



МОСКОВСКИЙ АВТОМОБИЛЬНО-ДОРОЖНЫЙ
ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
(МАДИ)

«Автоматизированные системы управления»

Николаев А.Б., Васюгова С.А.

Программирование роботов-манипуляторов

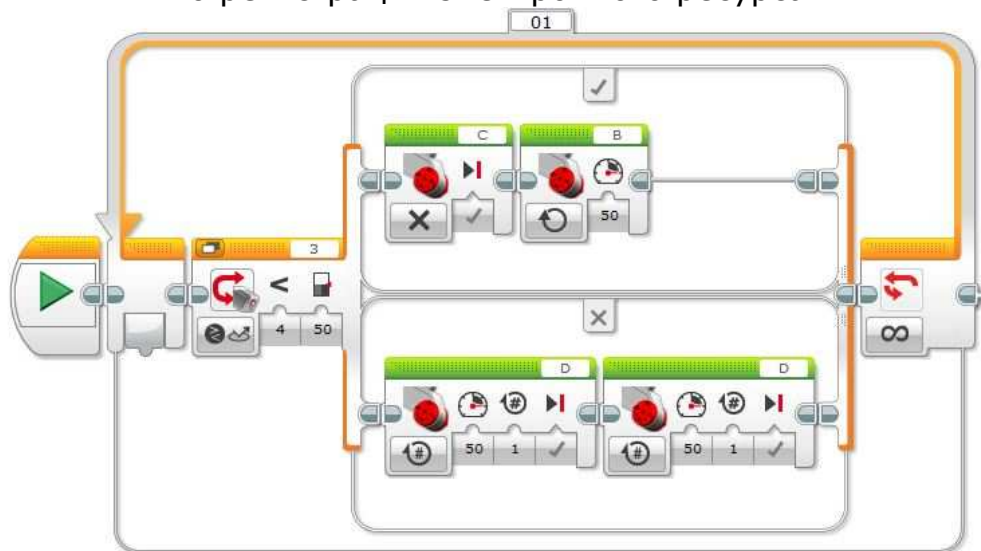
Методические указания

к лабораторным работам по дисциплине

«Интеллектуальные системы»

по направлениям подготовки 09.03.02 – «Информационные системы и технологии» 09.04.01 – «Информатика и вычислительная техника»

Свидетельство
о регистрации электронного ресурса



Москва 2015

УДК 681.3.06

ББК 32.973.26-018.2

Н63

Николаев А.Б.

Программирование роботов-манипуляторов: методические указания к лабораторным работам / А.Б. Николаев, С.А. Васюгова. – Красноярск: Научно-инновационный центр, 2015. – 94 с.

ISBN 978-5-906314-32-1

DOI: 10.12731/asu.madi.ru/IS.2015.94

Методические указания предназначены для студентов, изучающих дисциплину «Интеллектуальные системы» по направлениям подготовки 09.03.02 – «Информационные системы и технологии» и 09.04.01 – «Информатика и вычислительная техника».

Методические указания содержат материалы, необходимые для выполнения лабораторных работ и представляют интерес для широкой аудитории, осваивающей приемы программирования роботов с использованием современной инструментальной среды LEGO MINDSTORMS Education EV3. Знания и навыки, приобретенные при изучении дисциплин, могут быть полезны при работе над курсовыми работами и дипломными проектами.

© [Авторы](#), 2015

© [МАДИ](#), 2015

© [Научно-инновационный центр](#), 2015

Введение

Методические указания разработаны по основной профессиональной образовательной программе в соответствии с требованиями: Федерального государственного образовательного стандарта высшего профессионального образования по направлению подготовки 09.03.02 – «Информационные системы и технологии»; Федерального государственного образовательного стандарта высшего профессионального образования по направлению подготовки 09.04.01 – «Информатика и вычислительная техника».

Материал, изложенный в методических указаниях, позволяет освоить фундаментальные основы программирования роботизированных устройств, а также, получить навыки проектирования и конструирования базовых роботов.

На базе конструктора LEGO MINDSTORMS Education и контроллера EV3 студенты смогут научиться создавать роботизированные системы. При помощи программы LEGO MINDSTORMS Education EV3 смогут создавать различного рода программы, которые будут тестировать и проверять на разработанных роботах.

Для освоения дисциплины студенту необходимо обладать базовыми навыками программирования на простейших языках программирования.

Лабораторные работы проводятся в такой последовательности, чтобы студенты смогли постепенно освоить курс от простых программ до программ средней сложности и выше. В методических указаниях приведены 7 лабораторных работ.

[Лабораторная работа №1](#) содержит общие сведения о ПО и работе. Включает в себя сборку базовой модели робота и установку ПО.

[Лабораторная работа №2](#) содержит курс основ программирования контроллера EV3.

[Лабораторная работа №3](#) содержит курс программирования робота на выполнение простых задач.

[Лабораторная работа №4](#) посвящена шинам данных и научит регистрировать данные, собранные внешними узлами робота.

[Лабораторная работа №5](#) научит использовать блок «Переключатель» и запрограммировать робот при помощи графиков.

[Лабораторная работа № 6](#) позволит разобраться в конструировании более сложных роботов, на основе сборки робота-манипулятора.

[Лабораторная работа № 7](#) позволит узнать принцип программирования более сложных роботов, на примере создания многозадачной программы для робота-манипулятора.

Лабораторная работа №1

Знакомство со средой разработки LEGO MINDSTORMS Education EV3. Конструирование базовой модели робота

Краткие теоретические сведения

Робот - автоматическое устройство, созданное по принципу живого организма. Действуя по заранее заложенной программе и получая информацию о внешнем мире от датчиков(аналогов органов чувств живых организмов), робот самостоятельно осуществляет производственные и иные операции, обычно выполняемые человеком (либо животными). При этом робот может иметь связь как с оператором (получать от него команды), так и действовать автономно.

Контроллер - это устройство управления и контроля процессами системы, в которой он установлен. Контроллер преобразует код в управляющие сигналы и выдает на внешние устройства. С внешних устройств он получает данные о рабочих процессах и условиях окружающей среды, с помощью чего способен самостоятельно контролировать некоторые действия системы.

Модуль EV3- это программируемый интеллектуальный контроллер, который контролирует и управляет датчиками и моторами.

Технические характеристики модуля EV3:

- Процессор — ARM9;
- FLASH память — 16 мегабайт;
- Оперативная память — 64 мегабайт;
- Операционная система — Linux;
- Слот расширения SD;
- USB 2.0 (поддерживает USB Host, то есть можно подключить к WiFi);
- Bluetooth 2.1;
- 4 порта на вход и 4 порта на выход;
- Динамик.

Программирование на контроллере EV3 позволит легко и быстро освоить этапы составления программ и тестов для робота благодаря простому интерфейсу и готовым программным блокам (компонентам).

Большой мотор- это мощный мотор, который имеет встроенный датчик вращения с разрешением 1 градус для точного контроля.

Датчик цвета - это цифровой датчик, который может определять цвет или яркость света, поступающие в небольшое окошко на лицевой части датчика.

Гироскопический датчик- это цифровой датчик, который обнаруживает вращательное движение вокруг одной оси.

Датчик касания - это аналоговый датчик, который может определять, когда красная кнопка нажата, а когда отпущена.

Ультразвуковой датчик- это цифровой датчик, который определяет расстояние до находящегося перед ним объекта.

Задание на лабораторную работу:

1. **Цель** в ознакомлении со средой разработки LEGO MINDSTORMS Education EV3 и модулем EV3.
2. Рассмотреть и изучить датчики и контроллеры.
3. После знакомства с предметом собрать базовую модель робота.
4. После сборки протестировать встроенную программу.
5. Установить ПО LEGO MINDSTORMS Education EV3.

Порядок выполнения лабораторной работы:

1. Для знакомства с роботом EV3 и его программной средой разработки LEGO MINDSTORMS Education EV3, открыть PDF-файл «Руководство пользователя» и внимательно прочитать его.

2. С использованием «Руководства пользователя» студенты изучают датчики и контроллеры.

3. Студенты разбиваются на группы по четыре человека и выполняют сборку робота. Базовая модель робота содержит 4 основных датчика: ультразвуковой датчик, датчик касания, датчик цвета и гироскоп. Передвижение робота будет осуществляться при помощи двух больших моторов. Руководство по сборке представлено в распечатанном пособии по сборке Lego Mindstorms Education.

4. Для проверки правильности собранного робота мы запустим встроенную программу, которая сохранена на контроллере. Для этого включаем контроллер зажатием центральной кнопки. После включения высвечивается основное меню контроллера, изображенное на Рис. 1.

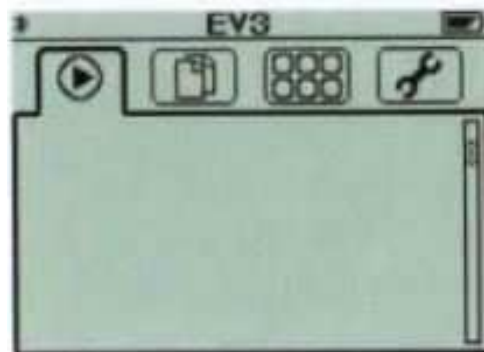


Рис. 1. Основное меню котроллера

Заходим в сохраненные документы Рис. 2.



Рис. 2. Сохраненные документы

Открываем папку **BrkProg_SAVE** и открываем файл **Demo** Рис. 3 и программа запускается.



Рис. 3. Сохраненная программа

После чего робот начинает движение. Если робот собран правильно и все датчики с моторами подключены верно, робот выполнит действие типа: проезд прямо-разворот-проезд прямо-остановка.

5. Для установки ПО LEGO MINDSTORMS Education EV3 открываем папку **Lego** и запускаем файл **LME-EV3-WIN32-RU-01-01-full-setup**. После установки запускаем файл **LME-EV3-DEP-WIN32-RU-01-00-full-setup**. Перезагружаем компьютер. После чего программа успешно установлена.

Контрольные вопросы

1. Что такое робот?
2. Что такое контроллер?
3. Что такое модуль EV3?
4. Способы работы с модулем EV3?
5. Какие основные датчики используются в базовой модели?
6. В какие порты подключаются моторы, а в какие датчики в модуле EV3?
7. Что такое датчик цвета?
8. Что такое ультразвуковой датчик?
9. Что такое гироскопический датчик?

Лабораторная работа №2

Программирование робота с использованием контроллера

Краткие теоретические сведения

Большой мотор- это мощный мотор, который имеет встроенный датчик вращения с разрешением 1 градус для точного контроля.

Датчик цвета - это цифровой датчик, который может определять цвет или яркость света, поступающие в небольшое окошко на лицевой части датчика.

Гироскопический датчик - это цифровой датчик, который обнаруживает вращательное движение вокруг одной оси.

Датчик касания - это аналоговый датчик, который может определять, когда красная кнопка нажата, а когда отпущена.

Ультразвуковой датчик- это цифровой датчик, который определяет расстояние до находящегося перед ним объекта.

Данный урок научит программировать робота, используя контроллер. При помощи составления простых программ управления роботом, можно получить базовые знания и навыки работы с роботом. Программирование на контроллере сегодня является одним из необходимых умений в работе с роботизированными системами, т.к. почти все современные контроллеры имеют функцию внутреннего программирования без подключения к компьютеру оператора/разработчика. Программирование на контроллере позволяет быстро создавать управляющие программы, устранять недостатки и сбои системы, а также, собирать основную информацию непосредственно с самого контроллера. Это позволяет уменьшить время реакции на поломку, повысить производительность и качество рабочих процессов.

Задание №1. Движение по заданной траектории

Цель данного примера в задании роботу условия движения. Перемещается робот при помощи двух больших сервомоторов. Управление моторами производится при помощи контроллера, к которому они подсоединены при помощи кабелей в порты выхода В и С. При помощи такого примера легко освоить принцип создания алгоритмов по управлению движением робота.

Общую программу студенты выполняют в группах по два человека на один робот. Самостоятельная работа выполняется каждым индивидуально.

Программа. Движение робота по заданной траектории

задать роботу траекторию движения в виде вращения на 360° в течение 2-ух секунд.

Выполнение.

Прежде чем приступить к программированию, необходимо включить контроллер, нажав центральную кнопку на панели Рис. 1.



Рис. 1. Контроллер


Далее заходим в «Приложения модулем» , и выбираем **Brick Program** Рис. 2.



Рис. 2. Окно управления контроллером

После чего открывается главное окно нашей будущей программы Рис. 3. Любая начальная программа уже имеет два блока: блок «Начало» программы и блок «Цикл».

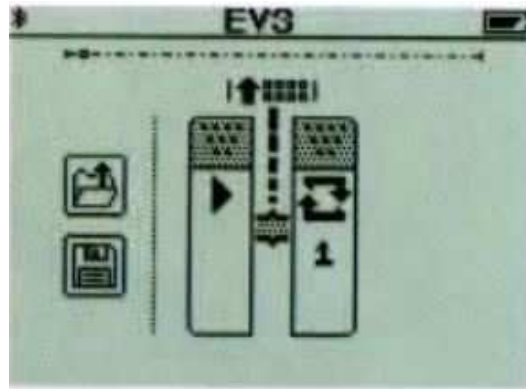


Рис. 3. Окно запуска

Все необходимые программные блоки находятся во всплывающем окне, при нажатии на верхнюю кнопку панели контроллера Рис. 4.

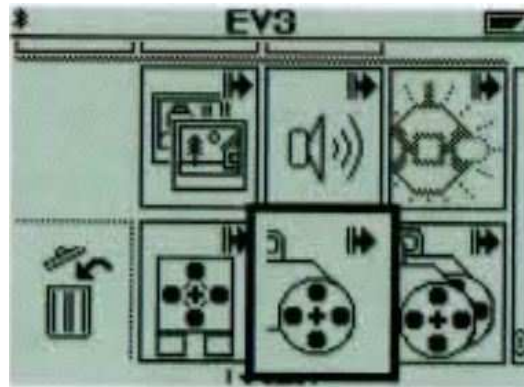


Рис. 4. Меню программных блоков

Для нашей задачи нам необходимо выбрать блок работы двух больших моторов Рис. 5.

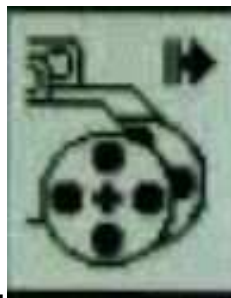


Рис. 5. Блок работы двух больших моторов

После чего он появляется в программе между двух основных блоков Рис. 6.

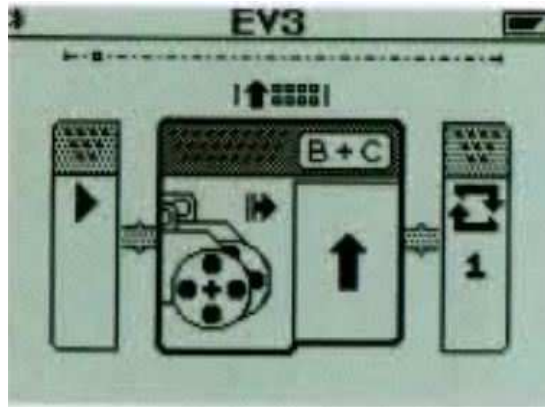


Рис. 6. Окно программы

Нажав на центральную кнопку панели контроллера, мы выбираем конфигурации данного блока: направление движения и порты вывода, к которым подсоединены моторы. В нашем случае направление движения будет круговым, а порты вывода В и С Рис. 7.

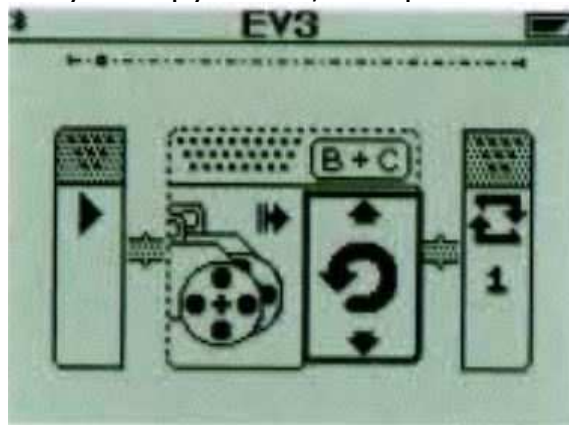


Рис. 7. Установка кругового движения робота

Выходим из настроек блока движения больших моторов нажатием на правую кнопку панели контроллера.

Далее необходимо задать время, на протяжении которого будет происходить движение робота. Заходим в меню программных блоков нажатием на верхнюю кнопку на панели контроллера. Выбираем блок «Время» Рис. 8.

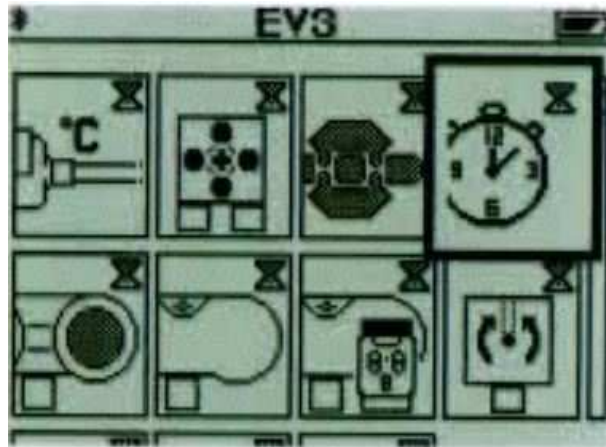


Рис. 8. Блок «Время» в меню перечня программных блоков

После чего он появляется в нашей программе Рис. 9. Заходим в настройки блока «Время» нажатием на среднюю кнопку. По умолчанию стоит время 02.00 сек., что необходимо для нашей задачи. При желании время можно менять кнопками панели контроллера.

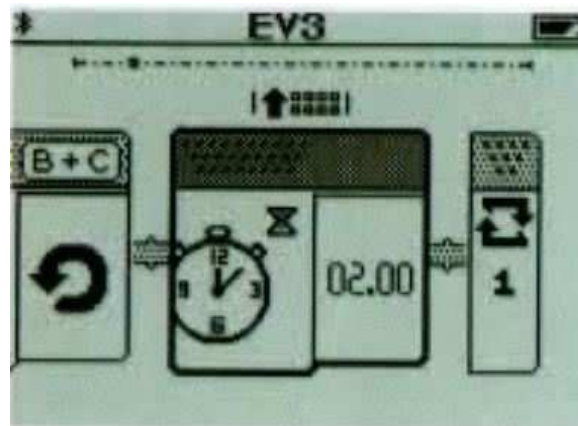


Рис. 9. Установка значений блока «Время»

Наша программа готова. Возвращаемся в начало программы и выбираем блок запуска Рис. 10.

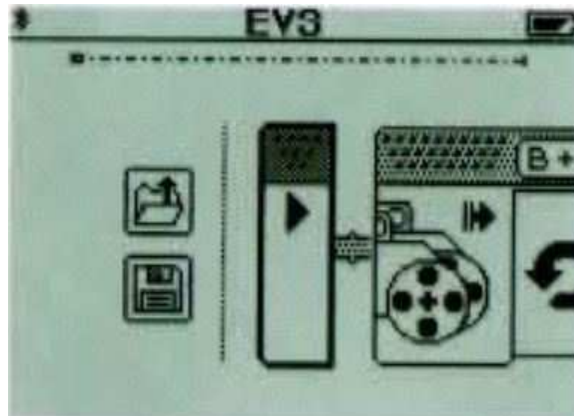


Рис. 10. Блок запуска программы

Робот выполняет все заданные условия.

Варианты для самостоятельной работы

Вариант №1.

Задать роботу движение типа: вперед 1 сек.-поворот направо 1 сек.-остановка.

Вариант №2.

Задать роботу движение типа: вперед 2 сек.-разворот 1 сек. –назад 1сек.

Вариант №3.

Задать роботу движение типа: вперед 1 сек.-разворот 1 сек.-назад 1 сек.-разворот 1 сек.

Вариант №4.

Задать роботу движение типа: назад 1 сек.-поворот налево 1 сек.-вперед 2 сек.

Вариант №5.

Задать роботу движение типа: разворот 1 сек.-вперед 2 сек.-поворот направо 1 сек.

Вариант №6.

Задать роботу движение типа: вперед 2 сек.-разворот 1 сек.-назад 1 сек.

Задание №2. Работа с ультразвуковым датчиком

Цель заключается в освоении этапа программирования ультразвукового датчика робота. Ультразвуковой датчик позволяет определять расположение предметов в пространстве и измерять расстояния до них. **Ультразвуковой датчик** расстояния определяет расстояние до объекта, измеряя время отображения звуковой волны от объекта.

