



# Младший брат LabVIEW

Рудниченко Д.С., г.Одесса

На первый взгляд, непонятно, зачем 10-ти летнему ребенку ломать голову над программированием во время игры? Можно ведь подарить простой конструктор или что-нибудь в этом роде, и Ваше чадо будет счастливо, самостоятельно развлекаая себя, конструируя нечто. Но как долго это продлится? Неделя, может, две. Детям во время игры нужен постоянный интерес, который, непременно, помогает в познании чего-то нового.

**Ж**ивое LEGO - так можно назвать конструктор Mindstorms всемирно известной датской фирмы. Живое потому, что в состав Mindstorms входят миниатюрный компьютер NXT, серводвигатели и датчики. Но прежде чем электронное чудо из забавных кирпичиков начнет двигаться, ощущать, осязать и распознавать, его создатель должен написать программу. Как это сделать ребенку, или даже взрослому, которому ранее не приходилось даже составлять алгоритм? Конечно, найдутся вундеркинды, которые без проблем не то чтобы на C - на ассемблере "свяжут" необходимый софт. А как же остальные миллионы людей, желающие творить и экспериментировать.

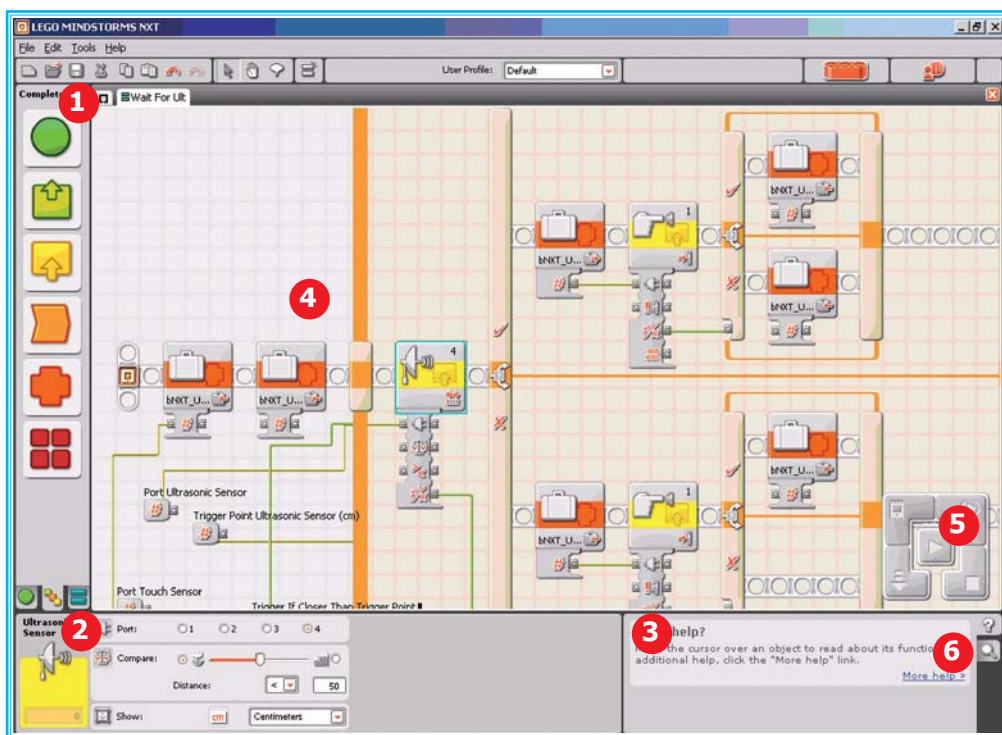
Не видать Mindstorms такой популярности среди людей разных возрастов и профессий, а может и вообще бы инициатива энтузиастов из Массачусетского Технологического Института осталась бы пустой затеей, если бы основу LEGO® MINDSTORMS® Software не составила платформа самого популярного программного продукта в мире - среда графического программирования NI LabVIEW™! Созданная совместными усилиями двух компаний, мировых лидеров в своих областях - LEGO Group и National Instruments, среда разработки приложений для микрокомпьютера NXT имеет интуитивно понятный интерфейс "drag&drop" и позволяет решать задачи программирования простым

перемещением иконок на экране монитора и их конфигурированием. По сути своей это младший брат уважаемого LabVIEW.

