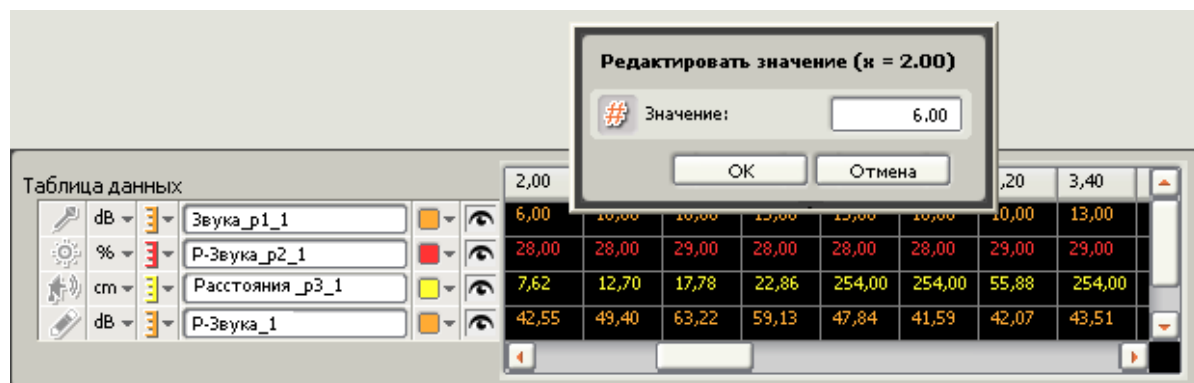
## Скрытие/Отображение линий графика



Щёлкая на пиктограмме с изображением глаза, можно скрывать или показывать линии графика.

Предположим, вы анализируете три прогноза и три набора данных, представленных на одном и том же графике. При этом количество отображенной информации может существенно затруднить такой анализ. Скрывая те или иные линии, можно упростить анализ других линий на графике.

## Редактирование значений прогноза



Прогнозируемые значения, занесенные в таблицу данных, всегда можно изменить. Для этого нужно дважды щёлкнуть на ячейке, содержимое которой следует изменить, а затем отредактировать это значение в открывшемся диалоговом окне.

## График

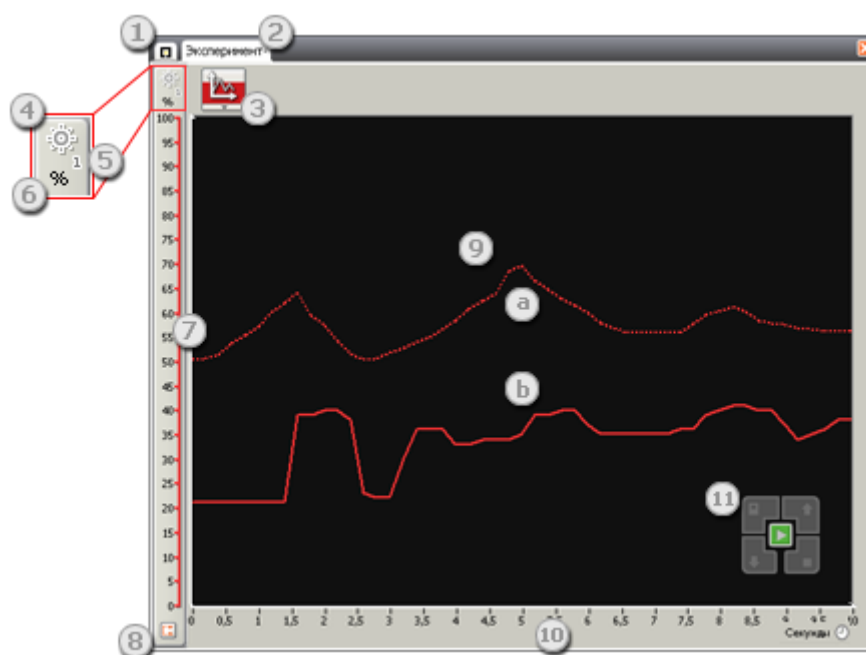


График - это визуальное представление файла протокола. Для отображения прогнозов и/или наборов данных используется одна ось X и одна или несколько осей Y. Кроме того, на графике имеется контроллер, панель параметров эксперимента, а также закладки экспериментов.

## Содержание

- [График](#)
- [Курсор-перекрестье](#)
- [Самоучитель и Мой Портал](#)
- [Лобби и закладки экспериментов](#)
- [Режим масштабирования / Переключение](#)
- [Совет: Когда следует использовать режим масштабирования?](#)
- [Цвета линий на графике](#)

## График



1. Закладка Лобби
2. Закладки экспериментов
3. Панель параметров эксперимента
4. Пиктограмма датчика для оси Y
5. Порт датчика для оси Y
6. Единица измерения для оси Y
7. Диапазон измерения для оси Y
8. Переключатель режима масштабирования для оси Y: переключение между полным и автоматическим масштабом
9. Линия графика или просто линия: выберите одну из линий, щелкнув по ней на графике

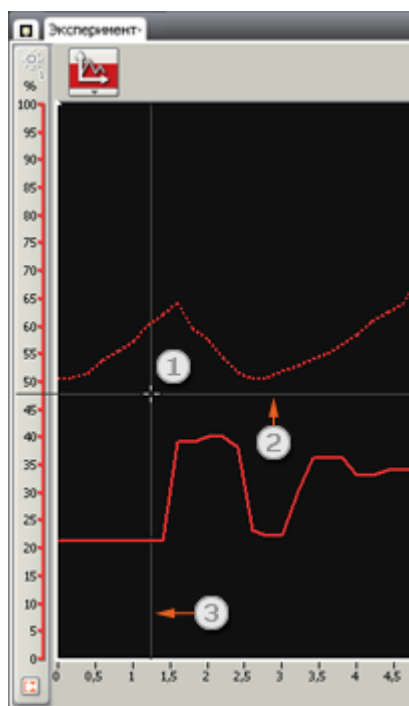


- a. Прогноз: всегда отображается точечной линией
- b. Набор данных: всегда отображается сплошной линией