Общую программу студенты выполняют в группах по два человека на один робот. Самостоятельная работа выполняется каждым индивидуально.

Программа. Добавление ультразвукового датчика и работа с ним

Робот двигается в прямом направлении до тех пор, пока на пути не возникнет препятствие в виде любого предмета, в нашем случае это цветной квадрат. Робот должен остановиться за 30 см. до этого предмета.

Выполнение.

Программа будет состоять из нескольких блоков. Для начала заходим в «Приложения модуля»  и выбираем Brick Program.

Создаем новую программу. В программе заходим в меню перечня программных блоков нажатием на верхнюю кнопку на панели контроллера. Выбираем блок управления движением двух больших моторов. В настройке блока движение выбираем в прямом направлении. Порты вывода В и С Рис. 11.

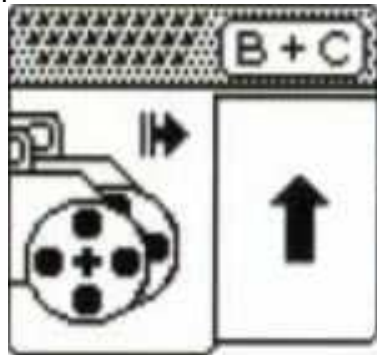


Рис. 11. Блок моторов

Далее добавляем блок ультразвукового датчика. В настройках блока указываем расстояние до предмета, которое необходимо для нашего условия. Оно составляет 30 см. Порт входа №4. Настройки блока представлены на Рис. 12.

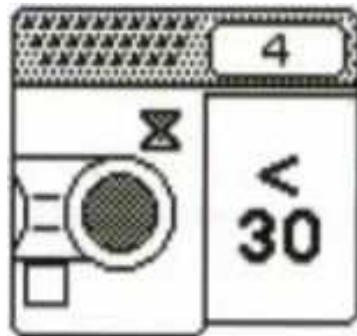


Рис. 12. Блок ультразвукового датчика

Далее, для того, чтобы наш робот остановился, необходимо добавить еще один блок для управления большими моторами. В настройке блока в направлении движения выбираем знак остановки Рис. 13. Порты выхода В и С.

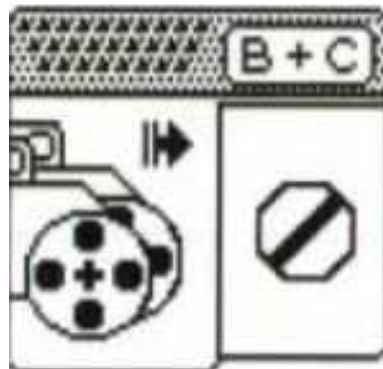


Рис. 13. Блок моторов

Общий вид программы представлен на Рис. 14.

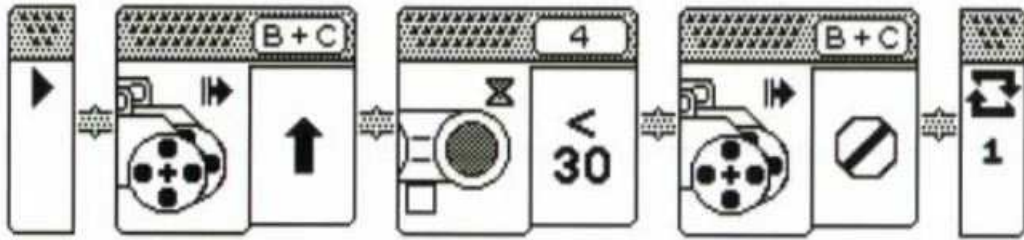


Рис. 14. Общий вид программы

Варианты для самостоятельной работы

Вариант №1.

Задать роботу движение типа: вперед, пока не обнаружит предмет на расстоянии до 30 см.-остановится-поворот направо 1 сек.

Вариант №2.

Задать роботу движение типа: разворот 1 сек.-вперед, пока не обнаружит предмет на расстоянии до 30 см.-остановка.

Вариант №3.

Задать роботу движение типа: вперед, пока не обнаружит предмет на расстоянии до 30 см.-остановка-поворот налево 1 сек.

Вариант №4.

Задать роботу движение типа: вперед, пока не обнаружит предмет на расстоянии до 30 см.-остановка-назад 2 сек.

Вариант №5.

Задать роботу движение типа: вперед 1 сек.-назад 1 сек.-вперед, пока не обнаружит предмет на расстоянии до 30 см.-остановка.

Вариант №6.

Задать роботу движение типа: вперед, пока не обнаружит предмет на расстоянии до 10 см.-остановка-разворот 1 сек.-назад 1 сек.

Задание №3. Применение гироскопического датчика в движении робота


Цель заключается в разработке программы, которая поможет легко и быстро освоить и запрограммировать гироскоп, установленный на корпусе робота. Гироскоп - устройство, способное реагировать на изменение угловой ориентации тела, на котором оно установлено, относительно инерциальной системы отсчета.

Общую программу студенты выполняют в группах по два человека на один робот. Самостоятельная работа выполняется каждым индивидуально.

Программа. Добавление гироскопического датчика и его программирование

Робот двигается в прямом направлении 1 сек., поворачивается на 45° и двигается еще 1 сек. в прямом направлении.

Выполнение.

Заходим в «Управление контроллером»  и выбираем Brick Program.

Создаем новую программу. В программе заходим в меню программных блоков нажатием на верхнюю кнопку на панели контроллера. Выбираем блок управления движением двух больших моторов. В настройке блока движение выбираем в прямом направлении. Добавляем блок «Время» и задаем значение в 1 секунду.

Далее выбираем блок управления движением двух больших моторов и выбираем позицию разворота в качестве траектории движения.

Добавляем блок «Гироскоп» и угол поворота ставим 45° , как указано на Рис. 15.

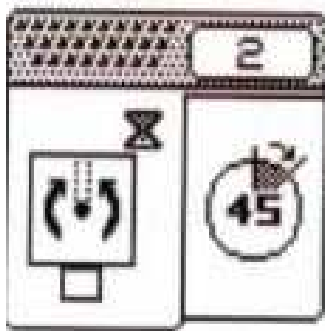


Рис. 15. Блок гироскопа

Добавляем блок двух больших моторов и ставим траекторию движения «Прямо». После добавляем блок «Время» и устанавливаем время движения в 1 секунду.

Программа готова. Общий вид программы представлен на Рис. 16.

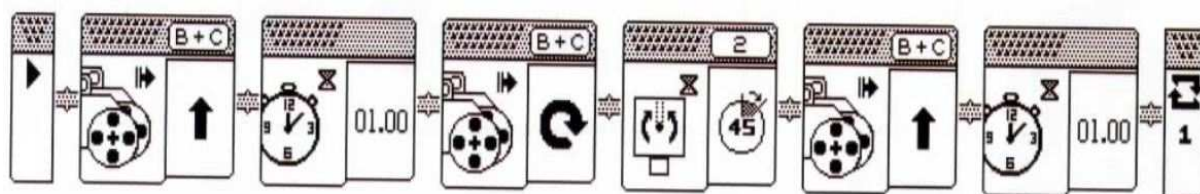


Рис. 16. Общий вид программы

Варианты для самостоятельной работы

Вариант №1.

Задать роботу движение типа: вперед 1 сек.-поворот на 45⁰-движение вперед 1 сек.-поворот на 90⁰-вперед 1 сек.

Вариант №2.

Задать роботу движение типа: назад 1 сек.-поворот на 180⁰-вперед 1 сек.-поворот на 45⁰-назад 1 сек.

Вариант №3.

Задать роботу движение типа: разворот 1 сек.-вперед 1 сек.-поворот на 360⁰-вперед 1 сек.

Вариант №4.

Задать роботу движение типа: вперед 1 сек.-поворот на 360⁰-назад 1 сек.-поворот на 360⁰.

Вариант №5.

Задать роботу движение типа: поворот на 180⁰-вперед 1 сек.-поворот направо 1 сек.-поворот на 45⁰-вперед 1 сек.

Вариант №6.

Задать роботу движение типа: поворот на 45⁰-вперед 2 сек.-поворот на 45⁰-назад 1 сек.

Задание №4. Принцип работы с датчиком касания

Цель раздела в ознакомлении студентов с принципами срабатывания датчика касания. Датчик касания реагирует на прикосновение, что позволяет контролировать многие условия окружающей среды.

Общую программу студенты выполняют в группах по два человека на один робот. Самостоятельная работа выполняется каждым индивидуально.

Программа. Добавление датчика касания и его программирование

При прикосновении к датчику, срабатывает звуковой сигнал, и робот начинает движение назад на протяжении 1 сек.

Выполнение.

Открываем новую программу и добавляем блок датчика касания Рис. 17. В настройках ставим порт ввода 1.

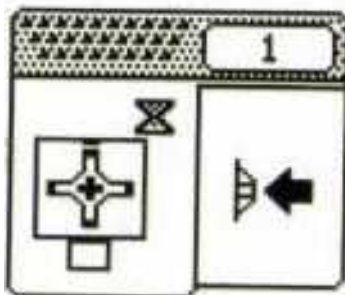


Рис. 17. Блок датчика касания

Далее добавляем блок звукового сигнала. В настройках устанавливаем один звуковой сигнал Рис. 18.

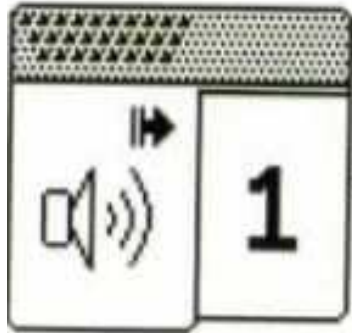


Рис. 18. Блок звукового сигнала

После звукового сигнала робот должен начать движение. Добавляем блок двух больших моторов. В настройках устанавливаем направление движения «Назад» Рис. 19.

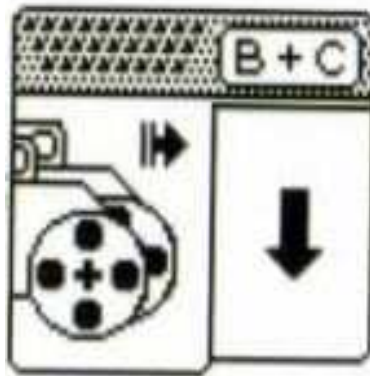


Рис. 19. Блок моторов

Далее добавляем блок «Время». В настройках блока ставим цифру 1.

Общий вид программы представлен на Рис. 20.

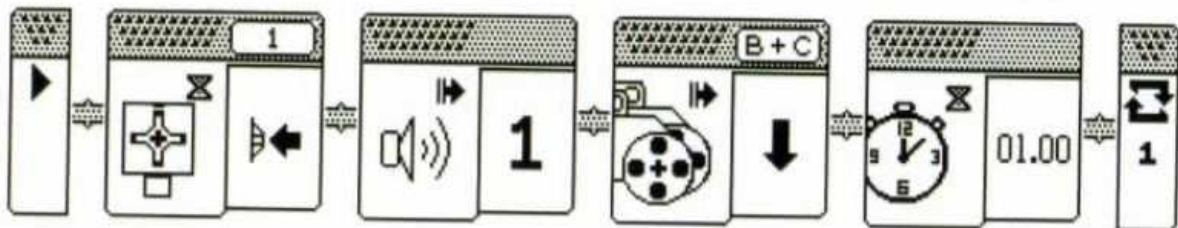


Рис. 20. Общий вид программы

Варианты для самостоятельной работы

Вариант №1.

При нажатии на кнопку робот двигается в прямом направлении 1 сек., после чего поворачивается на 90^0 . Далее движение идет в прямом направлении 1 сек., поворачивается на 45^0 и едет назад 2 сек.

Вариант №2.

При нажатии на кнопку робот двигается в прямом направлении, пока не обнаружит препятствие в виде цветного квадрата на расстоянии 25 см. от него, после чего поворачивается на 160^0 и едет в прямом направлении еще 2 сек.

Вариант №3.

Нажатием на кнопку робот начинает движение прямо, пока не обнаружит препятствие в виде цветного квадрата на расстоянии 30 см. от него, после чего останавливается и выдает звуковой сигнал.

Вариант №4.

Робот двигается в прямом направлении 2 секунды, выдает звуковой сигнал, далее нажимаем на датчик касания и робот поворачивается на 45^0 проезжает прямо 1 секунду и останавливается.

Вариант №5.

Нажатием на кнопку робот начинает движение прямо 2 сек., поворачивается на 180^0 , едет прямо 1 сек., выдает звуковой сигнал и едет назад еще 1 сек.

Вариант №6.

Нажатием на кнопку робот поворачивает на 45^0 , едет прямо 2 сек., поворачивается на 360^0 и едет назад еще 1 сек.

Контрольные вопросы

1. Что такое контроллер?
2. Что такое датчик цвета?
3. Расскажите о принципе работы сервомотора?
4. Что такое датчик касания, и в каких режимах он работает?
5. Что такое ультразвуковой датчик и принцип работы?
6. Какие порты используются для подключения моторов?
7. Какие порты используются для подключения датчиков?
8. Почему датчики и моторы подключаются в разные типы портов?

Лабораторная работа №3


Программирование робота на выполнение простых и сложных действий. Многозадачность

Краткие теоретические сведения


Многозадачность- свойство среды выполнения обеспечивать возможность параллельной обработки нескольких процессов.

Цикл — разновидность управляющей конструкции в высокоуровневых языках программирования, предназначенная для организации многократного исполнения набора инструкций.




Блок «Начало» ()- отмечает начало последовательности программного блока в вашей программе. Ваша программа может иметь более одной последовательности, Все последовательности с блоком «Начало» начнутся автоматически при запуске программы, и последовательность будет выполняться одновременно.




Блок «Рулевое управление» ()- может заставлять робота двигаться вперед, назад, поворачиваться или останавливаться. Вы можете регулировать рулевое управление, чтобы заставить вашего робота идти прямо, двигаться по дуге или делать резкие повороты.

Используйте блок «Рулевое управление» для роботизированных транспортных средств, в которых имеются два больших мотора, где один мотор управляет левой стороной транспортного средства, а второй мотор управляет правой стороной. Блок «Рулевое управление» управляет обоими моторами одновременно, чтобы ваше транспортное средство двигалось в выбранном вами направлении.



Блок «Ожидание» ()- заставляет вашу программу ждать чего-либо, прежде чем перейти к следующему блоку в последовательности. Можно ждать определенное количество времени до тех пор, пока датчик не достигнет определенного значения, или, пока значение датчика не изменится.



Блок «Цикл» () – это контейнер, содержащий последовательность программных блоков. Он заставляет последовательность блоков, находящихся внутри него, повторяться. Вы можете выбрать постоянное повторение блоков, определенное количество раз или до того, пока проверка датчика или другие условия не будут истинной.

Повторяются только блоки, находящиеся внутри цикла. После окончания цикла программа продолжит с блоками, находящимися за пределами цикла.

Задание №1. Простые действия робота

Цель данного задания заключается в освоении простых действий в программировании и построении программных блоков. Задание содержит две общие программы и программы для самостоятельной работы.


Студенты делятся на группы по два человека и выполняют 2 программы. После выполнения двух общих программ, студенты берут в соответствии со своим вариантом задание и выполняют его индивидуально.

Программа №1. Перемещение по прямой линии

Этой программой робот будет двигаться по прямой линии вперед и назад. Работа моторов будет измеряться тремя единицами: количеством оборотов, количеством секунд и количеством градусов.

Выполнение.

Прежде чем открыть новый проект, запускаем программу LEGO

MINDSTORMS Education EV3. Ярлык  находится на рабочем столе. После открытия программы выбираем вкладку «Файл» Рис. 1.

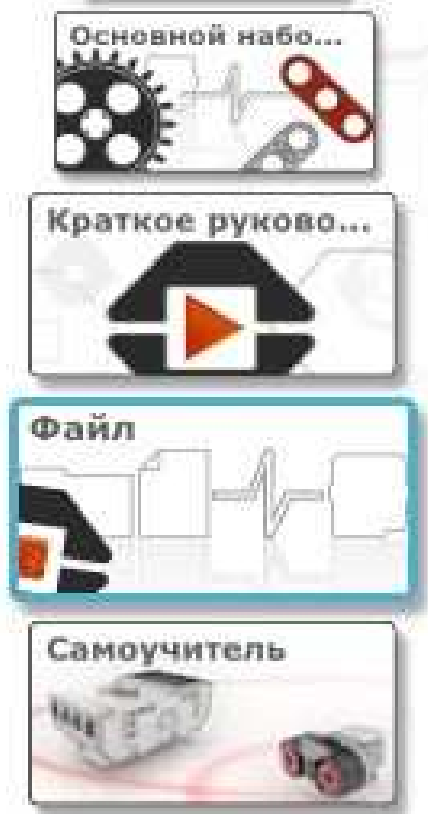


Рис. 1. Вкладка «Файл»

Нажав на вкладку «Файл», выбираем «Новый проект» Рис. 2.



Рис. 2. Вкладка «Новый проект»

Далее выбираем вкладку «Программа» и нажимаем на кнопку «Открыть», расположенную справа Рис. 3.

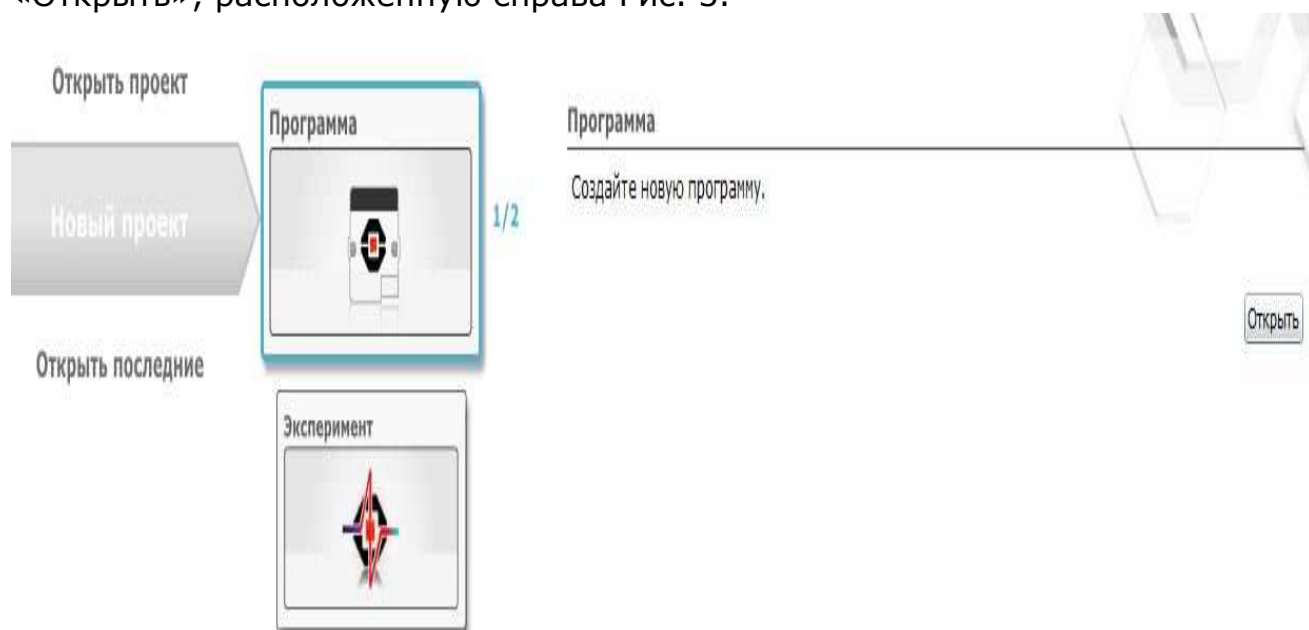


Рис. 3. Вкладка «Программа»

После этого нам открывается стартовое окно нашей новой программы Рис. 4.