Будучи чрезвычайно простым в использовании, LEGO® MINDSTORMS® Software, тем не менее, предоставляет начинающему программисту практически неограниченную массу возможностей. И стар, и млад, придут в восторг, когда напишут свою первую программу, загрузят ее в NXT и механическая конструкция оживет. Всего за несколько часов можно собрать и запрограммировать интеллектуальную игрушку, адекватно реагирующую на входные сигналы и послушно исполняющую команды. Пожалуй, предложить, как еще полезнее провести время за игрой, сложно.

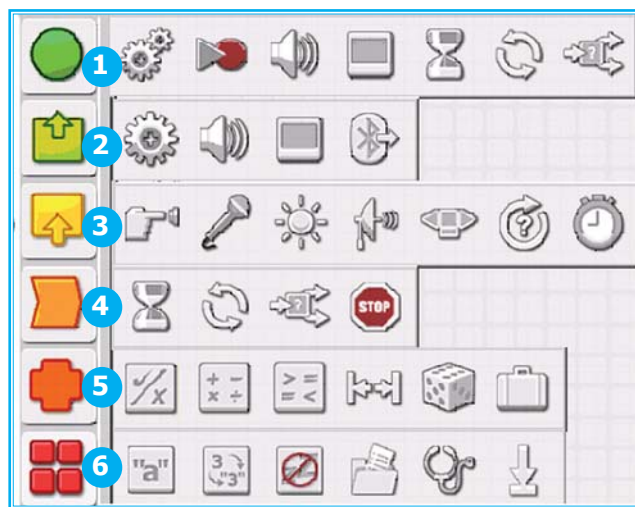
А далее у Вас появляется новый друг, который надолго увлечет Вас в мир современных технологий и даст мощный толчок для развития фантазии в реальном, а не в виртуальном мире. Вот почему есть смысл поближе познакомиться с необычным LabVIEW.

В верхней части главного окна программы расположены панели "Меню" и "Управление". Это стандартные панели, используемые в различных компьютерных приложениях, здесь все как везде. С левой стороны находятся панели блоков программирования, назовем их палитрой программирования (1). Вклад-

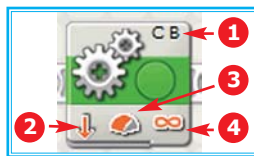
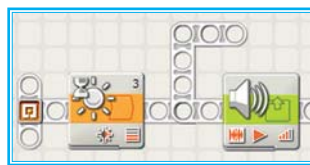


ки внутри палитры позволяют переключаться между общей палитрой, содержащей часто используемые блоки, полной палитрой и палитрой, которую пользователь загружает или создает. Внизу находится панель конфигурации (2) для настройки блоков программирования и окно помощи (3). В центре - пространство для листинга программы - рабочая область (4). Сюда из палитры программирования перетаскиваются программные блоки и устанавливаются на луч последовательности. В правой нижней части рабочей области находится блок загрузки программы в NXT-контроллер (5). Под блоком загрузки расположены окно помощи и инструмент масштабирования (6). Вот, пожалуй, и все основные компоненты программы. Есть, правда, еще и инструментарий для калибровки датчиков, обновления прошивки NXT-контроллера, загрузки программ в несколько контроллеров, и канал доступа к ресурсам по конструированию на основе Mindstorms (если компьютер подключен к всемирной паутине), и подробные инструкции по сборке интересных моделей, и редактор блоков и иконок пользователя и еще много всякого полезного. Но главное перечислено выше.

Все блоки (полная палитра), используемые для написания программы, кроме пользовательских, разделены на шесть групп - основные блоки (1), блоки действия (2), блоки работы с датчиками (3), блоки управления выполнением программы (4), блоки обработки данных (5), специальные блоки (6):



Начало всех программ - иконка, из которой формируются направляющие лучи в виде стилизованных балок серии LEGO Technic. Лучи, если это необходимо, можно дополнительно создавать в любом



месте программы: Первый блок основной группы - **Move** (блок движения). Рассмотрим подробно его структуру и станет понятна идеология графического программирования для начинающих. Блок **Move** позволяет управлять несколькими моторами одновременно. Что конкретно он будет выполнять, отображено на самом блоке: 1 - активизируемые порты, 2 - направление вращения, 3 - уровень сигнала (мощность), подаваемого на двигателя (0..100%), 4 - длительность вращения. Настройка этого блока выполняется с помощью панели конфигурации, которая откроется автоматически, как только блок будет помещен на рабочую область ?