10. Продолжительность по оси X: отображается в секундах или минутах

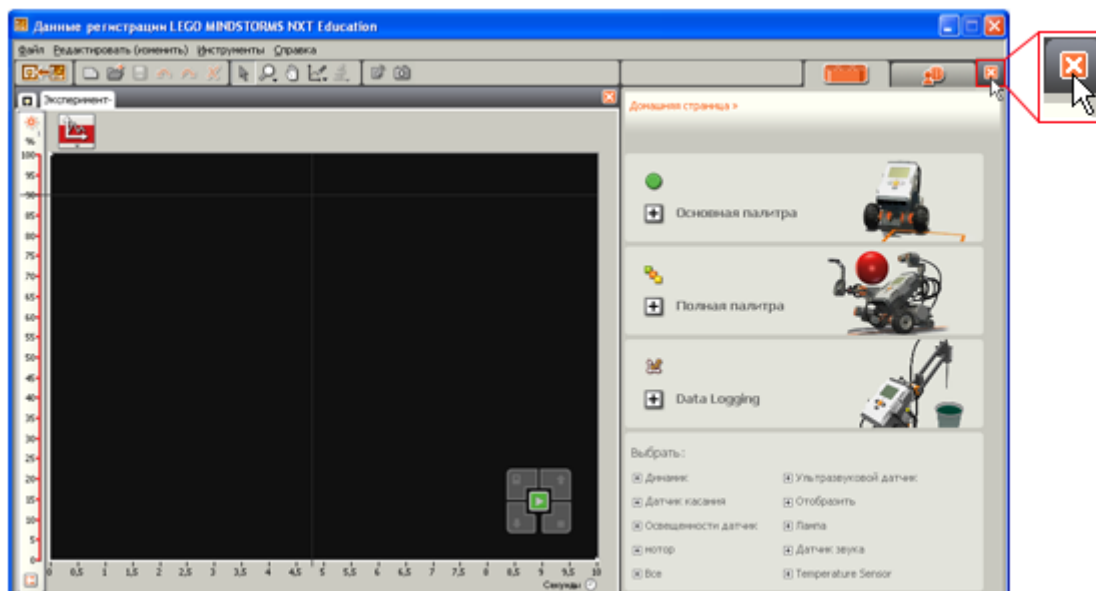
11. Контроллер регистрации данных

## Курсор-перекрестье



1. Курсор-перекрестье: при помощи курсора-перекрестья можно быстро определить приблизительные значения по осям X и Y графика. Переместите курсор к нужной точке на графике, а затем при помощи линий, исходящих из перекрестья, прочтите соответствующие этой точке числа на осях.
2. Линия, исходящая из перекрестья к оси Y
3. Линия, исходящая из перекрестья к оси X

## Самоучитель и Мой Портал



Для того чтобы развернуть график, нажмите на кнопку "Заккрыть Академию" в верхнем правом углу.

## Лобби и закладки экспериментов



Можно создать столько экспериментов, сколько необходимо. Просмотр имеющихся экспериментов осуществляется при помощи закладок экспериментов. Если вы создадите больше экспериментов, чем может быть отображено над графиком, справа от закладок экспериментов появятся навигационные стрелки (см. рис. 2). Используйте эти стрелки, для перехода от одного эксперимента к другому.

Для перехода к лобби системы регистрации данных нажмите на самой левой закладке.

## Режим масштабирования / Переключение



Вы можете переходить от принятого по умолчанию диапазона измерений к реальному диапазону данных, полученных от датчика, при помощи пиктограммы переключения режима масштабирования, которая расположена у нижнего края оси Y.

1. Режим полного масштаба: значения отображаются в масштабе, соответствующем принятому по умолчанию для данного датчика диапазону измерения (см. рис. 1)
2. Режим автоматического масштаба: значения отображаются в масштабе, соответствующем реальному диапазону данных, полученных от датчика в ходе эксперимента (см. рис. 2)

Вы можете выбирать режим масштабирования для каждой оси Y независимо от других осей. Это означает, что переход от полного масштаба к автоматическому на одной из осей Y никак не скажется на масштаб на другой оси Y.

Переключать режим масштабирования можно в любое время, в том числе - во время эксперимента.