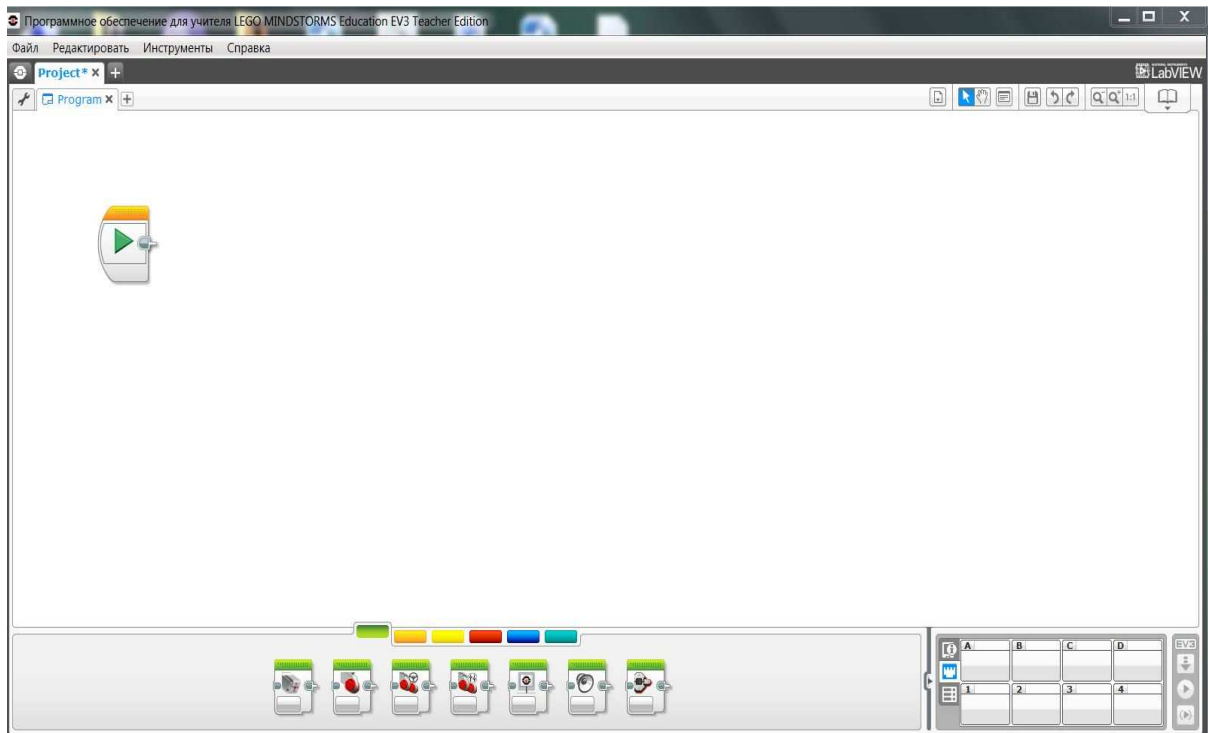


Рис. 4. Стартовое окно новой программы



Во вкладке «Действие»  выбираем блок «Рулевое управление»  и переносим к блоку «Старт», соединяем два блока Рис. 5.





Рис. 5. Блок «Старт» и «Рулевое управление»

В конфигурациях блока «Рулевое управление» количество оборотов ставим равное 2 Рис. 6.



Рис. 6. Настройка блока «Рулевое управление»

Открываем вкладку «Управление операторами»  и выбираем блок «Ожидание» , соединяем с блоком «Рулевое управление» Рис. 7. Блок «Ожидание» по умолчанию имеет значение 1 сек.

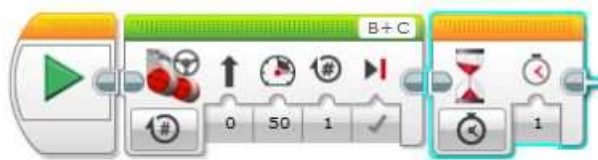


Рис. 7. Добавление блока «Ожидание»




Во вкладке «Действие»  выбираем блок «Рулевое управление»  и переносим к блоку «Ожидание»  Рис. 8.



Рис. 8. Добавление блока «Рулевое управление»

И редактируем последний добавленный блок «Рулевое управление». Нажимаем на блок, выбираем значок оборотов и выбираем «Включить на количество градусов» Рис. 9.

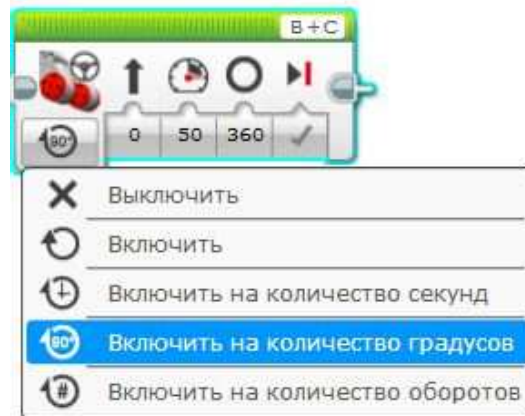


Рис. 9. Изменение блока «Рулевое управление»

Далее жмем на значок «Мощность» и выставляем мощность работы моторов -50 Рис. 10. Это значит, что робот будет двигаться назад на средней мощности.

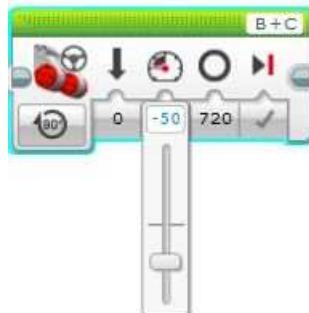




Рис. 10. Изменение мощности в блоке «Рулевое управление»

После этого нажимаем на значок «Градусы» и ставим количество градусов 720 Рис. 11. Это означает, что моторы выполнят два оборота.



Рис. 11. Редактирование блока «Рулевое управление»

Открываем вкладку «Управление операторами»  и выбираем

блок «Ожидание» , соединяем с блоком «Рулевое управление» Рис. 12.

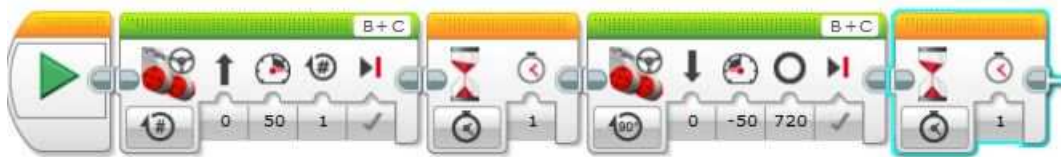




Рис. 12. Добавление нового блока

Во вкладке «Действие»  выбираем блок «Рулевое

управление»  и переносим к блоку «Ожидание» . В настройках блока нажимаем на значок оборотов и выбираем «Включить на количество секунд» Рис. 13. Это значит, что работа моторов будет измеряться в секундах.

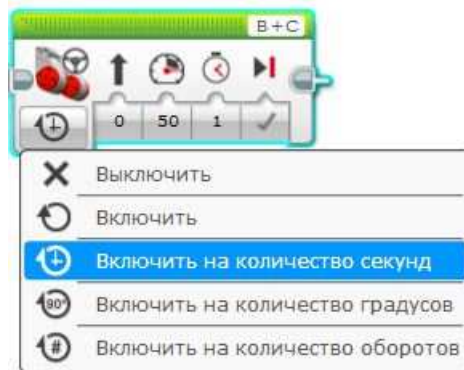


Рис. 13. Настройка блока

Готовая программа продемонстрирована на Рис. 14.



Рис. 14. Готовая программа

Далее проверяем подключение к контроллеру. В правом нижнем углу располагается окно управления подключением Рис. 15.

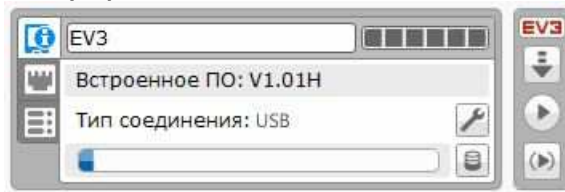



Рис. 15. Окно управления подключением

При активном подключении значок  будет красным. Заходим в «Просмотр портов» и проверяем соответствие номеров портов с моторами. Моторы подключены к портам В и С Рис. 16.

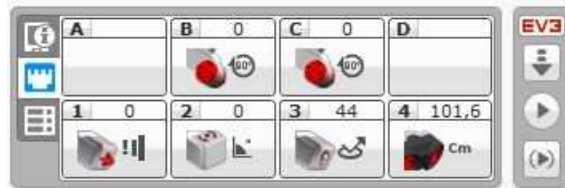



Рис. 16. Просмотр портов подключения

Если все подключено правильно, нажимаем на «Загрузить и запустить» . Программа готова.

Программа №2. Работа с объектом

В данной программе роботу будет задана задача в виде сближения с предметом и удаления от него. В основе действия лежит работа ультразвукового датчика.

Выполнение.

Создаем новый проект. Во вкладке «Действие»  выбираем



блок «Рулевое управление» и добавляем в программу. Нажимаем на значок оборотов и выбираем «Включить» Рис. 17. Это означает, что движение робота неограниченно и будет продолжаться до тех пор, пока не будет добавлено новое ограничение.

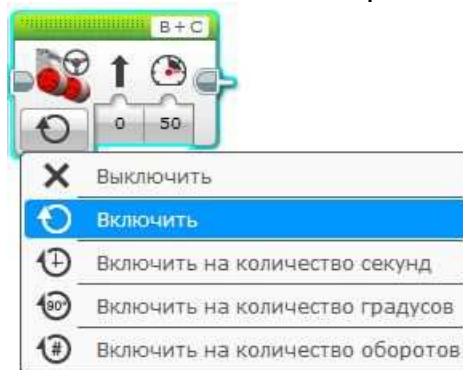




Рис. 17. Редактирование блока

Открываем вкладку «Управление операторами»  и выбираем

блок «Ожидание» , добавляем в программу. Нажимаем на блок и начинаем его редактирование. Выбираем значок «Время». Во всплывающем списке выбираем «Ультразвуковой датчик»-«Изменить»-«Расстояние в сантиметрах» Рис. 18.

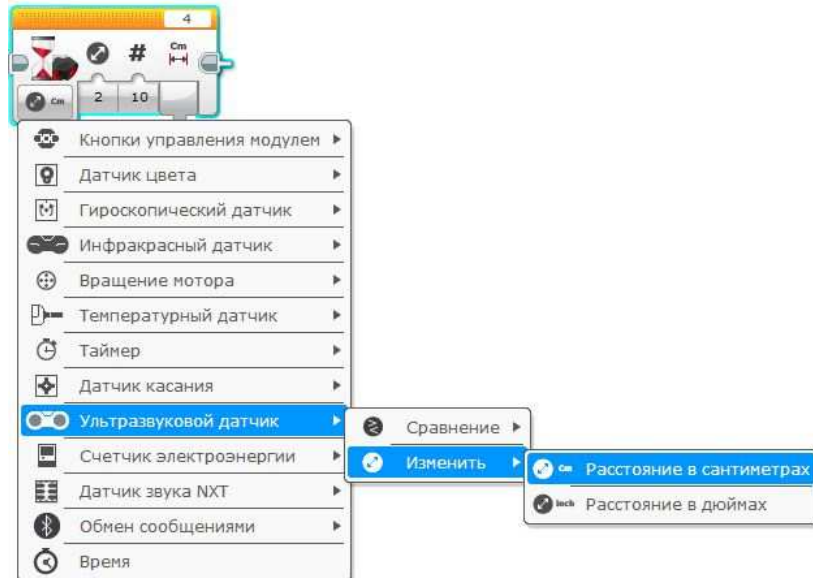


Рис. 18. Редактирование блока «Ожидание»

Далее нажимаем на цифру 2 и выбираем «1», что означает «Уменьшить» Рис. 19. Это означает, что направление движения робота будет стремиться к уменьшению расстояния между ним и предметом.

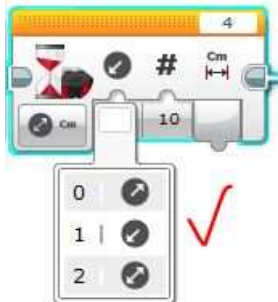


Рис. 19. Редактирование блока «Ожидание»

Далее в строке «Сумма» ставим цифру 11. Это значит, что робот уменьшит свое расстояние до предмета на 11 см. Рис. 20.

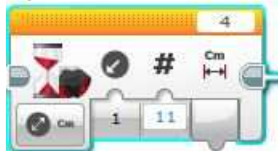



Рис. 20. Заполнение строки «Сумма»

Во вкладке «Действие»  выбираем блок «Рулевое

управление»  и добавляем в программу к последнему блоку. Нажимаем на значок оборотов и выбираем «Выключить» Рис. 21. Это значит, что робот остановится после совершения предыдущего действия.

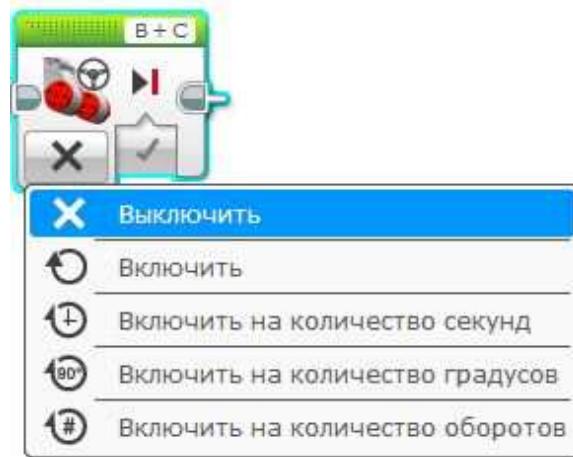





Рис. 21. Редактирование оборотов

Открываем вкладку «Управление операторами»  и выбираем

блок «Ожидание» , добавляем в программу к последнему блоку. Во вкладке «Действие»  выбираем блок «Рулевое управление»



и добавляем в программу к последнему блоку. Редактируем блок. Нажимаем на значок оборотов и выбираем «Включить» Рис. 22.

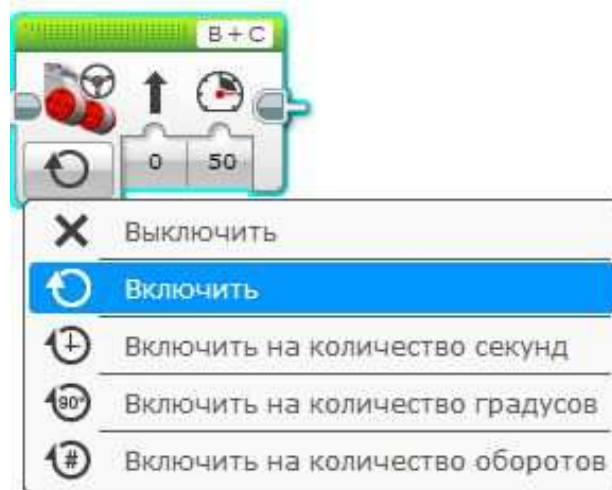


Рис. 22. Редактирование блока

Далее устанавливаем мощность «-50» Рис. 23.

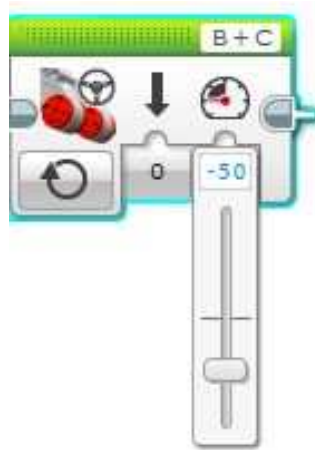




Рис. 23. Установка мощности моторов

Открываем вкладку «Управление операторами»  и выбираем

блок «Ожидание» , добавляем в программу. Нажимаем на блок и начинаем его редактирование. Выбираем значок «Время». Во всплывающем списке выбираем «Ультразвуковой датчик»-«Изменить»-«Расстояние в сантиметрах» Рис. 24.

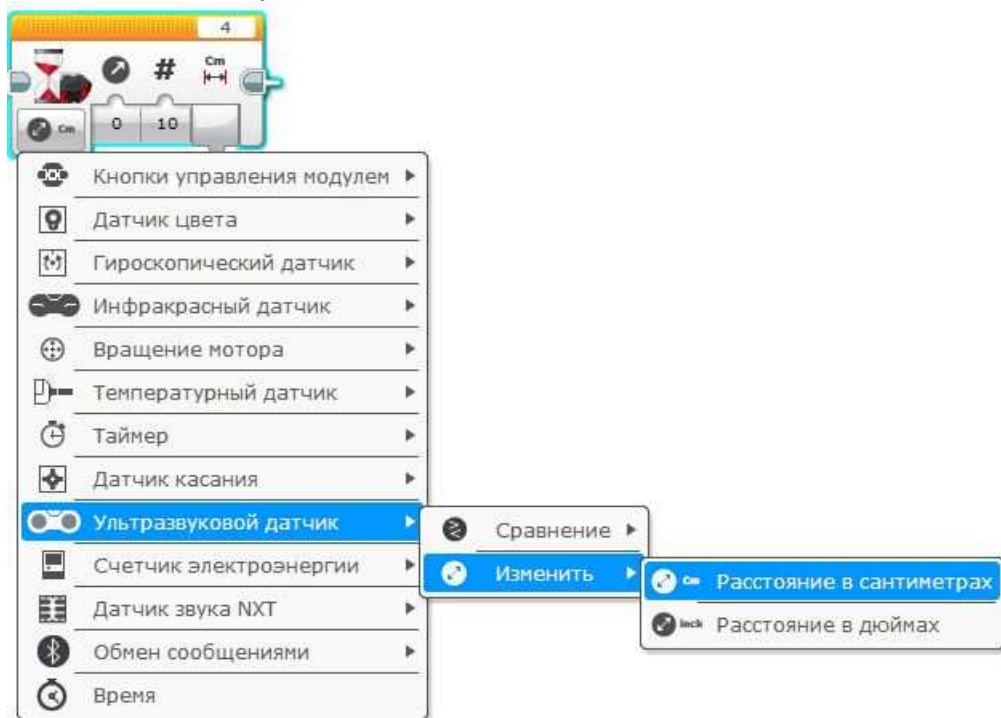


Рис. 24. Редактирование блока

Далее нажимаем на цифру 2 и выбираем «0», что означает «Увеличить» Рис. 25. Это означает, что направление движения робота будет стремиться к увеличению расстояния между ним и предметом.

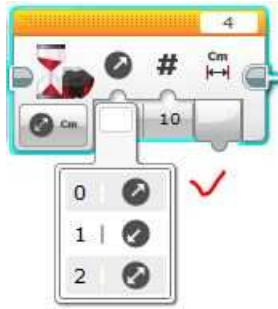




Рис. 25. Установка направления движения

Далее в строке «Сумма» ставим цифру 6. Это значит, что робот увеличит свое расстояние от предмета на 6 см. Рис. 26.



Рис. 26. Заполнение строки «Сумма»

Во вкладке «Действие»  выбираем блок «Рулевое управление»  и добавляем в программу к последнему блоку. Нажимаем на значок оборотов и выбираем «Выключить» Рис. 27. Это значит, что робот остановится после совершения предыдущего действия.

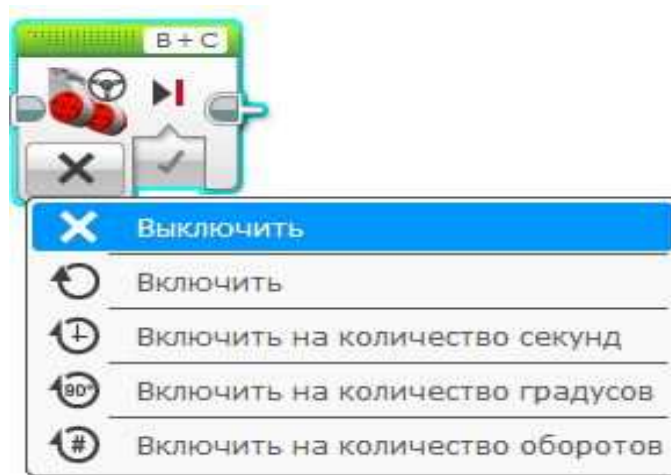


Рис. 27. Остановка движения

После этого программа завершена. Общий вид программы представлен на Рис. 28.



Рис. 28. Общий вид программы

Запускаем программу.

Варианты для самостоятельной работы

Вариант №1.

Задать роботу движение типа: вперед 1 сек-назад 1 сек.

Вариант №2.

Задать роботу движение типа: вперед 2 сек.-назад 1сек.

Вариант №3.

Задать роботу движение типа: уменьшить расстояние до предмета на 10 см-остановка.

Вариант №4.

Задать роботу движение типа: уменьшить расстояние до предмета на 15 см-остановка.

Вариант №5.

Задать роботу движение типа: увеличить расстояние от предмета на 10 см-остановка.

Вариант №6.

Задать роботу движение типа: вперед 2 сек.-назад 2 сек.

Задание №2. Более сложные действия

Цель задания №2 в рассмотрении более сложных процедур, которые необходимы для программирования робота. Задание содержит две общие программы и программы для самостоятельной работы.

Студенты делятся на группы по два человека и выполняют 2 программы. После выполнения двух общих программ, студенты берут в соответствии со своим вариантом задание и выполняют его индивидуально.

Программа №1. Многозадачность

Напишем программу на выполнение нескольких действий одновременно. Робот будет двигаться вперед и одновременно воспроизводить звуковой сигнал.

Выполнение.

Создаем новый проект.