В особых комментариях эта панель вроде, как и не нуждается, и тем не менее:



5 - выбор моторов, которыми необходимо управлять. Если выбраны порты **B** и **C**, то работа их будет синхронизирована, а если все три - **A**, **B** и **C** - то синхронизированы только **B** и **C**;

6 - выбор направления вращения (вперед, назад, стоп);

7 - синхронное управление двумя моторами для организации режима поворотов;

8 - установка уровня мощности;

9 - установка длительности вращения (без ограничения по времени или в течение заданного числа оборотов, секунд или градусов);

10 - установка режима работы выбранных портов при переходе к следующему программному блоку (останов или продолжение вращения);

11 - индикатор состояния датчиков угловых перемещений моторов NXT.

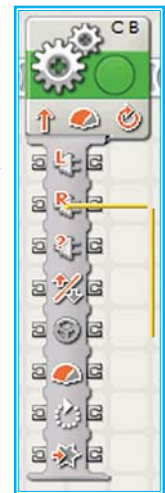
Итак, установил блок на луч, сконфигурировал его - переходи к следующему. Какое же это программирование? Однако не надо делать поспешных выводов, лучше щелкнуть левой клавишей манипулятора на



вкладке, расположенной в нижней левой части блока. Откроется концентратор данных,

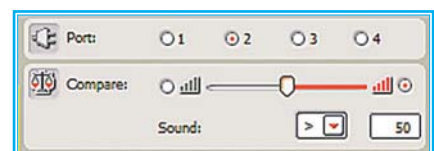
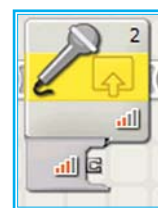
который позволит переменные блока устанавливать динамически. Вот это уже будет посерьезнее.

Концентратор данных выполнен в виде разъема с поименованными контактами: слева - входы, справа - выходы. Соединение концентраторов различных блоков выполняется проводниками, причем для различных типов данных используются провод-



ники разных цветов: зеленый - логические данные, желтый - числовые, и оранжевый - текстовые. При некорректном соединении проводник превращается в серую пунктирную линию. О причине такого преобразования можно узнать в окне ошибки. Ну, почти как в LabVIEW. Неправда ли? Даже вид курсора - катушка.

Большинство программных блоков имеет скрытый концентратор данных. Однако есть блоки, применение которых требует обязательного использования концентратора. При установке таких блоков на рабочую область, концентратор открывается автоматически. В качестве примера такого блока рассмотрим программный блок акустического датчика. Выбрав в группе блоков для ра-

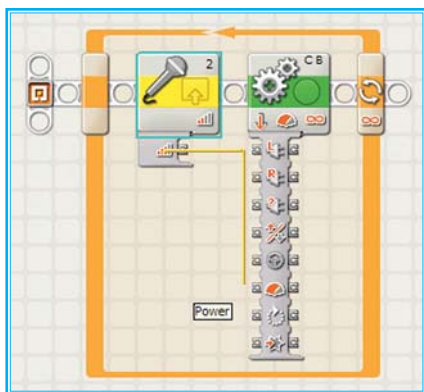




боты с датчиками изображение микрофона, устанавливаем блок на рабочую область - автоматически открывается выход уровня акустического сигнала, воспринимаемого датчиком.

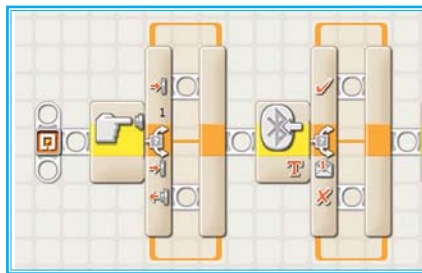
Если внимательно посмотреть на панель конфигурации этого блока, то нетрудно догадаться, что концентратор данных раскрылся не полностью. Есть там еще входы/выходы, которые позволяют, например, организовать компаратор и сформировать логический сигнал превышения заданного уровня акустического шума. Функция небезполезна.