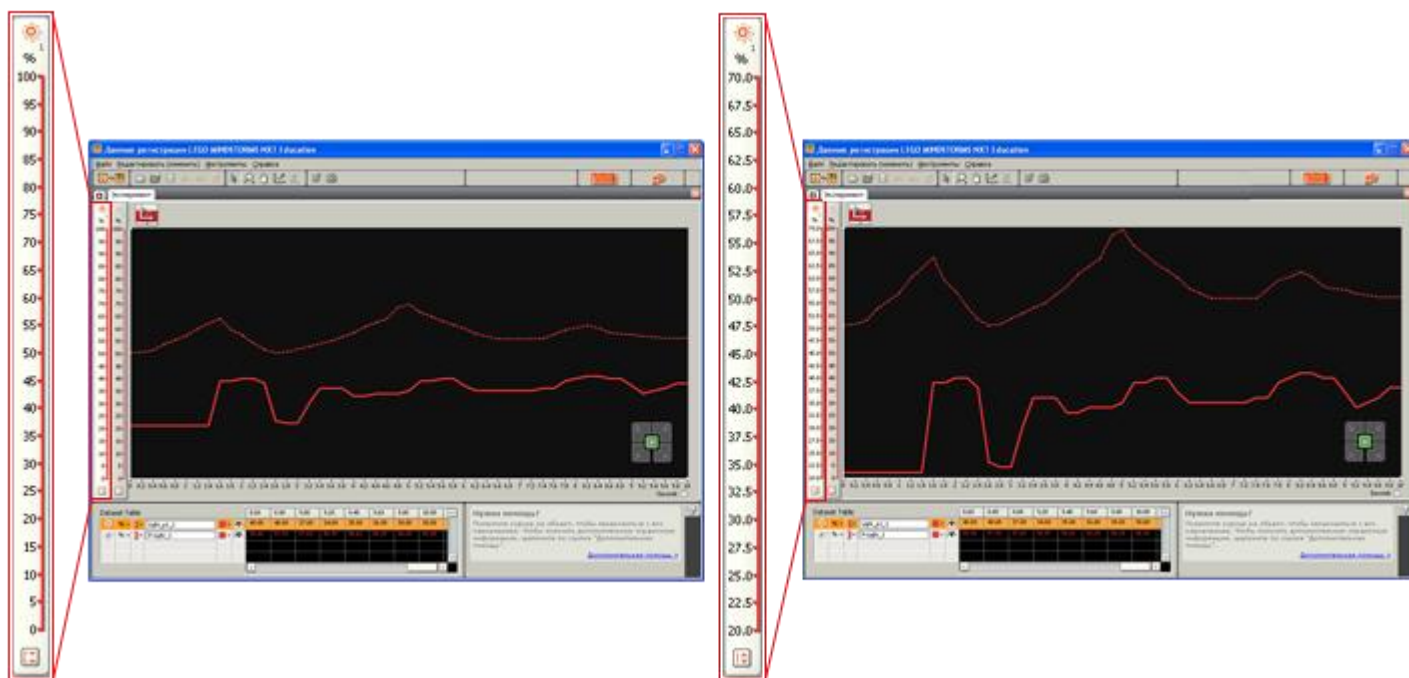


Совет: Когда следует использовать режим масштабирования?

Прогнозы или наборы данных, полученных в ходе эксперимента, часто имеют очень небольшой разброс, что затрудняет анализ пиков и спадов (см. рис. 1). Переход от полного масштаба к автоматическому может значительно облегчить рассмотрение этих различий (см. рис. 2).

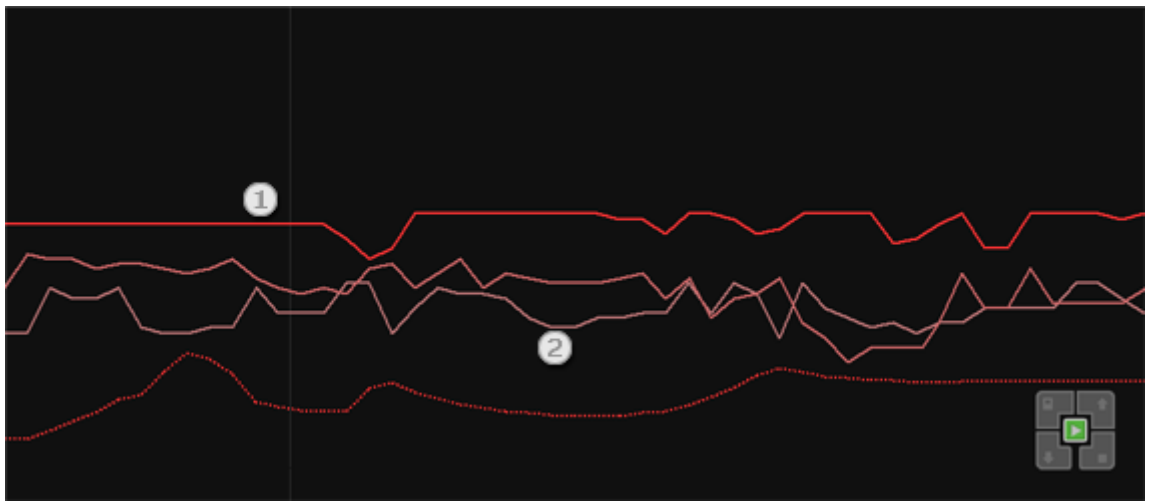
Например, в результате эксперимента, в котором задействован датчик температуры, может быть получен набор данных, значения которых не будут слишком отличаться друг от друга. На всем протяжении эксперимента разница между значениями может быть не более четырех-пяти градусов (например температура может падать до 18 °C (64 °F) или подниматься до 22 °C (72 °F)]. Переключение на автоматический масштаб значительно облегчит анализ этих небольших перепадов, так как при этом сужается диапазон измерений, а пики и спады становятся больше.

## Цвета линий на графике



Для того чтобы линии, построенные по данным, полученным от различных датчиков, лучше различались на графике, с каждым из датчиков связан определенный цвет, которым отображаются соответствующие ему линии.

Заданный по умолчанию цвет можно изменив, отредактировав соответствующее значение в таблице данных. В некоторых случаях назначение пользовательских цветов линий помогает лучше управлять отображением данных на графике.



Цвет линий автоматически делается более бледным по мере регистрации новых данных и построения новых прогнозов. Чем ярче цвет, тем новее данные или прогноз (1). Если собрано или спрогнозировано достаточно данных, устаревшие линии становятся серыми (2).

Представьте себе, что для одного и того же датчика собрано несколько наборов данных и сделано несколько прогнозов. Даже с учетом автоматического изменения яркости цвета на графике окажется множество линий примерно одинакового цвета. Поэтому, для того чтобы упростить чтение такого графика имеет смысл воспользоваться таблицей данных и изменить цвет некоторых линий. При этом они станут больше выделяться на общем фоне.

## Управление файлами экспериментов

Управление файлами экспериментов является важной частью процесса управления экспериментами, наборами данных и файлами экспериментов. Доступ к файлам экспериментов можно получить одним из трех способов: открыть, импортировать и экспортировать из NXT.

### Содержание

- [Открытие эксперимента](#)
- [Импорт эксперимента](#)
- [Копирование файла эксперимента из NXT на ваш компьютер](#)
- [Совет: как работать с файлами экспериментов?](#)
- [Открытие файлов экспериментов в текстовом редакторе](#)
- [Файлы экспериментов в электронных таблицах](#)