Во вкладке «Действие»  выбираем блок «Рулевое


управление»  и добавляем в программу на расстоянии от блока «Старт» Рис. 29.




Рис. 29. Добавление блока

Редактируем блок, указывая в количестве оборотов цифру 2 Рис. 30.



Рис. 30. Изменение количества оборотов

Во вкладке «Действие»  выбираем блок «Звук» и подключаем его к блоку «Рулевое управление». Редактируем блок «Звук». Нажимаем на блок и выбираем «Воспроизвести файл». Далее выбираем «Остановка» Рис. 31.

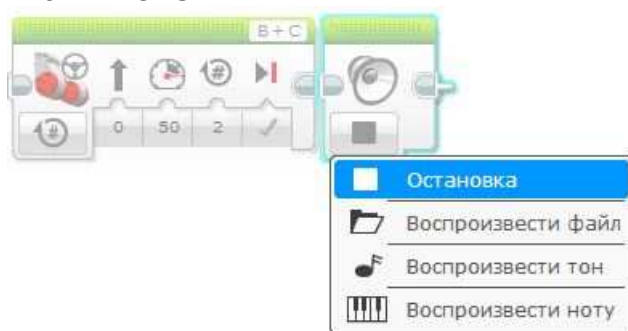


Рис. 31. Редактирование блока «Звук»


Далее во вкладке «Действие»  выбираем блок «Звук» и располагаем его ниже двух блоков, не соединяя Рис. 32.



Рис. 32. Общий вид программы

Далее редактируем последний блок. Нажимаем на пустой прямоугольник в верхнем правом углу блока и во всплывающем окне выбираем «Motor idle» Рис. 33.

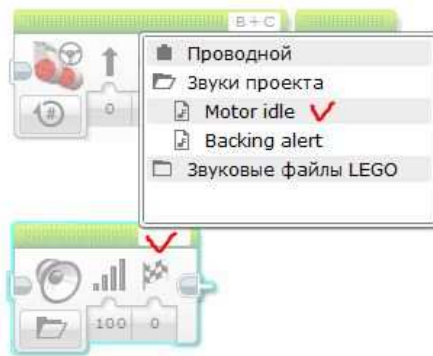


Рис. 33. Редактирование блока «Звук»

Далее нажимаем на флажок и во всплывшем списке выбираем 2
Рис. 34.



Рис. 34. Редактирование блока

Теперь соединяем блок «Старт» с блоком «Рулевое управление» и вторым блоком «Звук». Готовая программа представлена на Рис. 35.

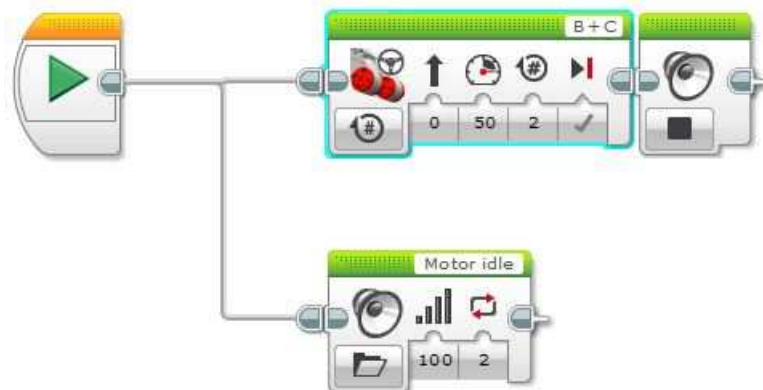


Рис. 35. Соединение блоков



Запустите программу.

Программа №2. Циклические действия

Создадим цикл повторяющихся движений робота вперед-назад в течении 5 секунд. Эта простая программа поможет освоить использования циклов в программе.

Выполнение.

Создайте новый проект. Открываем вкладку «Управление

операторами»  и выбираем блок «Цикл» . Соединяем с блоком «Старт» Рис. 36.

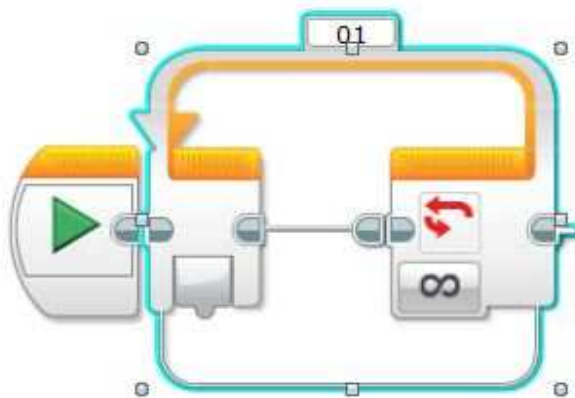




Рис. 36. Блок «Цикл»

Во вкладке «Действие»  выбираем два блока «Рулевое управление»  и добавляем в программу в центр цикла Рис. 37.

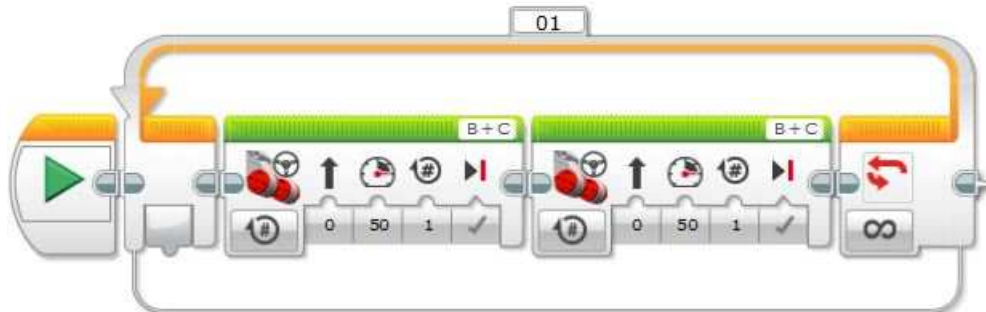


Рис. 37. Блоки в цикле

Первый блок оставляем без изменений. Во втором блоке редактируем мощность и ставим -50 Рис. 38.

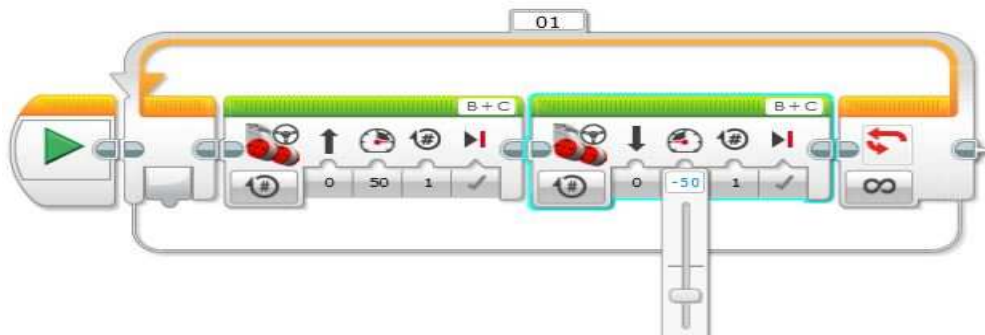


Рис. 38. Редактирование второго блока

Далее нажимаем на знак бесконечности в окончании цикла и выбираем из списка позицию «Время» Рис. 39.

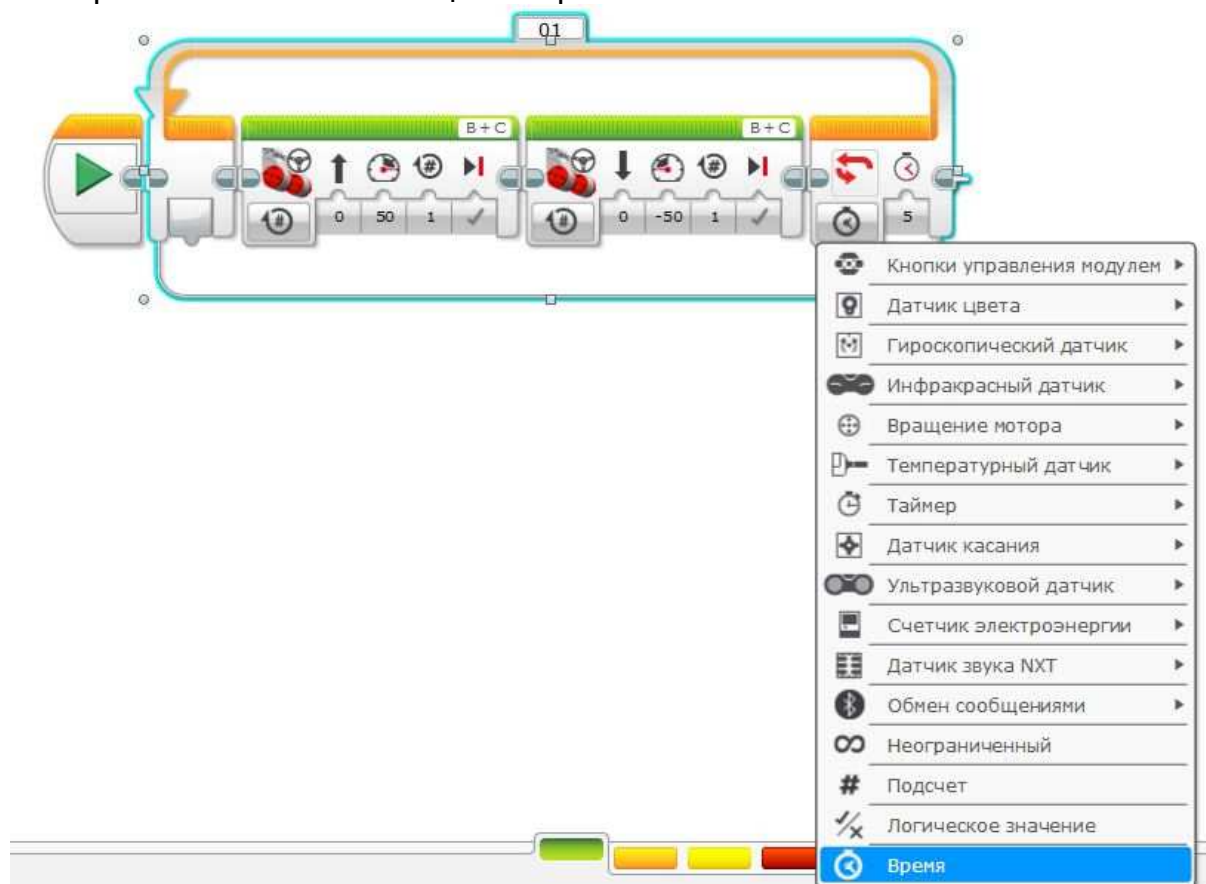


Рис. 39. Редактирование цикла

По умолчанию время данного цикла равно 5 сек., что удовлетворяет нашей задаче. Цикл готов, запускаем программу.

Варианты для самостоятельной работы

Вариант №1.

Цикл 4 сек.: робот двигается вперед- назад.

Вариант №2.

Робот двигается прямо 3 сек. и выдает звуковой сигнал одновременно.

Вариант №3.

Робот поворачивается вокруг себя 5 сек. и выдает звуковой сигнал одновременно.

Вариант №4.

Цикл 5 сек.: робот поворачивается вокруг себя 1 раз- едет прямо 1 оборот.

Вариант №5.

Цикл 4 сек.: робот едет 2 сек. назад- поворачивается на 360°.

Вариант №6.

Робот двигается назад 5 сек и выдает звуковой сигнал одновременно.

Контрольные вопросы

1. Что такое многозадачность?
2. Что такое цикл?
3. Каким способом задавалась роботу многозадачность действий?
4. Перечислите единицы измерения работы большого мотора?
5. Что такое блок «Ожидание» и принцип его работы?
6. Что такое блок «Цикл» и принцип его работы?
7. Что такое блок «Рулевое управление» и принцип его работы?
8. Что такое блок «Начало»?

Лабораторная работа №4

Шины данных. Регистрация собранных данных

Краткие теоретические сведения

Шина - это набор соединений, по которым передаются различные сигналы.

Шина данных - шина данных позволяет вам передать входящее значение для программного блока, используя выходное значение другого блока вашей программы. Это позволяет вам создавать взаимосвязи между блоками и программировать более сложное поведение вашего робота.

Шины данных передают значения от одного блока другому. Каждая шина данных имеет тип, который определяется типом вывода блока в начале шины. Кроме того, это является типом значения, передаваемого шиной данных.

Шины данных, вводы блока и выходы блока выглядят по-разному в зависимости от их типа, как показано в таблице ниже в Таблице №1.




















Блок «Ультразвуковой датчик» ()- блок ультразвукового цвета получает данные от ультразвукового датчика. Вы можете измерить расстояние, как в дюймах, так и в сантиметрах и получить числовой вывод. Кроме того, вы можете сравнить расстояние с пороговым значением и получить логический вывод (истина или ложь). Также вы можете обнаруживать другие ультразвуковые сигналы в пассивном режиме.

Таблица №1 - Типы шин данных.

Тип	Ввод блока	Вывод блока	Вывод блока Шина данных
Логическое значение			
Числовое значение			
Текст			
Числовой массив			
Логический массив			



Блок «Текст» () - текстовый блок может объединять до трех текстовых элементов в один текстовый элемент.

Вводы текстового блока позволяют объединять до трех текстовых элементов. Вы можете вводить значения ввода прямо в блоке. Или же, как вариант, значения можно передавать по шинам данных от выводов других программных блоков Таблица №2.

Таблица №2 - Вводы текстового блока


Ввод	Тип	Допустимые значения	Примечания
A	Текст	Любой текст	Пропущено, если не включается
B	Текст	Любой текст	Пропущено, если не включается
C	Текст	Любой текст	Пропущено, если не включается

Вывод текстового блока дает объединенный текстовый элемент. Для использования вывода воспользуйтесь шиной данных для его подключения к другому программному блоку Таблица №3.

Таблица №3 - Выводы текстового блока.

Вывод	Тип	Примечания
Результат	Текст	За элементом А следует элемент В , затем элемент С



Блок «Экран» ()- может отображать текст или графику на экране модуля EV3.

Осциллограф- прибор, предназначенный для исследования (наблюдения, записи, измерения) амплитудных и временных параметров электрического сигнала.

Осциллоскоп- прибор, предназначенный для наблюдения за процессами в электрических цепях, представляющий собою упрощенный осциллограф.

Режим осциллоскопа позволяет видеть значения датчиков, подключенных к модулю EV3 в режиме реального времени.

Задача №1. Знакомство с шинами данных

Цель данного раздела в ознакомлении с методом передачи информации при помощи шин данных. Для этого напишем программу, в которой используется три типа шин данных. Использование шин данных представлено в виде соединений между блоков, по которым будет передаваться необходимая информация, которая собирается ультразвуковым датчиком.

Общую программу студенты выполняют в группах по два человека на один робот. Самостоятельная работа выполняется каждым индивидуально.

Программа. Измерение расстояния до предмета

Запрограммируем робота при помощи ультразвукового датчика измерять расстояние в сантиметрах до цветного квадрата до тех пор, пока расстояние между датчиком и роботом не будет равно или меньше 8 см. Как только расстояние станет 8 см., на экране контроллера появляется на 2 сек. стандартное изображение «Up». Все данные измерения должны отображаться на экране контроллера.

Выполнение.

Создаем проект. Открываем вкладку «Управление операторами»



 и выбираем блок «Цикл» . Соединяем с блоком «Старт» Рис. 1.

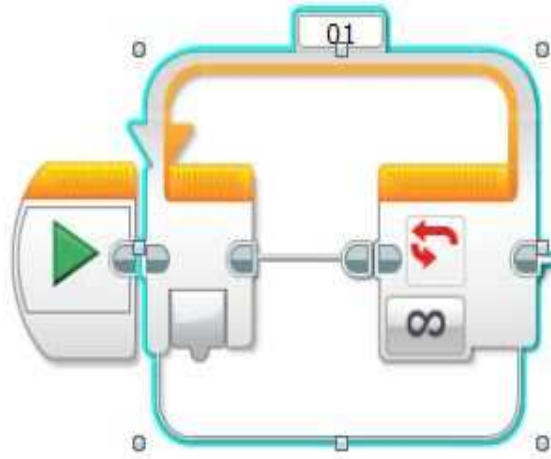



Рис. 1. Цикл

Во вкладке «Датчик»  выбираем блок «Ультразвуковой

датчик»  и добавляем в цикл. Теперь редактируем блок. Ждем на кнопку «Измерение». Далее во всплывающем списке выбираем «Сравнение»-«Расстояние в сантиметрах» Рис. 2.

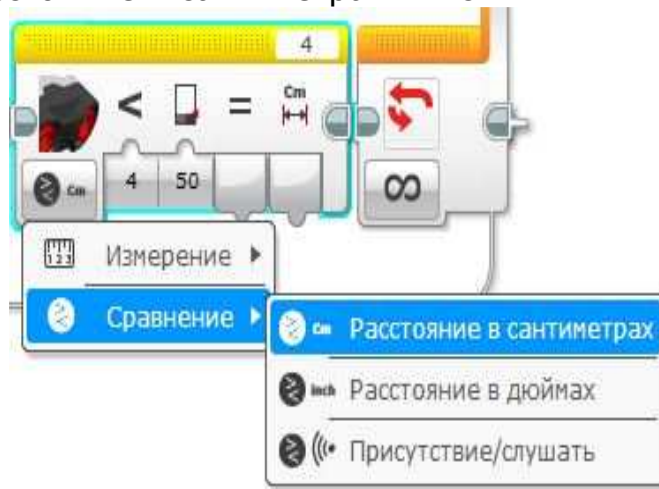


Рис. 2. Редактирование блока

В «Тип сравнения» выбираем значок «Меньше» Рис. 3.

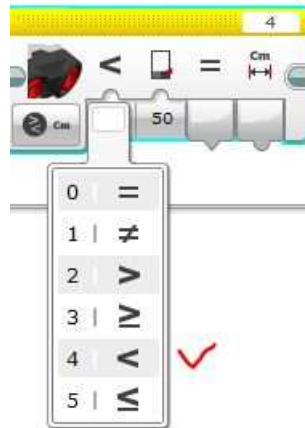


Рис. 3. Выбираем тип сравнения

Пороговое значение указываем 8 Рис.4.

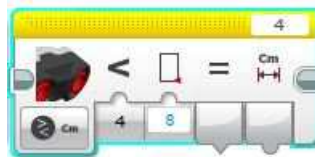


Рис.4. Пороговое значение

Во вкладке «Операции с данными»  выбираем блок «Текст»



и добавляем к блоку «Ультразвуковой датчик». Начинаем редактирование блока «Текст». Для этого возвращаемся к блоку «Ультразвуковой датчик» и соединяем точку «Расстояние в сантиметрах» ультразвукового датчика с точкой соединения «А» блока «Текст». На Рис. 5 желтая линия.




Рис. 5. Устанавливаем соединение

Далее в блоке «Текст» в позиции «В» ставим единицу измерения английскими буквами «см» Рис. 6.



Рис. 6. Ставим единицу измерения

После этого во вкладке «Действие»  выбираем блок «Экран»



и добавляем к блоку «Текст». Далее в блоке «Экран» нажимаем на кнопку «Изображение» и выбираем позицию «Текст»-«Сетка» Рис. 7.

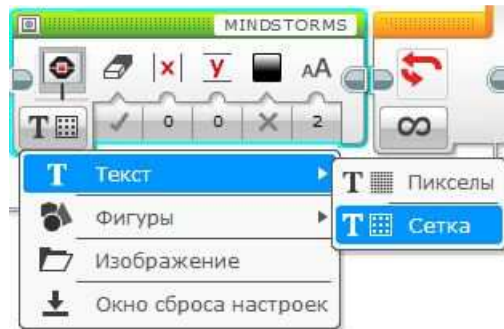


Рис. 7. Редактирование блока

Далее нажимаем на строку в правом верхнем углу блока, где написано «MINDSTORMS» и выбираем «Проводной» Рис. 8.



Рис. 8. Редактирование блока

Возвращаемся к блоку «Текст» и соединяем точку «Результат» с точкой «Текст» блока «Экран». На Рис. 9 оранжевая линия.

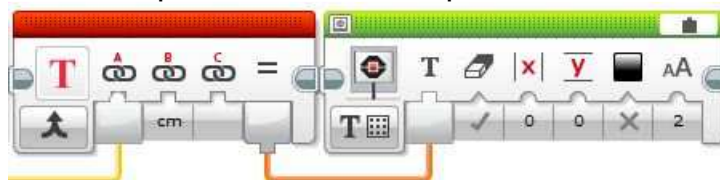


Рис. 9. Устанавливаем соединение

Возвращаемся к блоку «Экран». Столбец «X» будет иметь значение 5, а строка «Y» значение 6 Рис. 10.