Из рассмотренных двух блоков уже можно построить программку, которая в зависимости от уровня шума быстрее или медленнее вращала бы двигатель. Но настоящий программист справедливо заметит, что для полноценной программы нужно создать цикл. И такое в арсенале программных блоков тоже найдется. Этот блок, именуемый **Loop**, а по-нашему цикл, находится в группе блоков управления выполнения программы. Причем с помощью панели конфигурации можно задать бесконечный цикл или цикл с выходом по условию. А условие - это время, количество выполненных циклов, состояние заданного датчика или просто состояние логического сигнала на входе концентратора данных. Тогда задуманная программка будет выглядеть вот так:



Для ветвления алгоритма используется блок переключения **Switch**. Условием переключения между двумя последовательностями программы могут быть: состояние таймера, датчика, клавиатуры NXT-контроллера, сообще-

ние, принятое по каналу Bluetooth или логический, числовой и текстовый сигнал на управляющем входе.



А для организации циклов ожидания по времени, состоянию датчиков, в т.ч. и датчика угловых перемещений, приходится сообщать о сообщении или нажатия/отпуска-

ния кнопок на модуле контроллера, предусмотрены соответствующие конфигурируемые блоки

**Wait:**

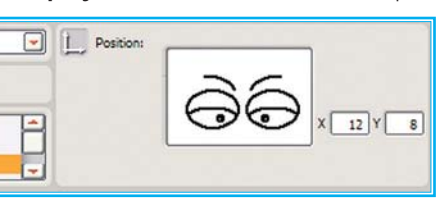


Ну, а как обстоят дела с подпрограммами? Конечно, и такая возможность имеется, только в LEGO® MINDSTORMS® Software это называется "Создайте свой блок". Отмечается группа блоков, лучей и связей, присваивается имя, если необходимо - вводится комментарий, и компоуется из готовых рисунков иконок подпрограммы. И все, готово. Проще не бывает, даже в LabVIEW.

А вот то, что вряд ли присутствует в программном обеспечении для взрослых, так это блок записи/воспроизведения. Он позволяет зафиксировать действия, выполненные моторами, а затем воспроизвести их. В ситуациях, когда трудно описать сложное движение, например, манипулятора, применение этого блока решает проблему. Т.е. моторы вращаются в ручную, а контроллер N X T

фиксирует показания датчиков угловых перемещений в течение заданного интервала времени в файл. При последующем воспроизведении этого файла в "боевой" программе, моторы будут вращаться так, как это было в режиме записи, но уже под управлением контроллера.

Наличие LCD-экрана (100x64), мини-клавиатуры и динамика, естественно, предполагает наличие соответствующих программных блоков для работы с ними. Блок **Display** позволяет вывести на экран

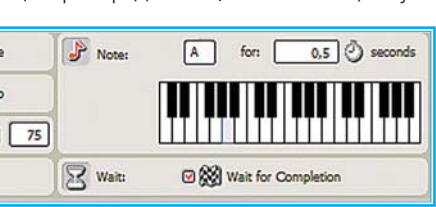


стандартное изображение, текст, или с помощью простейшего встроенного редактора (точка, линия окружность) создать собственный рисунок. Для этого можно использовать как панель конфигурации, так и концентратор данных. Выстраивая последовательность таких блоков, чередуя их с блоками задержки, можно получить "сложные" анимации.

Блок кнопок **NXT** обрабатывает состояние кнопок указателей влево/вправо и ввода. Причем панель конфигурации позволяет установить характер действия: нажатие, от-

пускание или нажатие/отпускание. Результатом работы такого блока будет активизация сигналов на выходных контактах концентратора.

По-своему интересен и блок **Sound**. С его помощью создаются музыкальные фрагменты и даже воспроизводятся речевые сообщения на любом языке. Этот блок также имеет концентратор данных, позволяющий уп-



равлять процессом воспроизведения (начать, окончить, циклический режим, ожидание окончания, уровень громкости).