### Открытие эксперимента

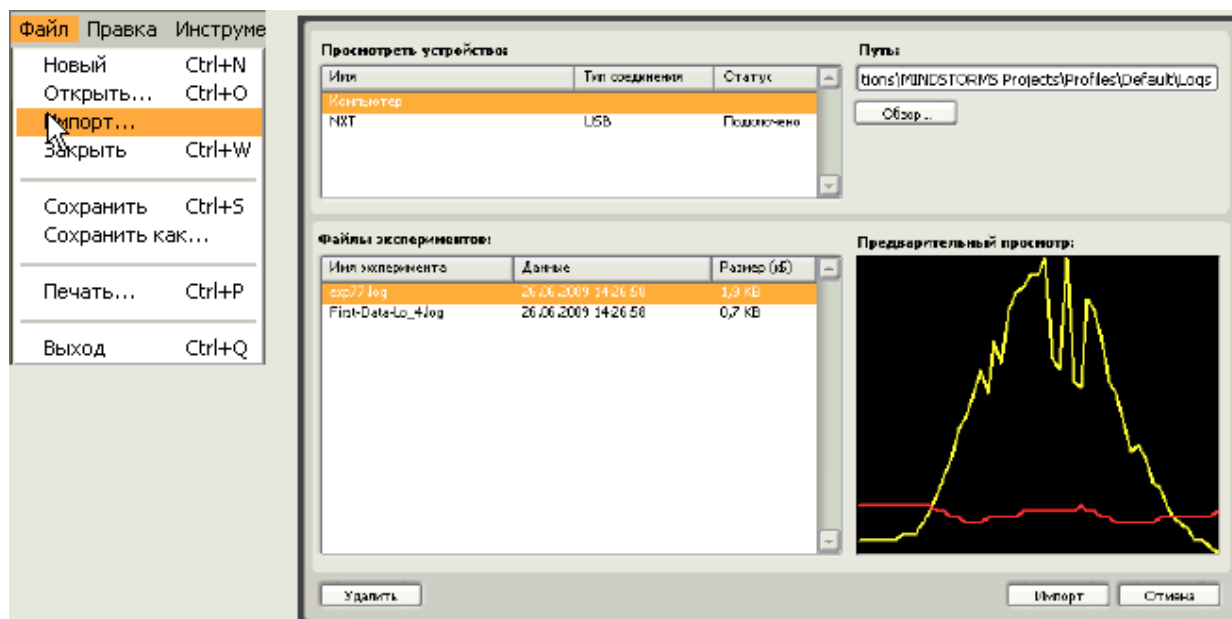


Открыть эксперимент можно одним из трех способов:

1. Открыть эксперимент, с которым ранее проводилась работа на компьютере или NXT, с вкладки Старт.

- Щёлкнув на пиктограмме Открыть Эксперимент на панели инструментов
- Используя пункт Открыть в меню Файл

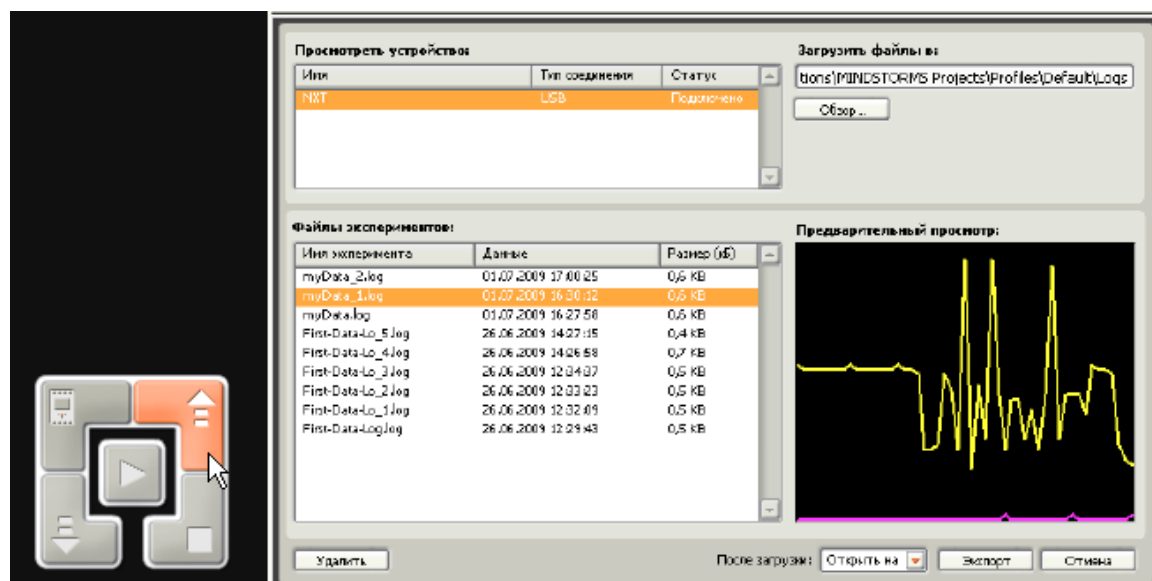
## Импорт эксперимента



Импорт эксперимента позволяет переносить файлы экспериментов, хранящиеся на компьютере или в подключенном NXT, на открытую и активную вкладку эксперимента. Лучше всего этот способ применять, например, в случае, когда вы создаёте эксперимент, делаете прогноз, загружаете данные в NXT и запускаете эксперимент. Затем можно импортировать эксперимент и сравнить собранные данные с исходными прогнозами или сравнить этот эксперимент с другим.

## Копирование файла эксперимента из NXT на ваш компьютер

При этом файлы экспериментов скачиваются из NXT на компьютер.



**Что нужно сделать:**

Экспорт файлов из NXT осуществляется в 3 шага:

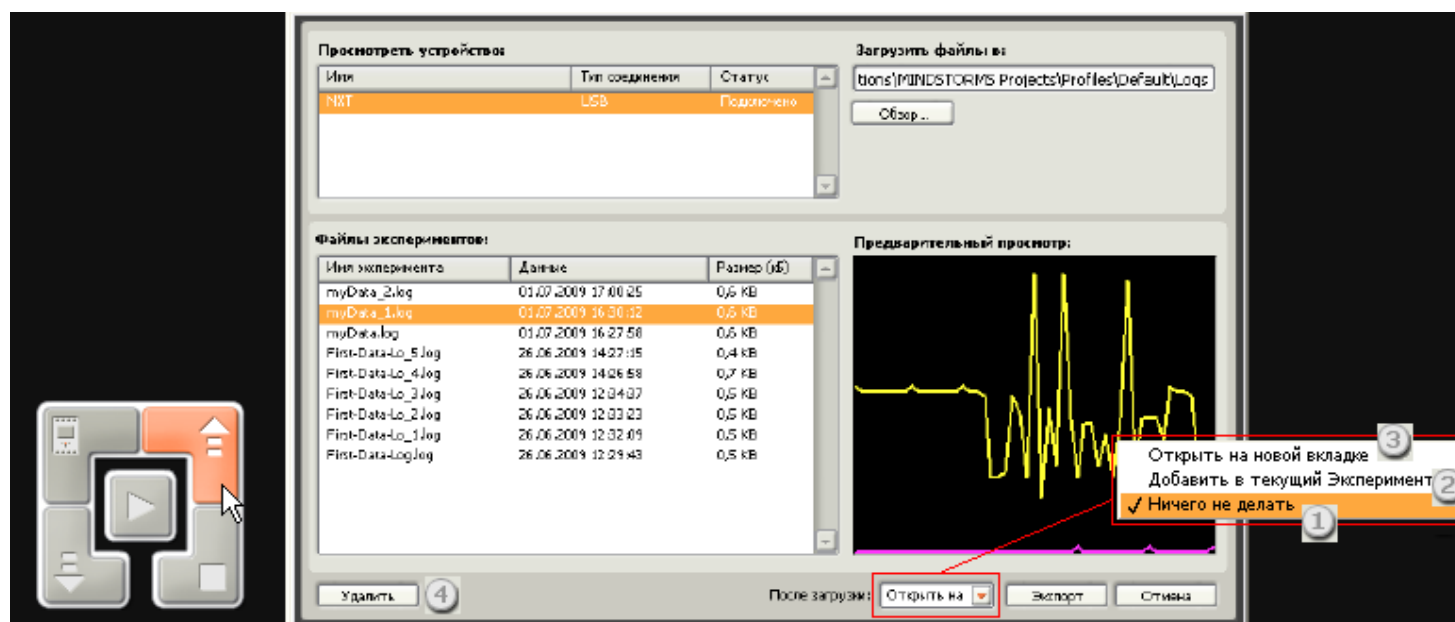
Шаг 1: Убедитесь в том, что выбран нужный NXT, и он подключен к компьютеру.

Шаг 2: Выберите один или несколько файлов экспериментов

Предварительно просмотрите файл, прежде чем щёлкнуть на нём и открыть из списка.

Шаг 3: Примите решение о том, что следует сделать с выбранными файлами экспериментов

(См. ниже)



После экспорта файла из NXT можно выбрать одно из следующих действий:

1. Ничего не делать: сохранить файлы экспериментов в папке, указанной в поле «Загрузить файлы в:»
2. Добавить в текущий эксперимент. После того как импортируемый файл эксперимента будет включен в открытый эксперимент, на графике будут построены новые линии и, при необходимости, созданы новые оси Y.
3. Открыть файл эксперимента на новой вкладке эксперимента.
4. Удалить: удалить нежелательные или ненужные файлы экспериментов из подключенного NXT.

### Совет: как работать с файлами экспериментов?

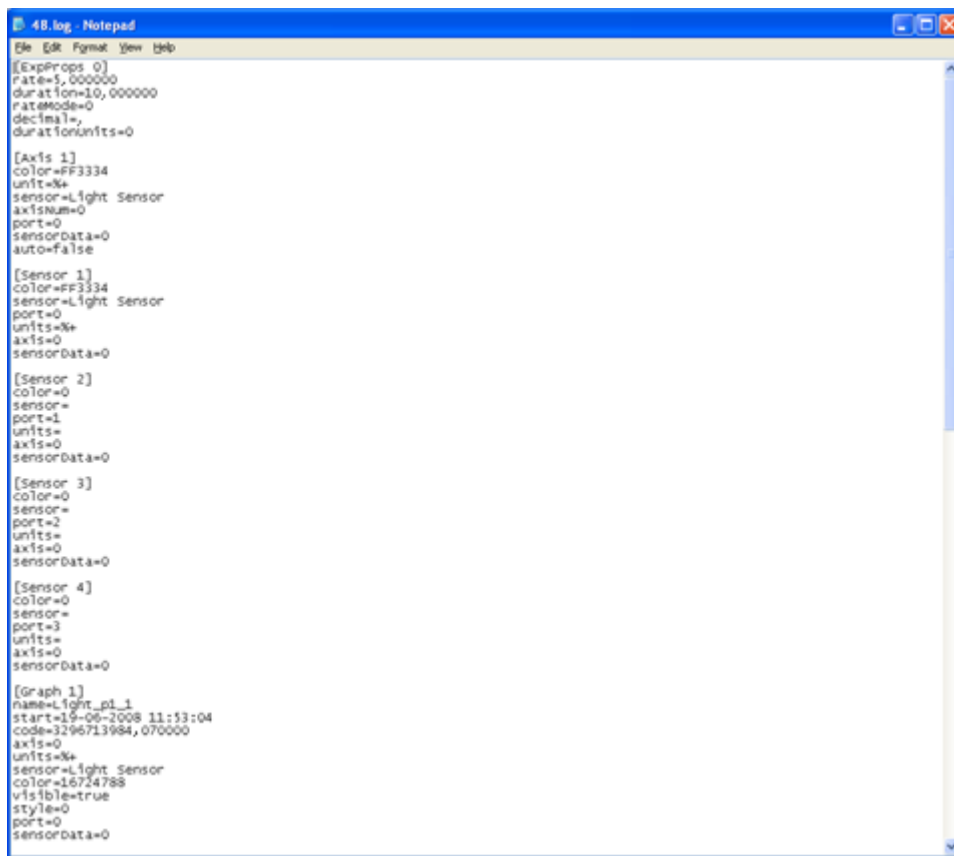
Во избежание необратимого удаления файлов экспериментов из NXT, удостоверьтесь, что при удалении нужный NXT не был выбран.

При скачивании на компьютер, файлы экспериментов из NXT автоматически не удаляются.

Для того чтобы одновременно выбрать несколько файлов экспериментов, удерживайте нажатой клавишу Ctrl или Shift, щёлкая на нужных файлах.

Вы можете отправить файл эксперимента другу или учителю. При этом они могут открыть или импортировать их в своей системе регистрации данных MINDSTORMS NXT.