Рис. 10. Устанавливаем значения столбца и строки

Теперь все блоки цикла у нас имеются. Осталось отредактировать сам цикл. Для этого в конце цикла в блоке нажимаем на значок бесконечности и выбираем в списке «Логическое значение» Рис. 11.



Рис. 11. Настраиваем конфигурации цикла

После этого возвращаемся к блоку «Ультразвуковой датчик» и

тянем соединительную линию от точки «Результат сравнения» с

точкой «Пока не будет истина: Истина» в конце цикла на Рис. 12 зеленая линия.

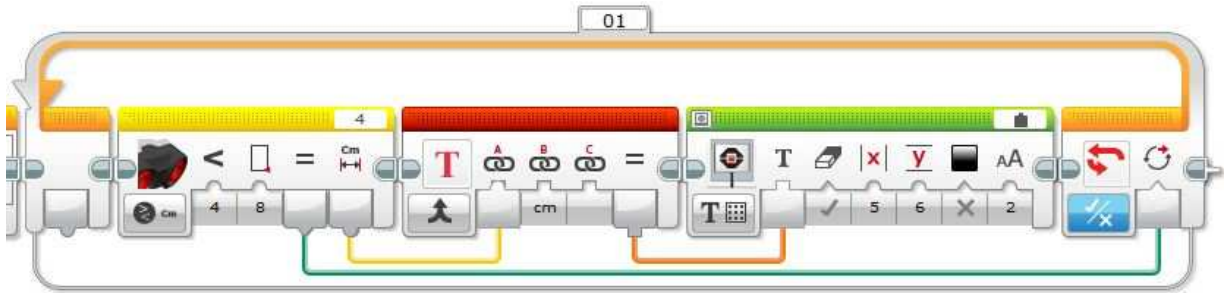



Рис. 12. Готовый цикл


После этого во вкладке «Действие»  выбираем блок «Экран»



и соединяем с циклом. Редактируем блок. Нажимаем на строку «Имя файла» в верхнем правом углу и выбираем из списка «Изображение проекта»-«Up» Рис. 13.



Рис. 13. Редактирование датчика

Открываем вкладку «Управление операторами»  и выбираем




блок «Ожидание» , добавляем в программу к блоку «Экран». Редактируем блок и в строке «Секунды» ставим количество секунд: 2 Рис. 14.



Рис. 14. Устанавливаем секунды

Готовая программа представлена на Рис. 15.

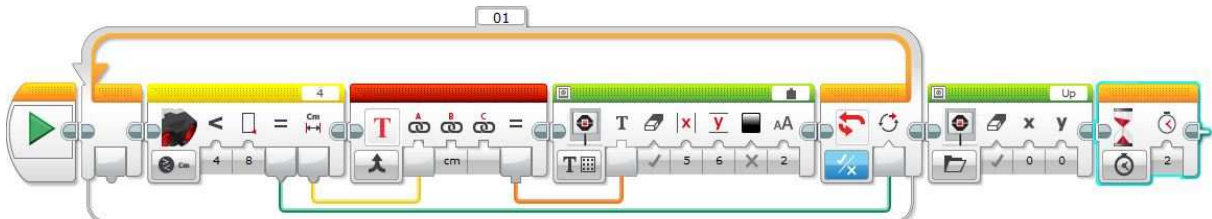


Рис. 15. Готовая программа

Варианты для самостоятельной работы

Вариант №1.

Запрограммируйте робота при помощи ультразвукового датчика измерять расстояние в сантиметрах до цветного квадрата до тех пор, пока расстояние между датчиком и квадратом не будет меньше или равно 15 см. Как только расстояние станет 15 см., на экране контроллера появляется на 5 сек. стандартное изображение «Up». Все данные измерения должны отображаться на экране контроллера.

Вариант №2.

Запрограммируйте робота при помощи ультразвукового датчика измерять расстояние в сантиметрах до цветного квадрата до тех пор, пока расстояние между датчиком и квадратом не будет равно 10 см. Как только расстояние станет 10 см., на экране контроллера появляется на 1 сек. стандартное изображение «Up». Все данные измерения должны отображаться на экране контроллера.

Вариант №3.

Запрограммируйте робота при помощи ультразвукового датчика измерять расстояние в сантиметрах до цветного квадрата до тех пор, пока расстояние между датчиком и квадратом не будет меньше 9 см. Как только расстояние станет менее 9 см., воспроизвести звуковой сигнал из перечня стандартных звуковых сигналов LEGO. Продолжительность сигнала 2 сек. Все данные измерения должны отображаться на экране контроллера.

Вариант №4.

Запрограммируйте робота при помощи ультразвукового датчика измерять расстояние в сантиметрах до цветного квадрата до тех пор, пока расстояние между датчиком и квадратом не будет меньше или равно 12 см. Как только расстояние станет менее 12 см., воспроизвести звуковой сигнал из перечня стандартных звуковых сигналов LEGO. Продолжительность сигнала 1 сек. Все данные измерения должны отображаться на экране контроллера.

Вариант №5.

Запрограммируйте робота при помощи ультразвукового датчика измерять расстояние в сантиметрах до цветного квадрата до тех пор, пока расстояние между датчиком и квадратом не будет равно 10 см. Как только расстояние станет менее 10 см., воспроизвести звуковой сигнал из перечня стандартных звуковых сигналов LEGO. Продолжительность сигнала 3 сек. Все данные измерения должны отображаться на экране контроллера.

Вариант №6.

Запрограммируйте робота при помощи ультразвукового датчика измерять расстояние в сантиметрах до цветного квадрата до тех пор, пока расстояние между датчиком и квадратом не будет меньше или равно 9 см. Как только расстояние станет 9 см., на экране контроллера появляется на 3 сек. стандартное изображение «Up». Все данные измерения должны отображаться на экране контроллера.

Задача №2. Регистрация собранных данных

Цель раздела заключается в ознакомлении с удаленным режимом сбора данных о внешней среде при помощи датчиков. Данный режим является основным режимом работы роботов и роботизированных систем.

В качестве базовой, будет рассмотрена программа по сбору данных датчика цвета в удаленном режиме и их регистрации на компьютере. Программу студенты выполняют в группах по два человека на один робот.

После освоения базовой программы, каждый студент в соответствии со своим вариантом выполняет индивидуальное задание на другие виды датчиков, при помощи которых также можно собирать информацию в удаленном режиме.

Программа. Регистрация удаленных данных

В данной программе мы запрограммируем робот в удаленном режиме собирать информацию о внешней среде определенный промежуток времени. После этого мы подключаемся к контроллеру и загружаем собранную информацию, которая отображается в удобном для нас виде. В нашем случае информация будет отображаться в виде графика.

Создадим программу по сбору в удаленном режиме данных датчика цвета, который будет измерять яркость внешнего освещения. После чего загрузим собранную информацию по яркости освещения и отобразим ее в виде графика на экране компьютера.

Выполнение.

Создаем новый эксперимент. Заходим «Файл»-«Новый проект»-«Эксперимент».

Сразу же необходимо сохранить наш эксперимент. Для этого заходим в «Файл»-«Сохранить проект» и сохраняем проект в соответствии со своим порядковым номером. Далее включаем модуль EV3 и подключаем к компьютеру посредством USB. Далее останавливаем режим осциллоскопа Рис. 16.

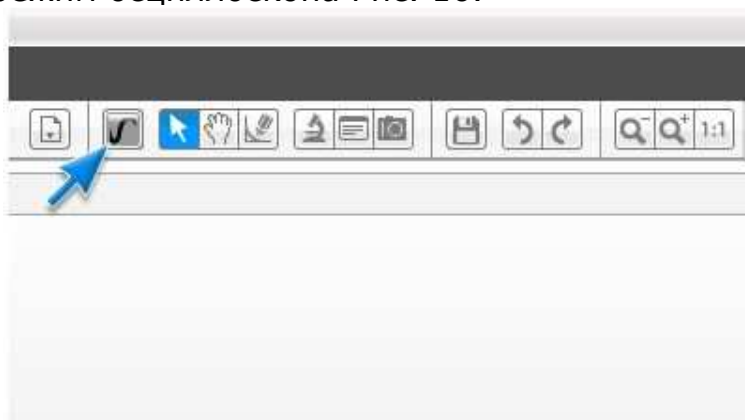


Рис. 16. Отключение режима осциллоскопа

Далее в настройках меняем «Частота выборки» на 20 выборок в секунду и меняем режим датчика цвета на «Яркость внешнего освещения» Рис. 17.

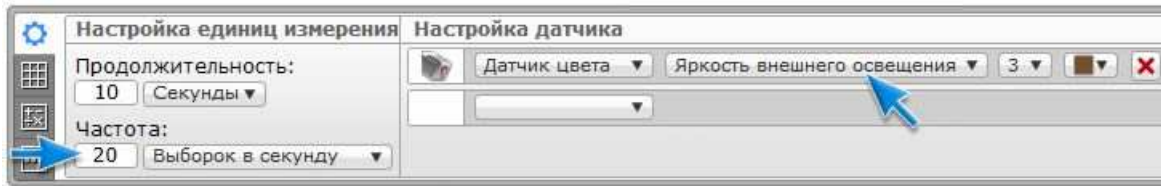


Рис. 17. Изменение настроек

Теперь загружаем эксперимент в модуль Рис. 18.



Рис. 18. Загрузка эксперимента в модуль

После этого отсоединяем USB- кабель и располагаем робот с датчиком возле источника света.

В контроллере находим программу под названием, которое использовалось при сохранении, и запускаем ее. После завершения эксперимента вновь подключаем USB- кабель к компьютеру. Далее нажимаем кнопку «Передать» Рис. 19.



Рис. 19. Загрузка эксперимента

Далее выбираем из списка наш файл и загружаем информацию Рис. 20.

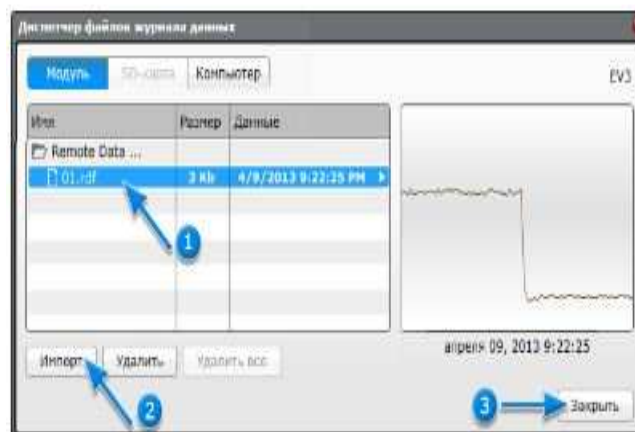


Рис. 20. Выбираем файл с экспериментом

После этого у нас появится график изменения яркости внешнего освещения. На данном графике мы можем проследить за изменениями яркости внешнего освещения в различные промежутки времени.

Варианты для самостоятельной работы

Вариант №1.

Собрать в удаленном режиме при помощи гироскопического датчика угол отклонения. Продолжительность эксперимента 10 сек. Частота выборок-10.

Вариант №2.

Собрать в удаленном режиме при помощи гироскопического датчика скорость перемещения. Продолжительность эксперимента 5 сек. Частота выборок-20.

Вариант №3.

Собрать в удаленном режиме информацию о вращении мотора. Сколько вращений произведет мотор за 7 секунд. Продолжительность эксперимента 7 сек. Частота выборок-10. Единица измерения- градусы.

Вариант №4.

Собрать в удаленном режиме информацию о вращении мотора. Сколько вращений произведет мотор за 5 секунд. Продолжительность эксперимента 5 сек. Частота выборок-10. Единица измерения- количество оборотов.

Вариант №5.

Собрать в удаленном режиме информацию о вращении мотора. Сколько вращений произведет мотор за 10 секунд. Продолжительность эксперимента 10 сек. Частота выборок-10. Единица измерения- количество оборотов.

Вариант №6.

Собрать в удаленном режиме при помощи гироскопического датчика угол отклонения. Продолжительность эксперимента 5 сек. Частота выборок-10.

Контрольные вопросы:

1. Что такое шина?
2. Что такое шина данных?
3. Какие операции производились в программе при помощи шин данных?
4. Что такое режим осциллоскопа?
5. В каких единицах измерения может анализироваться работа мотора?
6. Что такое осциллограф?
7. Что такое блок «Текст»?
8. Что такое блок «Ультразвуковой датчик» и как он работает?
9. Что такое блок «Экран»?


Лабораторная работа №5

Использование блока «Переключатель». Программирование графиков


Краткие теоретические сведения

Яркость – характеристика, равная отношению силы света в определенном направлении к площади проекции светящейся поверхности на плоскость, перпендикулярную этому направлению, и измеряется в канделах на квадратный сантиметр: кд/см².




Блок «Переключатель» ()- блок «Если ... то» – это контейнер, содержащий две или более последовательности программных блоков. Каждая последовательность называется вариантом. Проверка в начале «Если ... то» определяет, какой вариант будет запущен. При каждом выполнении «Если ... то» будет срабатывать только один вариант.



Блок «Большой мотор» ()- управляет большим мотором. Вы можете включать или выключать мотор, управлять его уровнем мощности или включать мотор на определенное количество времени или оборотов.



Блок «Независимое управление моторами» ()- может заставлять робота двигаться вперед, назад, поворачиваться или останавливаться. Используйте блок «Независимое управление моторами» для роботизированных транспортных средств, в которых имеются два больших мотора, где один мотор управляет левой стороной транспортного средства, а второй мотор управляет правой стороной. Можно заставить два мотора вращаться с разными скоростями или в разных направлениях, чтобы ваш робот поворачивался.

Режим программирования графика позволяет роботу выполнять действия на основании данных, зарегистрированных во время эксперимента.

Для программирования графика необходимо настроить пороговые зоны. Зона панели программирования используется для создания набора программных блоков, которые будут работать, пока значение датчика находится в пределах данной конкретной зоны. Каждая зона отделена от следующей зоны пороговым значением. Настройте пороговое значение, для этого перетащите пороговую линию вверх или вниз, или введите конкретное число в графу порогового значения.

Задача. Переключатель. Многопозиционный переключатель

Цель данного раздела заключается познакомиться с блоком «Переключатель» и «Многопозиционный переключатель». С помощью переключателя роботу можно задавать различные условия. Это является необходимым для построения больших программ, содержащих сложные алгоритмы и нацеленных на определение роботу сложных задач.


Используя многопозиционный переключатель, мы можем задавать роботу сложные условия.

Общие программы №1 и №2 студенты выполняют в группах по два человека на один робот. Самостоятельная работа выполняется каждым индивидуально.

Программа №1. Простой переключатель

В данной программе мы рассмотрим простой переключатель на примере небольшой программы. Напишем программу, в которой робот будет двигаться вдоль черной линии и измерять яркость отраженного света при помощи датчика цвета. Если яркость отраженного света будет меньше 50 кд, то робот отклоняется вправо, если яркость отраженного света будет больше 50 кд, то робот отклоняется влево. Действие циклично, пока робот не доедет до конца линии.

Выполнение

Создаем проект. Добавляем в программу блок «Цикл». Далее во вкладке «Управление операторами»  выбираем блок



«Переключатель» и добавляем его в цикл Рис. 1.

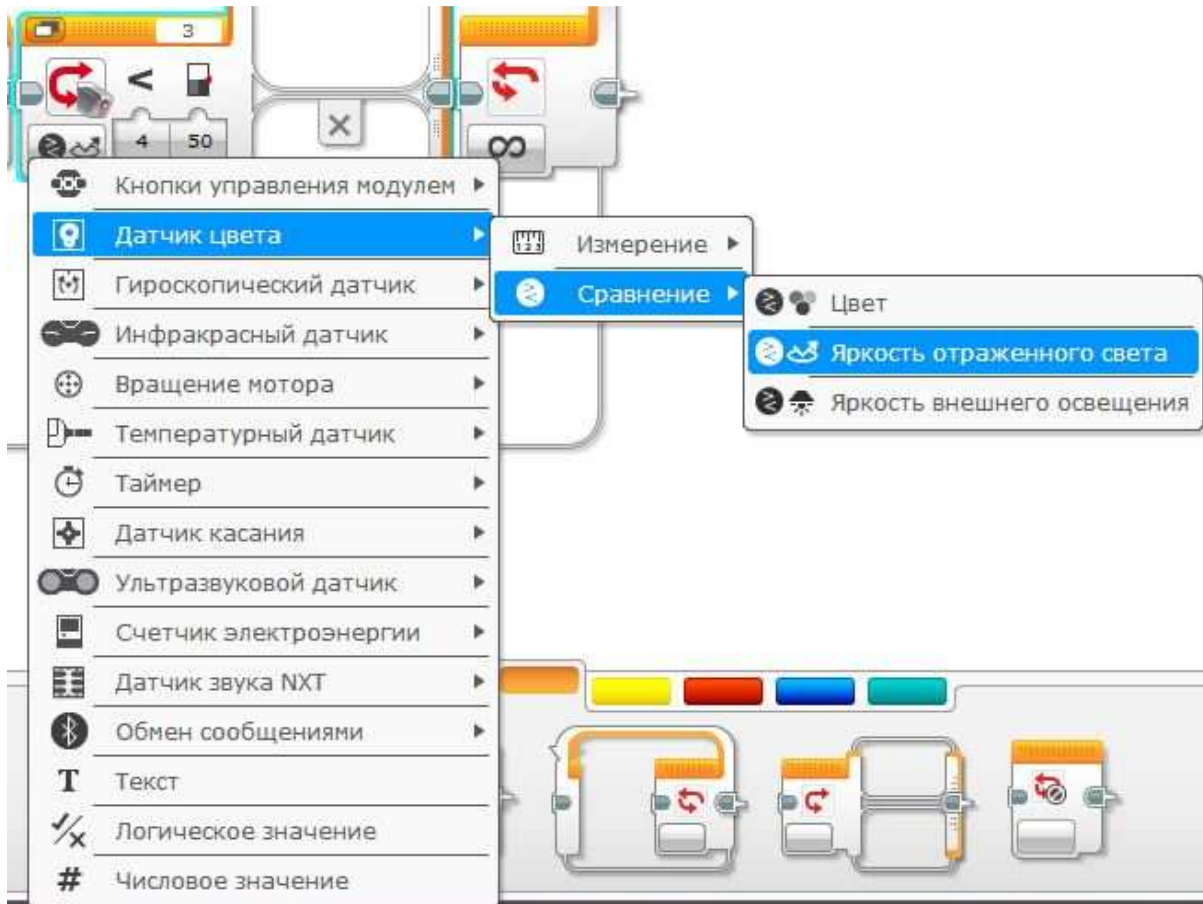


Рис. 2. Редактирование блока «Переключатель»

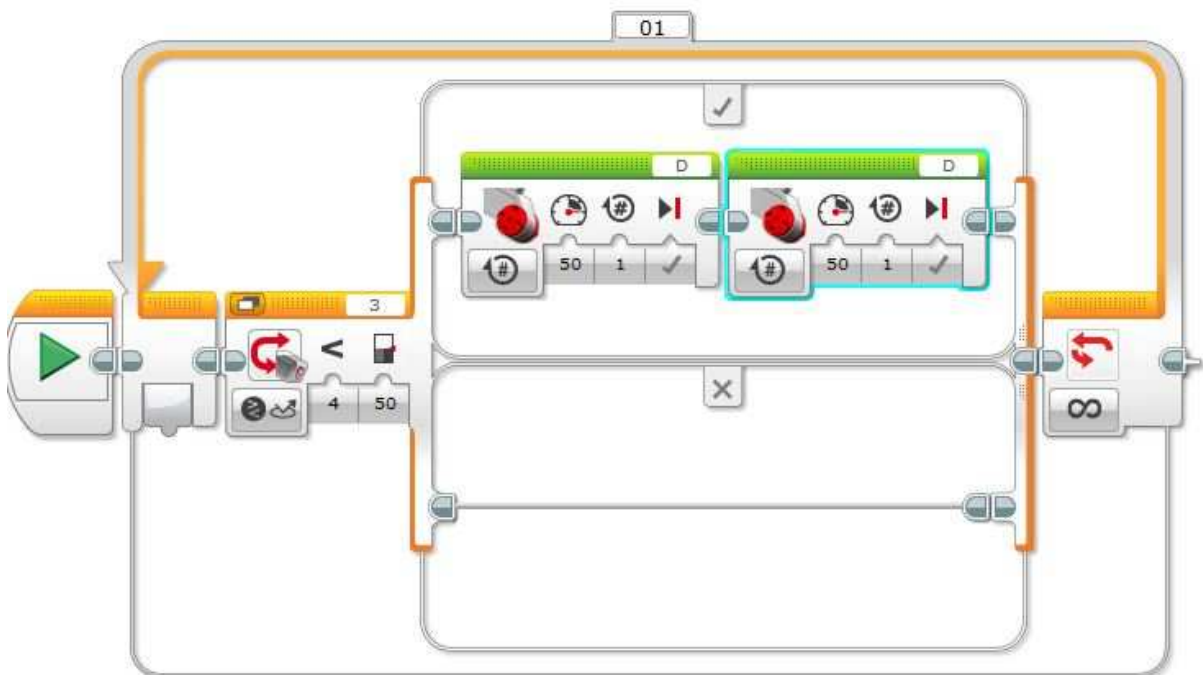


Рис. 3. Добавление блоков

Редактируем блоки. В первом блоке порт соединения «С», а действие выбираем «Выключить» Рис. 4.