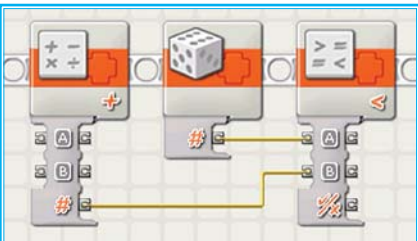
Три таймера NXT-контроллера могут быть использованы для точного формирования временных интервалов. Программный блок, с помощью которого выполняется сброс таймера, чтение его текущих показаний (в единицах мс) и сравнение с уставкой, так и называется - **Timer**.

Занятым может получиться взаимодействие нескольких контроллеров при использовании интерфейсного канала Bluetooth. Программная поддержка беспроводного канала реализуется блоком



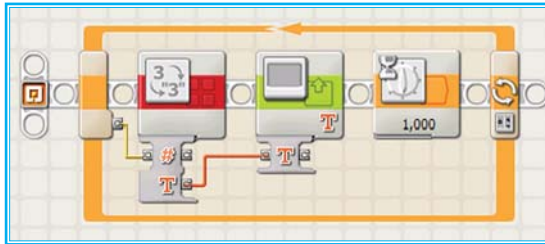
**Receive Message**. Каждый контроллер NXT имеет десять почтовых ящиков, куда и помещаются текстовые, числовые или логические данные. Кроме того, этот блок содержит еще и модуль сравнения данных, и его концентратор имеет 9 контактов для подключения источников/приемников сообщений различного типа и выводов результатов сравнения.

Группа блоков обработки данных содержит шесть "кубиков": блоки логических и математических опера-



ций (AND, OR, XOR, NOT и сложение, вычитание, умножение, деление), блоки сравнения (больше, меньше, равно и нахождение числа внутри заданного интервала), блок генератора случайных чисел и блок переменной. Кто-то скажет "Малова-то функций", а кто-то - "И зачем такие сложности?". Пусть будет! Наверняка, многим пригодится.

Будут востребованы и такие блоки как **Text** и **Number to Text**. Первый из названных "суммирует" три сообщения для вывода результата на экран, а второй - преобразует числовые данные в символы текста. Например, при выполнении такой программы на экране каждую секунду будет отображаться показания счетчика циклов:



Примером программного блока без панели конфигурации является блок **Keep Alive**. Этот блок следует использовать в случае, если по замыслу автора программы контроллер NXT длительное время должен находиться в состоянии ожидания. Иначе он впадет в спячку. А **Keep Alive** не позволит ему это сделать в течение некоторого времени, задаваемого на единственном контакте коммуникатора (0.. 4294967296 мс).



И, в заключение, нельзя не сказать и о функциях доступа к файлу



И, в заключение, нельзя не сказать и о функциях доступа к файлу NXT. Запись, чтение, удаление реализованы в блоке **File Access**. Каждый раз, записывая и читая файл, а это может быть текст или цифровые данные, необходимо выполнять процедуру закрытия файла. Данные всегда записываются в конец файла и если необходимо перезаписать файл, то текущий следует

просто удалить. Размер файла может составлять 32767 байт.

Если ничего не упущено, то тогда о LEGO® Mindstorms® Software сказано все. На первый взгляд, это просто отвечающая духу времени отличная интеллектуальная игрушка. Но если копнуть поглубже, то все намного серьезнее. Разрабатывая, программируя и тестируя системы на основе NXT-контроллеров, приобретаются навыки в области и конструирования, и программирования, осваиваются алгоритмы пошагового решения задач, выработки и проверки гипотез, анализируются неожиданные результаты. Освоить этот программный продукт, даже при отсутствии помощи со стороны педагога-специалиста, ребенок способен без проблем начиная возраста 10+. Что уж говорить о взрослых.

А если хотите расширить возможности платформы Mindstorms® NXT - сделайте это в LabVIEW!

**КОНТАКТЫ:**  
 тел: 8-050-333-06-75  
 8-095-828-12-12  
 e-mail: niokr@breezein.net