# Открытие файлов экспериментов в текстовом редакторе



```
4B.log - Notepad
File Edit Format View Help
[ExpProps 0]
rate=5,000000
duration=10,000000
rateMode=0
decimal=
durationUnits=0

[Axis 1]
color=FF3334
unit=%+
sensor=Light Sensor
axisNum=0
port=0
sensorData=0
auto=False

[Sensor 1]
color=FF3334
sensor=Light Sensor
port=0
units=%+
axis=0
sensorData=0

[Sensor 2]
color=0
sensor=
port=1
units=
axis=0
sensorData=0

[Sensor 3]
color=0
sensor=
port=2
units=
axis=0
sensorData=0

[Sensor 4]
color=0
sensor=
port=3
units=
axis=0
sensorData=0

[Graph 1]
name=Light_p1_1
start=19-06-2008 11:53:04
code=3296713984,070000
axis=0
units=%+
sensor=Light Sensor
color=16724788
visible=true
style=0
port=0
sensorData=0
```

Наборы данных автоматически сохраняются как простой текстовый файл, который может быть открыт в электронной таблице или текстовом редакторе. Для сохранения файлов используется путь, заданный по умолчанию, либо указанный вами. Файл экспериментов представляет собой таблицу, в которой имеется один столбец с данными о продолжительности эксперимента (времени) и по одному столбцу для данных, полученных от каждого датчика.

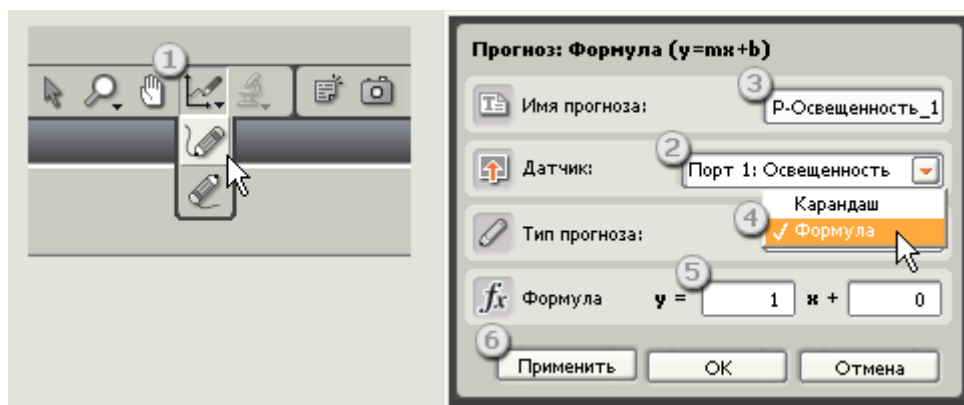
## Файлы экспериментов в электронных таблицах

	A	B	C	D	E	F	G	H	I	J
1	[ExpProps 0]									
2	rate=5,000000									
3	duration=10,000000									
4	rateMode=0									
5	decimal=,									
6	durationUnits=0									
7										
8	[Axis 1]									
9	color=FF3334									
10	unit=%+									
11	sensor=Light Sensor									
12	axisNum=0									
13	port=0									
14	sensorData=0									
15	auto=false									
16										
17	[Sensor 1]									
18	color=FF3334									
19	sensor=Light Sensor									
20	port=0									
21	units=%+									
22	axis=0									
23	sensorData=0									
24										
25	[Sensor 2]									
26	color=0									
27	sensor=									
28	port=1									

Набор данных, сохраненный в файле эксперимента, может быть скопирован в электронную таблицу через буфер обмена.

## Средства создания прогнозов

Функция Средства создания прогнозов используется для прогнозирования и редактирования возможных исходов эксперимента. Для прогнозирования используется два метода: карандаш и формула.



### Содержание

- [Что нужно сделать](#)
- [Создание прогноза методом "карандаш"](#)
- [Создание прогноза методом "формула"](#)
- [Редактирование прогноза, созданного методом "карандаш"](#)
- [Редактирование прогноза, созданного методом "формула"](#)
- [Совет: В каких случаях пользоваться функцией Удаление значений панели инструментов Набор данных?](#)



## Что нужно сделать

Данный процесс прогнозирования состоит из 6 шагов:

Шаг 1: Нажмите на кнопку Средства создания прогнозов и выберите пункт Новый прогноз ниспадающего меню.

Можно создать новый или отредактировать существующий прогноз.

Шаг 2: Выберите датчик, относительно показаний которого будет создаваться прогноз

Шаг 3: Проверьте имя прогноза

В зависимости от выбранного на шаге 2 датчика, по умолчанию прогнозу присваивается имя, которое может быть впоследствии изменено.

Шаг 4: Выберите тип прогноза

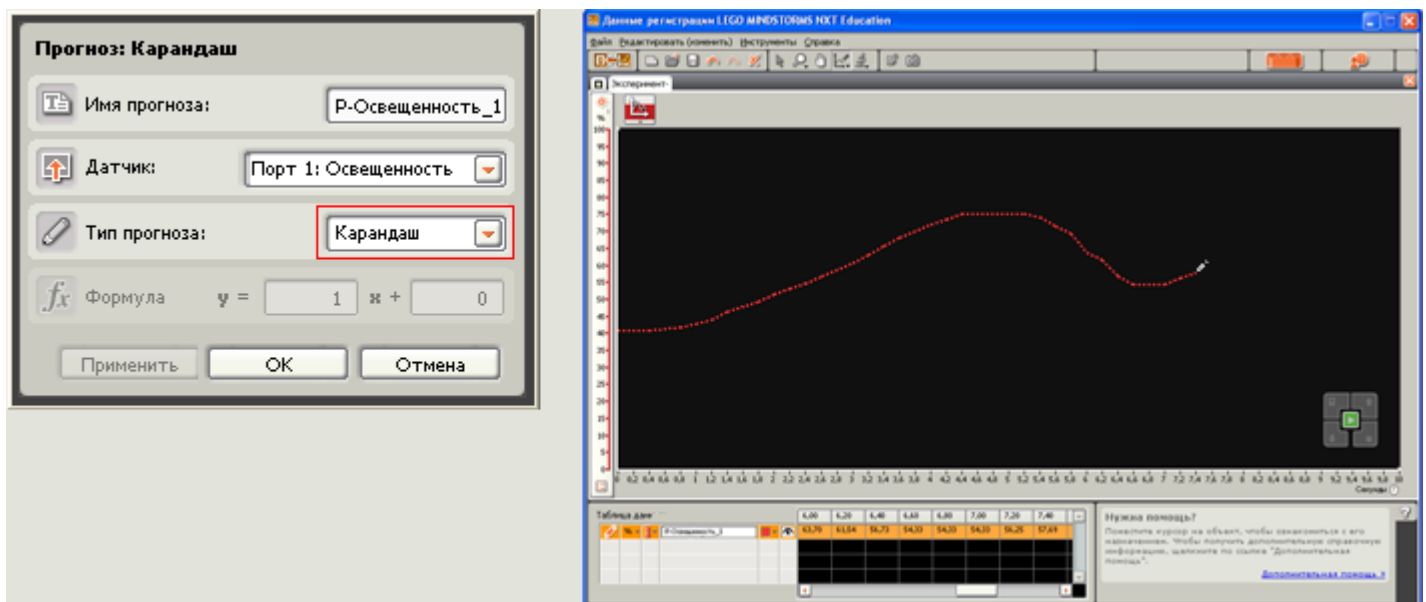
Выберите пункт Карандаш или Формула ниспадающего меню. При выборе пункта меню Формула автоматически становится активным нижнее поле ввода.