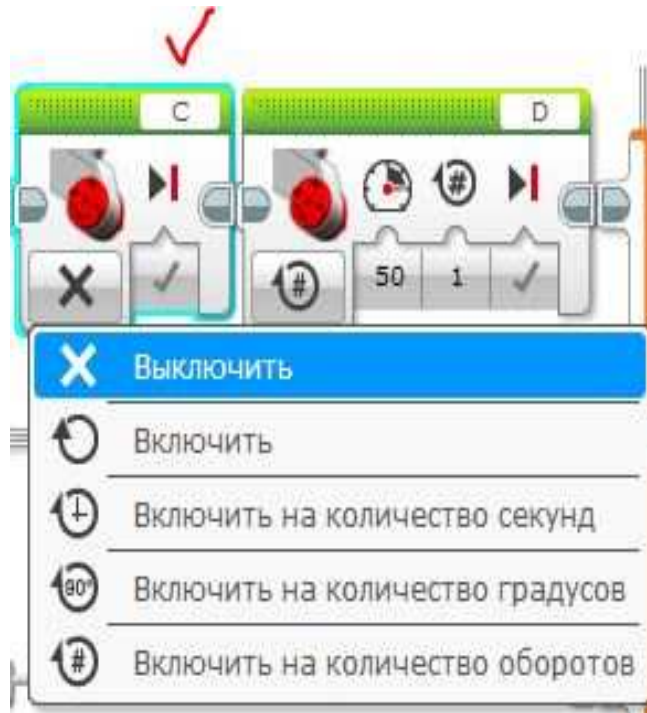


Рис. 4. Редактирование блока

Во втором блоке порт соединения «В», а действие выбираем «Включить» Рис. 5.

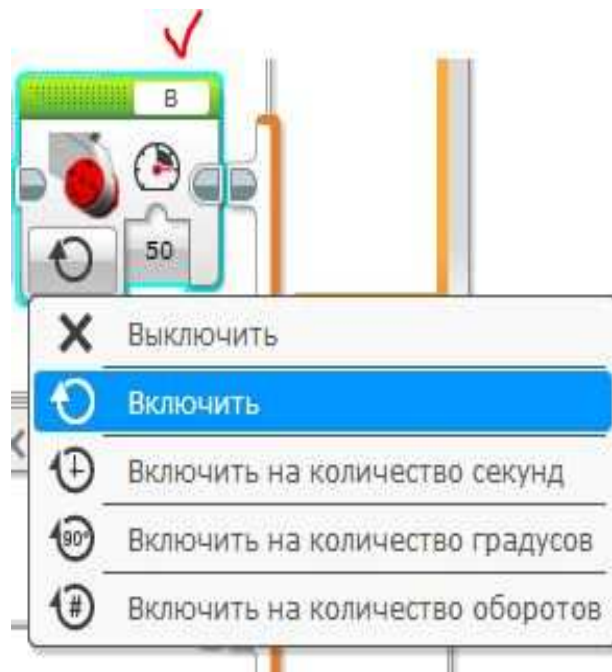



Рис. 5. Редактирование блока

Далее во вкладке «Действие»  выбираем два блока

«Большой мотор»  и располагаем последовательно в нижней части блока «Переключатель» Рис. 6.

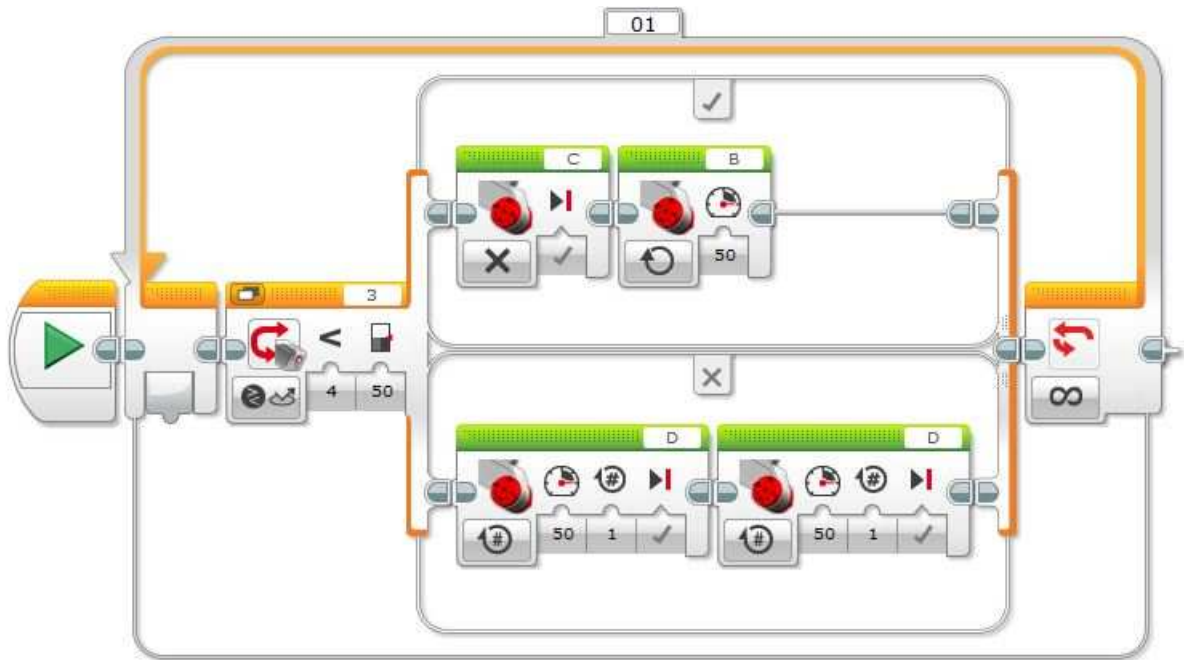


Рис. 6. Добавление новых блоков

Редактируем блоки. В первом блоке порт соединения «В», а действие выбираем «Выключить» Рис. 7.

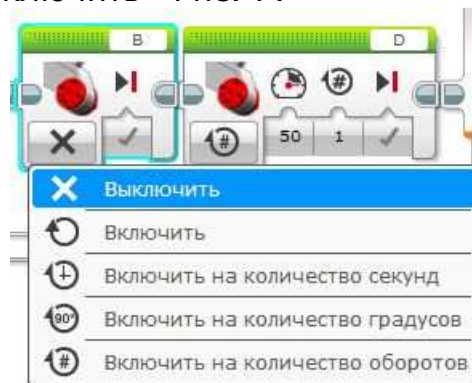


Рис. 7. Редактирование блоков

Во втором блоке порт соединения «С», а действие выбираем «Включить» Рис. 8.

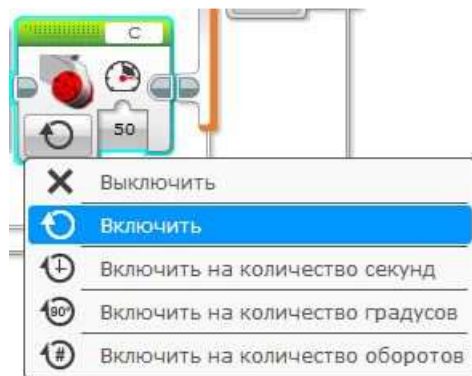


Рис. 8. Редактирование блоков

Программа готова. Общий вид программы представлен на Рис. 9.

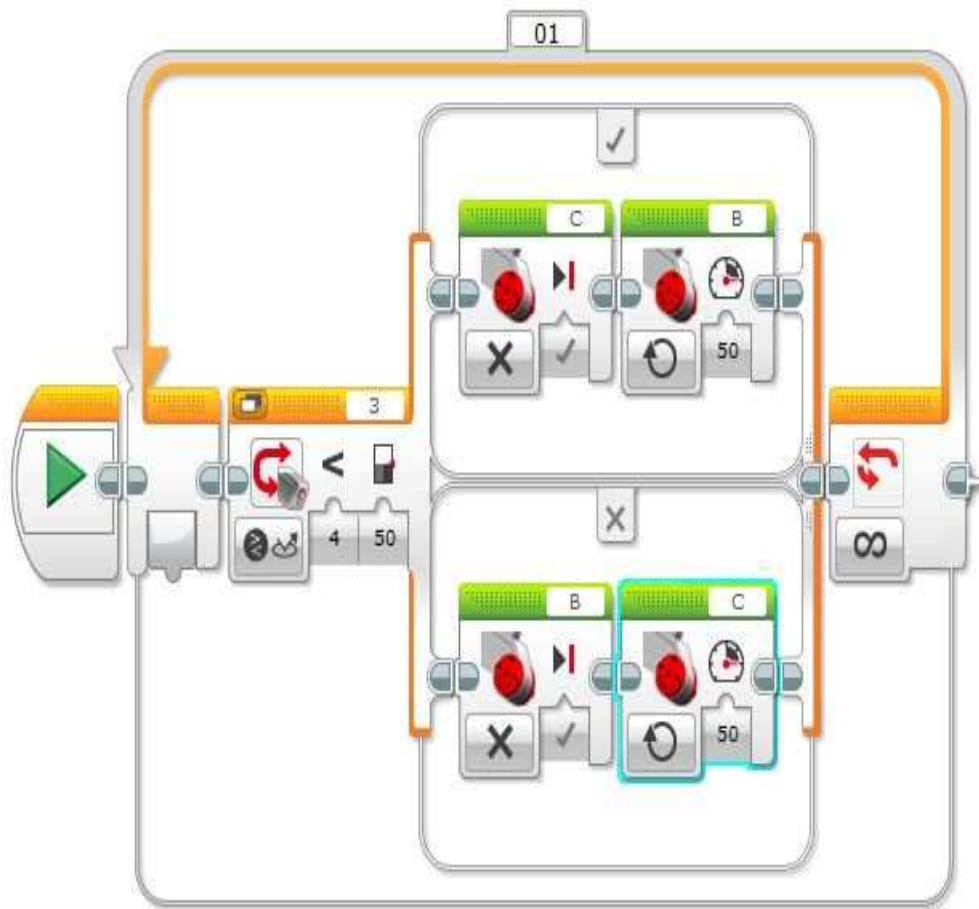


Рис. 9. Готовая программа

Программа №2. Многопозиционный переключатель

Напишем программу, в которой робот будет двигаться согласно измерениям датчика цвета. Цветовое значение датчик будет считывать с цветного квадрата. Робот едет прямо, если нет никаких цветов. Робот поворачивает налево, если перед ним синий цвет. Робот поворачивает направо, если перед ним желтый цвет.

Выполнение.

Создаем проект. Добавляем в программу блок «Цикл». Далее во вкладке «Управление операторами» выбираем блок



«Переключатель» и добавляем его в цикл Рис. 10.

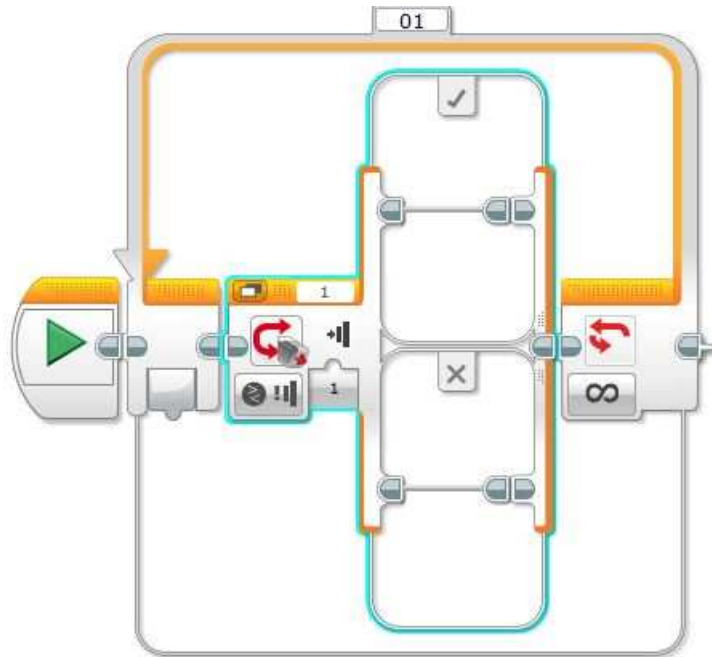


Рис. 10. Добавление блока «Переключатель»

Редактируем блок «Переключатель». Нажимаем на главную кнопку блока и выбираем «Датчик цвета»- «Измерение»- «Цвет» Рис. 11.

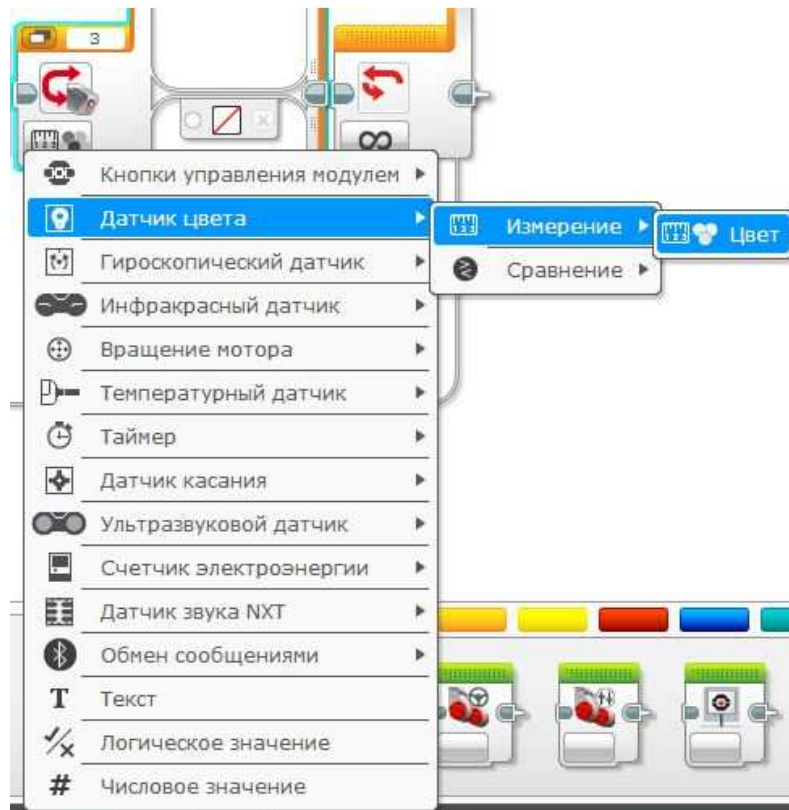




Рис. 11. Редактирование блока

Далее во вкладке «Действие»  выбираем «Независимое управление моторами»  и располагаем в верхней части блока «Переключатель» Рис. 12.

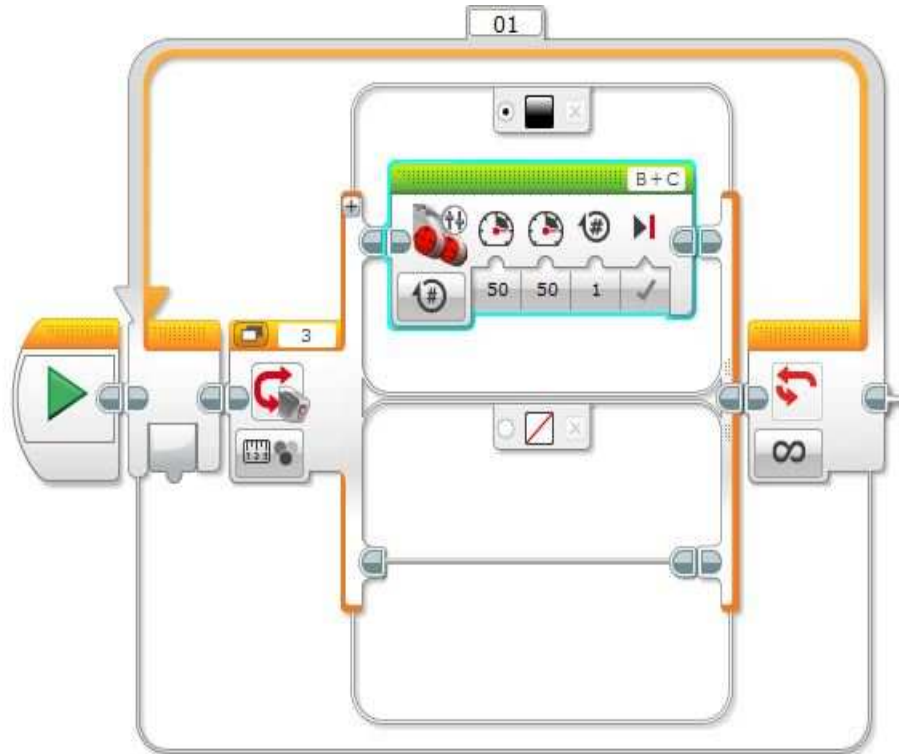


Рис. 12. Добавление блока

Редактируем блок. Выбираем действие моторов- «Включить». Выставляем мощность моторов «В» и «С» 40 Рис. 13.

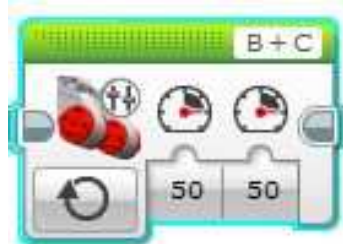






Рис. 13. Редактирование блока

Далее нажимаем на кнопку  и выбираем «Нет цвета» . Далее во вкладке «Действие»  выбираем «Независимое управление моторами»  и располагаем в нижней части блока «Переключатель» Рис. 14.

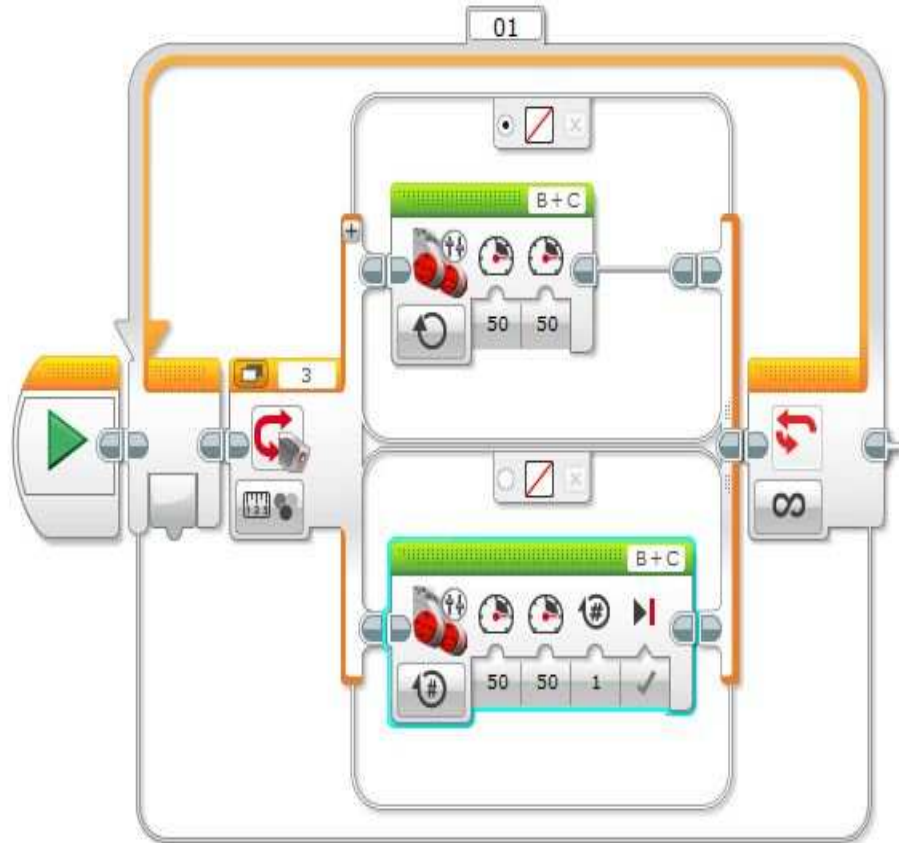



Рис. 14. Добавление блока

Редактируем блок. Выбираем действие моторов- «Включить». Выставляем мощность моторов «В»-10 и «С» 40 Рис. 15.



Рис. 15. Редактирование блока

Далее нажимаем на кнопку  и выбираем цвет «Синий» .

Теперь нам нужно добавить еще одну ячейку для третьего действия. Для этого находим значок «Добавить»  в блоке «Переключатель» и нажимаем на него. После этого у нас появляется еще одна ячейка для блока Рис. 16.

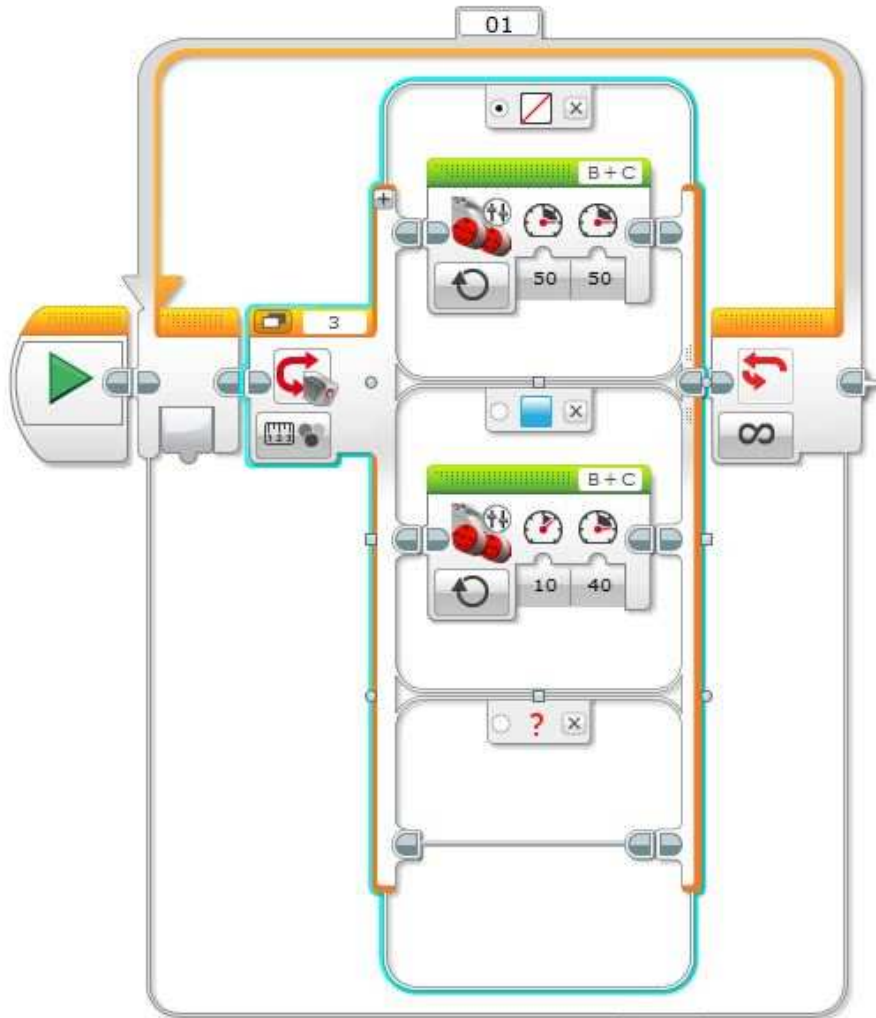




Рис. 16. Добавление новой ячейки в блоке «Переключатель»

Теперь добавляем в эту ячейку блок «Независимое управление моторами». Редактируем блок. Выбираем действие моторов «Включить». Выставляем мощность моторов «В»-40 и «С» 10 Рис. 17.



Рис. 17. Редактирование блока

После этого нажимаем на  и выбираем цвет «Желтый» . Готовая программа изображена на Рис. 18.

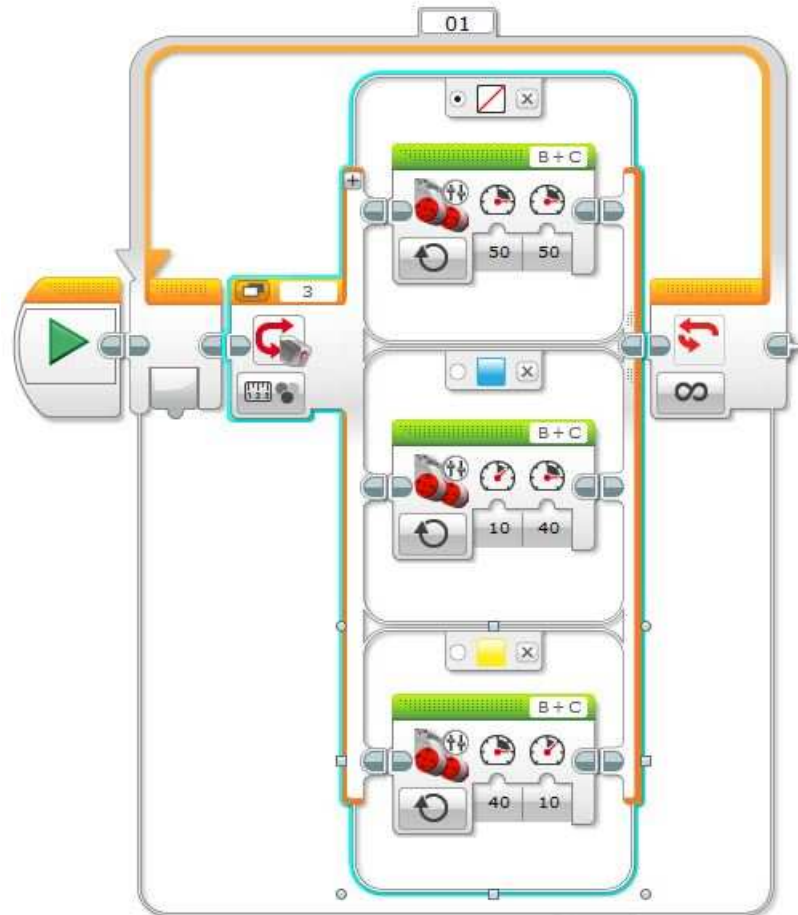


Рис. 18. Готовая программа

Варианты для самостоятельной работы

Вариант №1.

Напишите программу, в которой робот будет двигаться согласно измерениям датчика цвета. Цветовое значение датчик будет считывать с цветного квадрата. Робот едет прямо, если нет никаких цветов. Робот поворачивает налево, если перед ним красный цвет. Робот поворачивает направо, если перед ним зеленый цвет.

Вариант №2.

Напишите программу, в которой робот будет двигаться согласно измерениям датчика цвета. Цветовое значение датчик будет считывать с цветного квадрата. Робот едет прямо, если нет никаких цветов. Робот поворачивает налево, если перед ним желтый цвет. Робот останавливается, если перед ним зеленый цвет.

Вариант №3.

Напишите программу, в которой робот будет двигаться согласно измерениям датчика цвета. Цветовое значение датчик будет считывать с цветного квадрата. Робот едет прямо, если нет никаких цветов. Робот поворачивает налево, если перед ним зеленый цвет. Робот поворачивает направо, если перед ним красный цвет.

Вариант №4.

Напишите программу, в которой робот будет двигаться согласно измерениям датчика цвета. Цветовое значение датчик будет считывать с цветного квадрата. Робот едет прямо, если нет никаких цветов. Робот останавливается, если перед ним красный цвет. Робот поворачивает направо, если перед ним зеленый цвет.

Вариант №5.

Напишите программу, в которой робот будет двигаться согласно измерениям датчика цвета. Цветовое значение датчик будет считывать с цветного квадрата. Робот едет прямо, если нет никаких цветов. Робот поворачивает налево, если перед ним красный цвет. Робот поворачивает направо, если перед ним синий цвет.

Вариант №6.

Напишите программу, в которой робот будет двигаться согласно измерениям датчика цвета. Цветовое значение датчик будет считывать с цветного квадрата. Робот едет прямо, если нет никаких цветов. Робот поворачивает налево, если перед ним желтый цвет. Робот поворачивает направо, если перед ним зеленый цвет.

Задача №2. Программирование графиков

В данном разделе будем использовать среду программирования графиков, чтобы запрограммировать приводную платформу для выполнения действий при сборе данных. Это позволит освоить метод программирования с использованием графиков.

Программа. Перемещение робота относительно предмета

Программа основана на измерениях расстояния до цветного квадрата ультразвуковым датчиком. Робот едет прямо и останавливается за 10 см. от квадрата, потом отъезжает на расстояние до 20 см. Продолжительность эксперимента 10 сек.

Программирование при помощи графиков позволяет запрограммировать робота, без использования некоторых блоков. Вместо блоков есть график, на котором можно указать некоторые конфигурации и условия программы.

Общую программу студенты выполняют в группах по два человека на один робот. Самостоятельная работа выполняется каждым индивидуально.

Выполнение

Создайте и сохраните под своим именем эксперимент. Включите и подключите модуль EV3 к компьютеру посредством USB. Далее настраиваем эксперимент согласно данным, изображенным на Рис. 19.

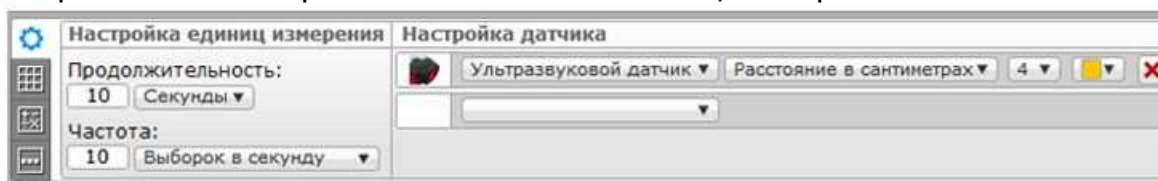


Рис. 19. Настройка конфигураций

Далее вручную установите масштаб по оси Y на максимальное значение 40 Рис. 20.

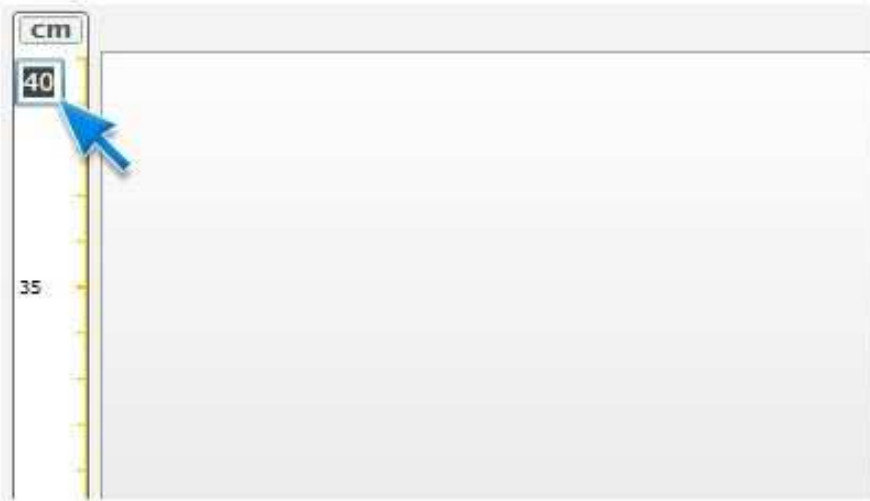


Рис. 20. Определение масштаба

Перейдите к режиму программирования графиков Рис. 21.

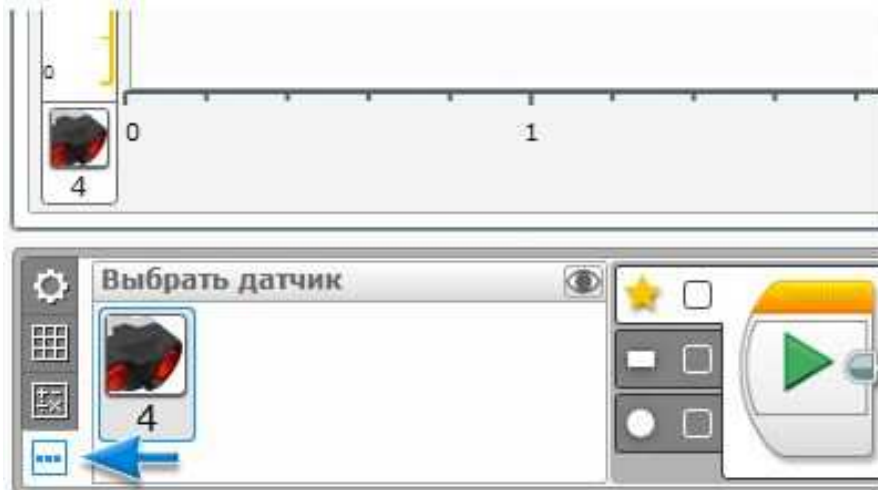


Рис. 21. Режим программирования графиков

Установите флажки в зоне звездочки и прямоугольника, чтобы отобразить пороговые линии в области графика Рис. 22.

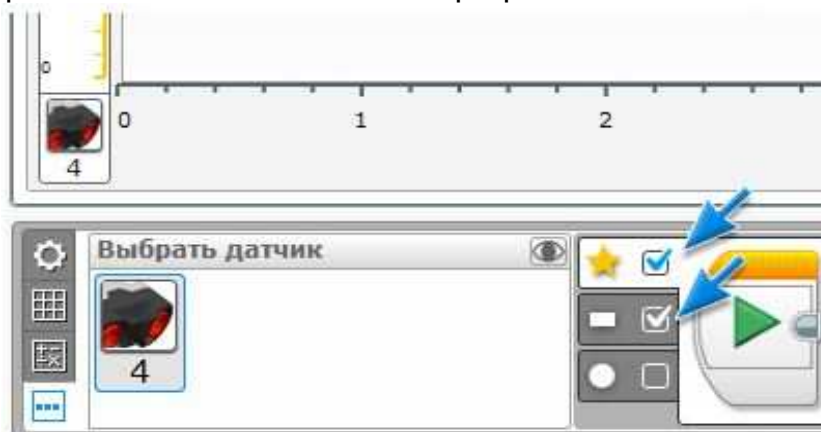


Рис. 22. Отображение пороговых линий графика

Перетащите первую пороговую линию на 20 см. Рис. 23. Также значение можно ввести вручную.

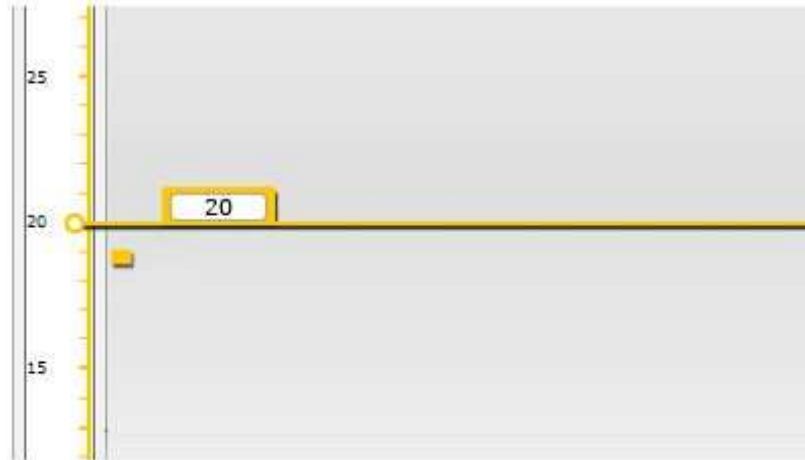


Рис. 23. Установка пороговой линии

Установите флажок в зоне круга, чтобы добавить вторую пороговую линию Рис. 24.

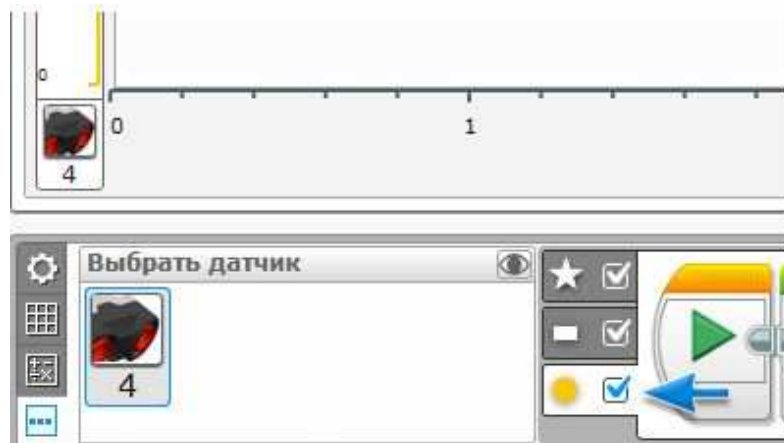


Рис. 24. Добавляем вторую пороговую линию

Установите вторую пороговую линию на 10 см. Рис. 25.

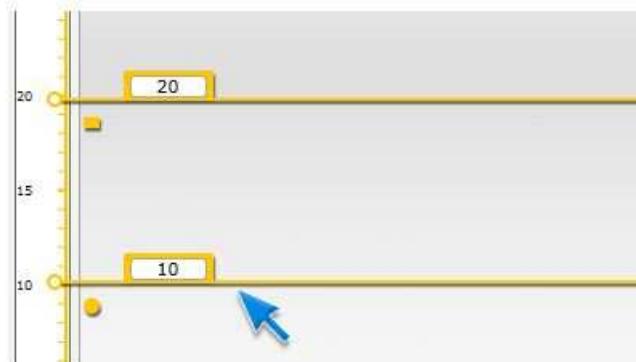


Рис. 25. Установка второй пороговой линии

Щелкните на зоне звездочки и откройте палитру программирования графика Рис. 26.



Рис. 26. Палитра программирования графика

Создаем программу, изображенную на Рис. 27.



Рис. 27. создание программы

Программа состоит из блоков «Рулевое управление» и «Индикатор состояния модуля».

Щелкните на зоне прямоугольника и снова откройте палитру программирования графика Рис. 28.



Рис. 28. Палитра программирования графика

Заново создайте программу, показанную на Рис. 29.



Рис. 29. Создание программы

Теперь щелкните на зоне круга и снова откройте палитру программирования графика Рис. 30.

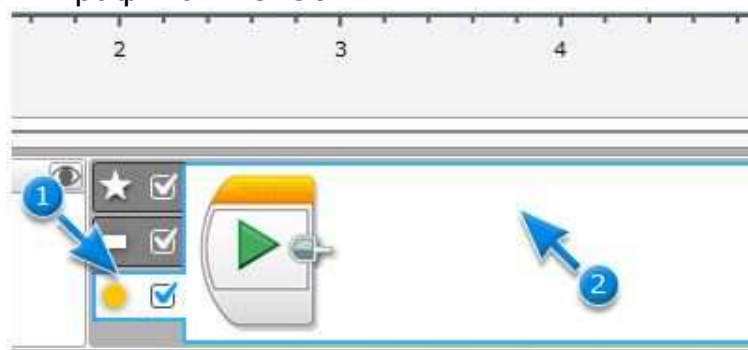


Рис. 30. Палитра программирования графика

Заново создайте, показанную на Рис. 31 программу.



Рис. 31. Создание программы

После этого мы указали все, что должна выполнять приводная платформа в каждой зоне. Установите робот на расстоянии примерно 40 см. от цветного квадрата. Загрузите и запустите программу Рис. 31.

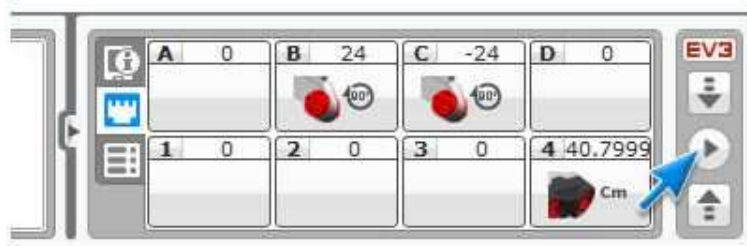


Рис. 31. Загрузка и запуск программы

Варианты для самостоятельной работы

Вариант №1.

Напишите программу, в которой робот едет прямо и останавливается за 20 см. от квадрата, потом отъезжает на расстояние до 30 см. Продолжительность эксперимента 10 сек.

Вариант №2.

Напишите программу, в которой робот едет прямо и останавливается за 20 см. от квадрата, потом отъезжает на расстояние до 30 см. Продолжительность эксперимента 9 сек.

Вариант №3.