Шаг 5: Если вы выбрали пункт меню Формула, введите параметры

Шаг 6: Для того чтобы ознакомиться с представлением формулы на графике, воспользуйтесь предварительным просмотром, нажав на кнопку Применить

## Создание прогноза методом "карандаш"

Для создания прогноза вручную выберите пункт меню Карандаш.



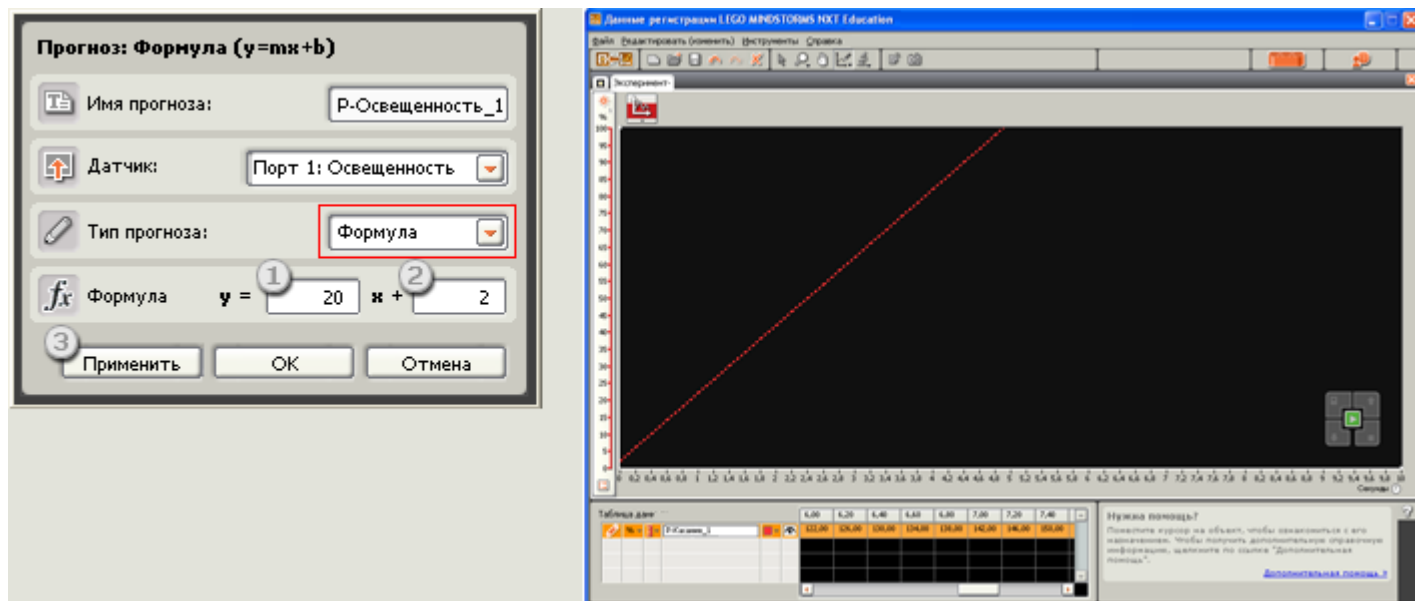
Для того чтобы создать прогноз методом "карандаш", нажмите на левую клавишу мыши и, удерживая ее нажатой, переместите курсор, который принимает форму карандаша, по графику. Кроме того, можно несколько раз щелкнуть левой клавишей мыши по графику. При этом линия прогноза будет проходить через указанные точки.

При нажатии на определенную точку на линии прогноз будет точно проведен через эту точку на графике.

Для того чтобы методом "карандаш" построить прямую линию, щелкните в произвольной точке графика, затем переместите курсор мыши в другое место и снова щелкните. При этом между двумя точками будет автоматически построена прямая линия.

## Создание прогноза методом "формула"

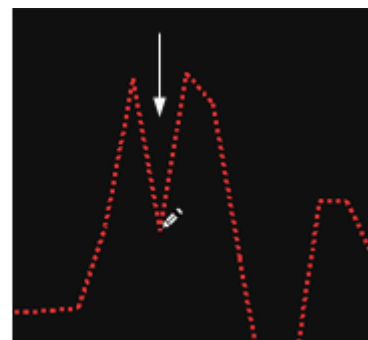
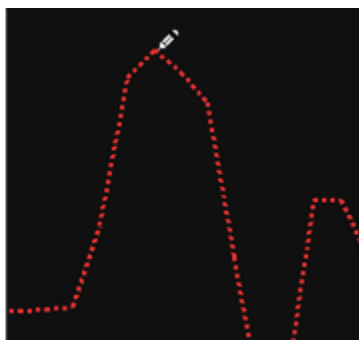
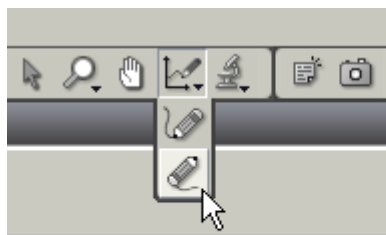
Для того чтобы построить прямую линию прогноза на основе линейной формулы, выберите пункт меню "Формула".



При создании прогнозов методом "формула" важно указать два значения:

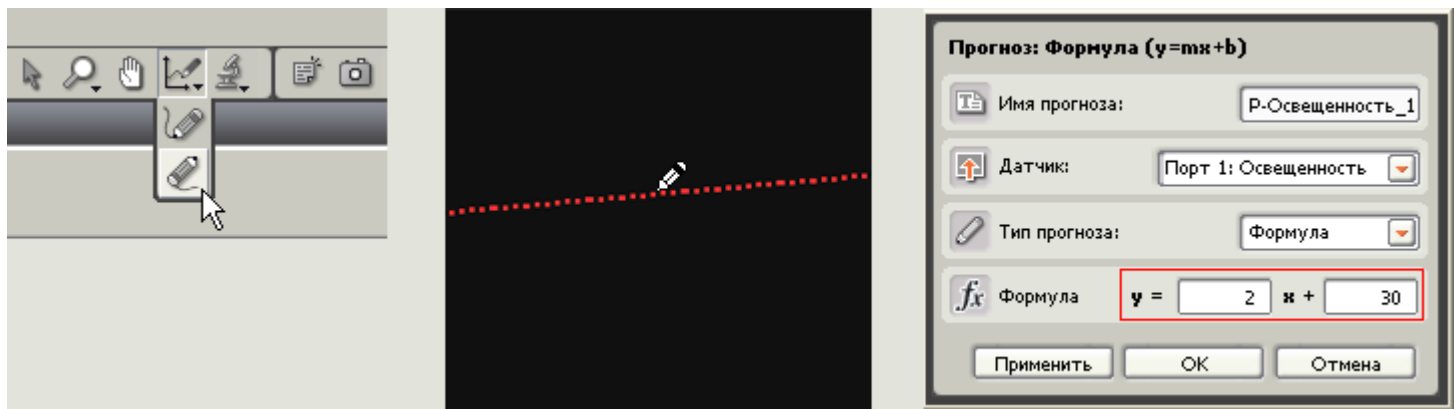
1. Наклон: это значение определяет наклон линии прогноза
2. Точка пересечения с осью Y: это значение определяет координату Y точки, в которой график пересекает ось Y.
3. Применить: предварительный просмотр формулы перед нажатием на кнопку ОК.

## Редактирование прогноза, созданного методом "карандаш"



Выберите средство редактирования прогнозов и щелкните по прогнозу, который необходимо изменить. После этого можно перемещать линию прогноза или привязывать ее к точкам, указанным на графике щелчками мышь.

## Редактирование прогноза, созданного методом "формула"



Прогнозы, созданные методом "формула", можно отредактировать при помощи соответствующей панели. Для того чтобы открыть эту панель, перейдите в режим редактирования прогнозов и щелкните по нужной линии на графике. Для того чтобы отредактировать прогноз, созданный методом "формула", выберите пункт меню Средство Редактирования и щелкните по прогнозу, который необходимо изменить. Для того чтобы открыть диалоговое окно Средства создания прогнозов, щелкните по линии прогноза на графике.

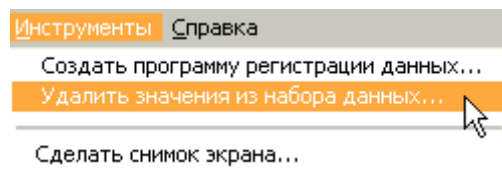


Совет: В каких случаях пользоваться функцией Удаление значений из набора данных?

Функция Удаление значений из набора данных может быть полезна, если необходимо удалить пики, которые иногда встречаются, если ультразвуковой сигнал не был отражен и не достиг датчика. Это может произойти по ряду причин (объект находится вне диапазона действия датчика, объект изготовлен из текстильного материала или имеет неровную поверхность, объект слишком мал, чтобы отразить сигнал). Так как некоторые поверхности неэффективно отражают звуковые волны, исходящие от ультразвукового датчика, последний считает, что они находятся "вне диапазона" и автоматически выводит на график значение 255 см (100 дюймов). При помощи функции Удаление значений из набора данных можно очистить результаты, удалив такие пики до анализа набора данных.

## УДАЛЕНИЕ ЗНАЧЕНИЙ

Команда «Удалить значения из набора данных» используется для удаления определенного диапазона значений из выбранного прогноза или набора данных. Эта команда, кроме того, предназначена для удаления выпадающих значений на графике, которые иногда регистрируются датчиками.



Для того чтобы удалить значения из того или иного прогноза или набора данных, выберите этот прогноз или набор данных щелчком по соответствующей линии на графике, либо в таблице данных. После этого выберите команду «Удаление значений из набора данных» в меню Инструменты.



Введите наибольшее и наименьшее значения, которые должны быть удалены. Все значения диапазона, включая указанные в полях ввода 1 и 2, будут удалены из выбранного прогноза или набора данных. Например, если в поле 1 ввести число 50, а в поле 2 - 100, будут удалены не только все значения, лежащие в пределах от 50 до 100, но и равные 50 и 100.

После нажатия ОК график и таблица данных будут автоматически обновлены в соответствии с произведенными изменениями.

 **Совет: в каких случаях следует использовать функцию «Удалить значения из набора данных» на панели инструментов?**

Функция «Удалить значения из набора данных» может быть полезна, если необходимо удалить выпадающие значения, которые иногда регистрируются, если ультразвуковой сигнал не был отражен и не попал в датчик. Это может произойти по ряду причин (объект находится вне диапазона действия датчика, объект изготовлен из текстильного материала или имеет неровную поверхность, объект слишком мал, чтобы отразить сигнал). Поскольку некоторые поверхности плохо отражают ультразвуковые волны, испускаемые датчиком расстояния, последний считает, что они находятся "вне диапазона" и автоматически выводит на график максимальное значение 255 см (100 дюймов). Используя функцию «Удалить значения из набора данных», можно очистить результаты, удалив такие выпадающие значения до анализа набора данных.