Напишите программу, в которой робот едет прямо и останавливается за 15 см. от квадрата, потом отъезжает на расстояние до 30 см. Продолжительность эксперимента 15 сек.

Вариант №4.

Напишите программу, в которой робот едет прямо и останавливается за 5 см. от квадрата, потом отъезжает на расстояние до 15 см. Продолжительность эксперимента 10 сек.

Вариант №5.

Напишите программу, в которой робот едет прямо и останавливается за 10 см. от квадрата, потом отъезжает на расстояние до 25 см. Продолжительность эксперимента 8 сек.

Вариант №6.

Напишите программу, в которой робот едет прямо и останавливается за 15 см. от квадрата, потом отъезжает на расстояние до 30 см. Продолжительность эксперимента 10 сек.

Контрольные вопросы

1. Что такое яркость?
2. Для чего используется переключатель?
3. Для чего используется многопозиционный переключатель?
4. Для чего применяется метод программирования графиков?
5. Что такое блок «Большой мотор»?
6. Что такое блок «Независимое управление моторами»?
7. Что такое блок «Переключатель»?
8. Как происходит программирование робота при помощи графиков?

Лабораторная работа № 6

Сборка робота- манипулятора

Краткие теоретические сведения

Промышленный робот- это манипуляционный робот, который предназначен для выполнения двигательных и управляющих функций в производственном процессе. Другими словами, это автоматическое устройство, состоящее из манипулятора и перепрограммируемого устройства управления, которое формирует управляющие воздействия, задающие требуемые движения исполнительных органов манипулятора. Применяется для перемещения предметов производства и выполнения различных технологических операций.

Манипулятор — это механизм для управления пространственным положением орудий и объектов труда.

Манипуляторы, как правило, включают в себя подвижные звенья двух типов:

- звенья, обеспечивающие поступательные движения;
- звенья, обеспечивающие вращательные перемещения.

Сочетание и взаимное расположение звеньев определяет степень подвижности, а также область действия манипуляционной системы робота.

Управление

В развитии систем управления промышленных роботов можно проследить два направления. Одно из них берёт своё начало от систем программного управления станками и преобразовалось в создание автоматически управляемых промышленных манипуляторов. Второе привело к появлению полуавтоматических биотехнических и интерактивных систем, в которых в управлении действиями промышленного робота участвует человек-оператор.

Таким образом, промышленные роботы можно подразделить на следующие три типа:

- **Автоматические роботы:**
 - *Программные роботы* (роботы с программным управлением) — простейшая разновидность автоматически управляемых промышленных роботов, до сих пор широко используемых в силу их дешевизны на различных промышленных предприятиях для обслуживания несложных технологических процессов. В таких роботах отсутствует сенсорная часть, а все действия выполняются циклически по жёсткой программе, заложенной в память запоминающего устройства.

- *Адаптивные роботы* (роботы с адаптивным управлением) — роботы, оснащённые *сенсорной частью* (системой очувствления) и снабжённые набором программ. Сигналы, поступающие к системе управления от датчиков, анализируются ею, и в зависимости от результатов принимается решение о дальнейших действиях робота, предполагающее переход от одной программы к другой (смена технологической операции). Аппаратное и программное обеспечение — в принципе то же, что и в предыдущем случае, но к его возможностям предъявляются повышенные требования.

- *Обучаемые роботы* — роботы, действия которых полностью формируются в ходе обучения (человек при помощи специальной платы задаёт порядок действий робота, и этот порядок действий записывается в память запоминающего устройства).

- *Интеллектуальные роботы* (роботы с элементами искусственного интеллекта) — роботы, способные с помощью сенсорных устройств самостоятельно воспринимать и распознавать обстановку, строить модель среды, и автоматически принимать решение о дальнейших действиях, а также самообучаться по мере накопления собственного опыта деятельности.

- **Биотехнические роботы:**

- *Командные роботы* (роботы с командным управлением) — манипуляторы, в которых человек-оператор дистанционно задаёт с командного устройства движение в каждом сочленении (строго говоря, это — не роботы в полном смысле слова, а «полуроботы»).

- *Копирующие роботы* (роботы с копирующим управлением) — манипуляторы, копирующие действия приводимого в движение оператором задающего устройства, кинематически подобного исполнительному механизму манипулятора.

- *Полуавтоматические роботы* — роботы, при управлении которыми человек-оператор задаёт лишь движение рабочего органа манипулятора, а формирование согласованных движений в сочленениях система управления роботов осуществляет самостоятельно.

- **Интерактивные роботы:**

- *Автоматизированные роботы* (роботы с автоматизированным управлением) — роботы, чередующие автоматические режимы управления с биотехническими.

- *Супервизорные роботы* (роботы с супервизорным управлением) — роботы, выполняющие автоматически все этапы заданного цикла операций, но осуществляющие переход от одного этапа к другому по команде человека-оператора.

- *Диалоговые роботы* (роботы с диалоговым управлением) — автоматические роботы (любой разновидности), способные взаимодействовать с человеком-оператором, используя язык того или иного уровня (включая подачу текстовых или голосовых команд и ответные сообщения робота).

Большинство современных роботов функционирует на основе принципов обратной связи, подчинённого управления и иерархичности системы управления роботом.

Задача. Сборка робота-манипулятора

В данной работе мы спроектируем робота, который будет перемещать предметы и располагать их в заданных местах Рис. 1.



Рис. 1. Робот- манипулятор

Робот состоит из контроллера, моторов, датчика цвета и датчика касания. Цель проекта заключается в проектировании робота, который будет находить в пространстве предметы и перемещать их в заданные места. Такой робот широко используется в производстве.

Сборку робота студенты выполняют, разбившись на группы по два человека.

Выполнение.

Открываем программу LEGO MINDSTORMS Education EV3. Выбираем вкладку «Основной набор модели» Рис. 2.



Рис. 2. Вкладка «Основной набор модели»

Далее выбираем вкладку «Инструкции к модели» и находим справа в списке «Рука робота» Рис. 3.



Рис. 3. Вкладка «Рука робота»

И нажимаем на «Открыть». После этого нам открывается программа, где в правом верхнем углу есть вспомогательное окно, которое имеет три страницы Рис. 4.

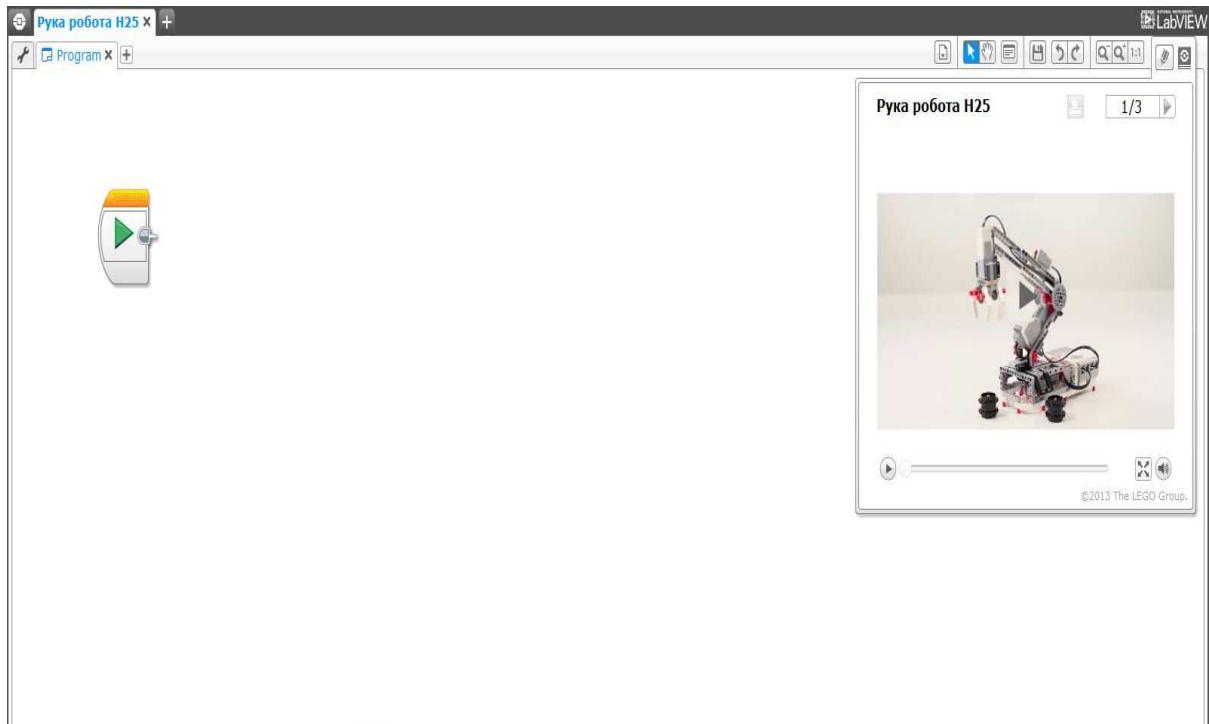


Рис. 4. Окно программы

На первой странице находится видео работы готового робота. Нажимаем на кнопку «Следующая страница» и выбираем вторую страницу. На ней содержится руководство по сборке робота Рис. 5.

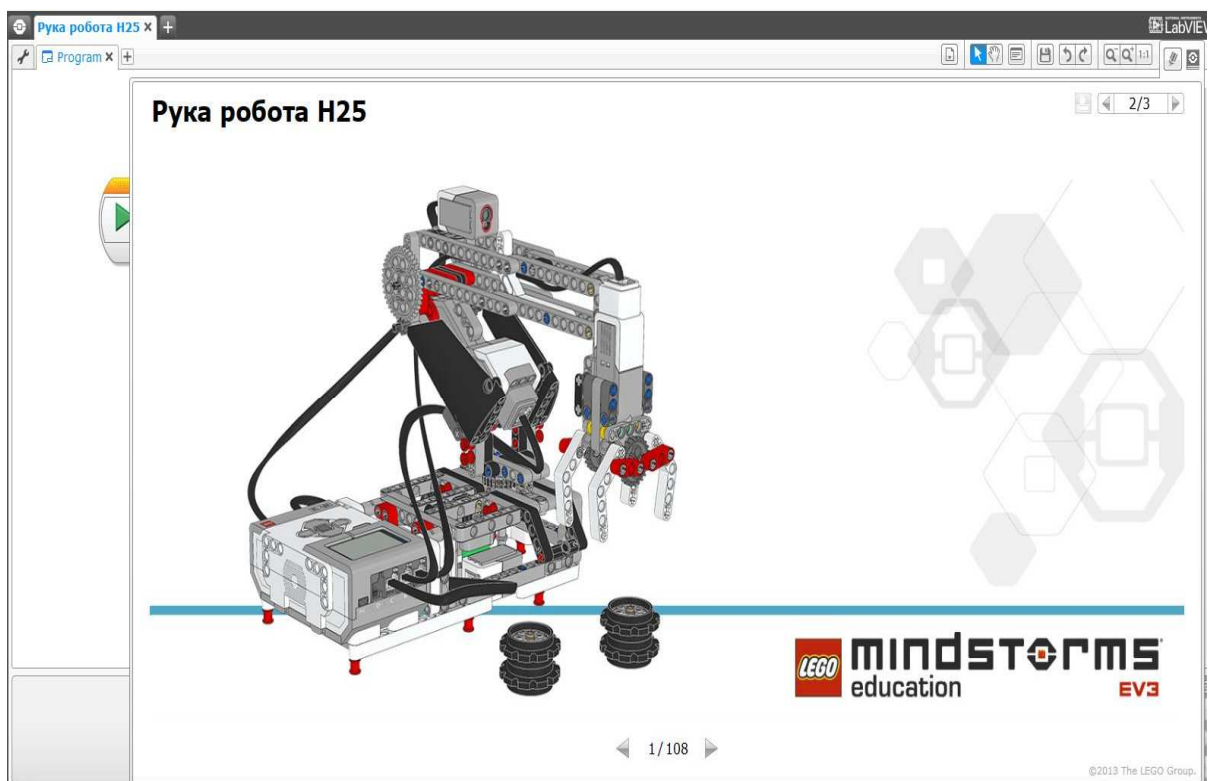


Рис. 5. Руководство по сборке

Открываем руководство и начинаем сборку робота, следуя всем указаниям.

Контрольные вопросы


1. Что такое промышленный робот?
2. Что такое манипулятор и для чего он используется?
3. Назовите основные три типа роботов с точки зрения управления.
4. Перечислите, какие бывают автоматические роботы?
5. Что такое биотехнические роботы? Перечислите виды биотехнических роботов.
6. Что такое интерактивные роботы?
7. Перечислите виды биотехнических роботов.

Лабораторная работа № 7


Программирование робота- манипулятора

Краткие теоретические сведения




Блок «Средний мотор» () - блок «Средний мотор» управляет средним мотором. Вы можете включать или выключать мотор, управлять его уровнем мощности или включать мотор на определенное количество времени или оборотов.




Блок «Индикатор состояния модуля» () - управляет индикатором состояния модуля. Индикатор состояния модуля находится вокруг кнопок управления модулем на лицевой панели модуля EV3. Вы можете сделать индикатор состояния модуля зеленым, оранжевым или красным, выключить его или заставить его мигать.



Блок «Звук» () - блок звука издает звук, используя динамик в модуле EV3. Вы можете проигрывать записанные звуковые файлы или указать музыкальную ноту или тон.



Блок «INI» () - данный блок позволяет создавать программу, которую можно использовать в качестве дополнительной программной части в других программах.

Задача. Программирование манипулятора

Целью данного раздела является приобретение навыка создания более сложных программ с большим количеством компонентов. Мы запрограммируем манипулятор выполнять стандартные для себя действия: перемещение предметов в пространстве Рис. 1.

Это позволит понять принцип работы крупных промышленных роботов- аналогов, а также, способ их программирования и управления.

Общую программу студенты выполняют в группах по два человека на один робот. Самостоятельная работа выполняется каждым индивидуально.



Рис. 1. Робот- манипулятор

Программа. Перемещение предметов

Программа заключается в перемещении предметов в заданную позицию. Справа и слева от платформы робота лежит по одному предмету. Робот дожидается команды оператора, которая заключается в действии нажатия на кнопку: либо «Вверх» - предмет справа, либо «Вниз» - предмет слева. В зависимости от того, какая кнопка будет нажата оператором, робот захватывает предмет и располагает перед собой. Каждое нажатие кнопки оператором сопровождается звуковым сигналом и соответствующим изображением на экране контроллера.

Выполнение.

Открываем программу LEGO MINDSTORMS Education EV3. Выбираем вкладку «Основной набор модели» Рис. 2.



Рис. 2. Вкладка «Основной набор модели»

Далее выбираем вкладку «Инструкции к модели» и находим справа в списке «Рука робота» Рис. 3.



Рис. 3. Вкладка «Рука робота»

И нажимаем на «Открыть». После этого нам открывается программа, где в правом верхнем углу есть вспомогательное окно, которое имеет три страницы Рис. 4.

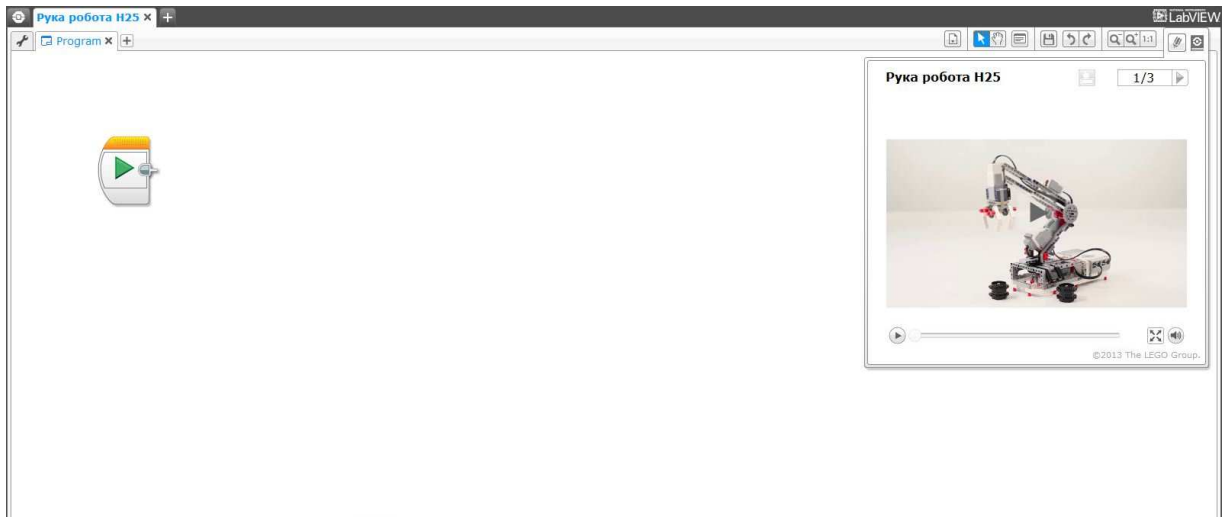






Рис. 4. Окно программы

Закрываем вспомогательное окно и начинаем программирование.

Заходим в «Мои блоки»  и выбираем блок «INI»  и добавляем в программу. Открываем вкладку «Управление

операторами» , и выбираем блок «Цикл»  и добавляем к блоку «INI» Рис. 5.

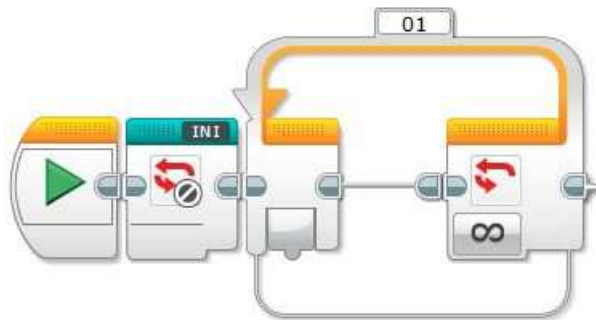




Рис. 5. Цикл

После этого во вкладке «Действие»  выбираем блок «Экран»  и добавляем его в начало цикла и редактируем блок. Нажимаем на строку в правом верхнем углу блока и выбираем из списка «Question mark» Рис. 6.

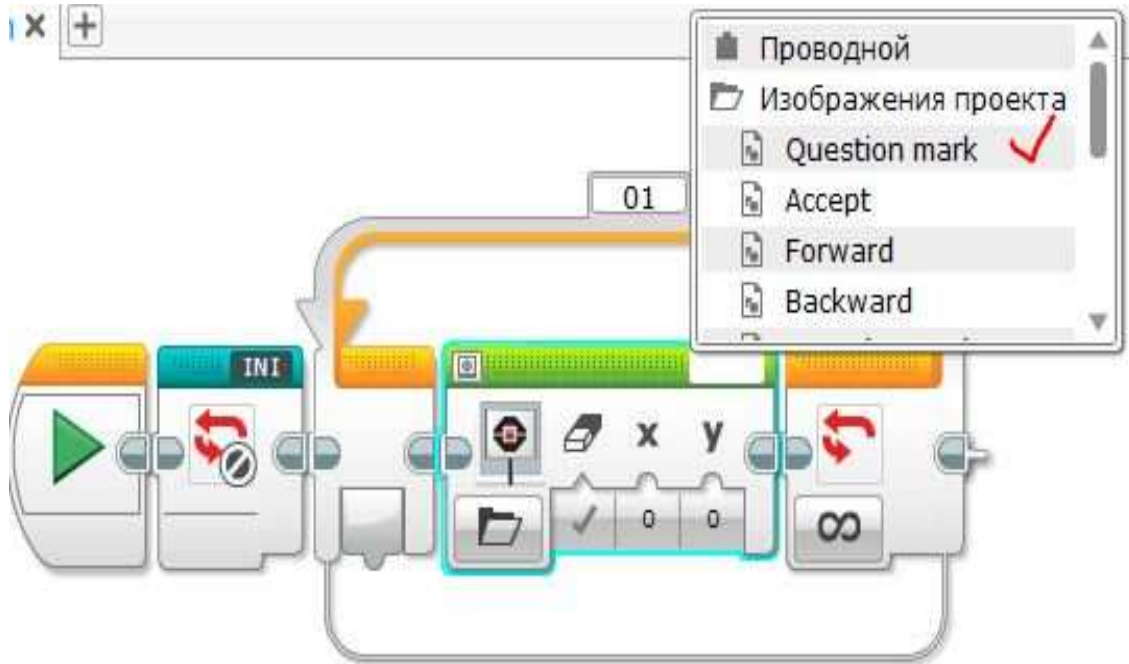




Рис. 6. Редактирование блока

Во вкладке «Действие»  выбираем блок «Индикатор состояния модуля»  и добавляем к блоку «Экран» Рис. 7.

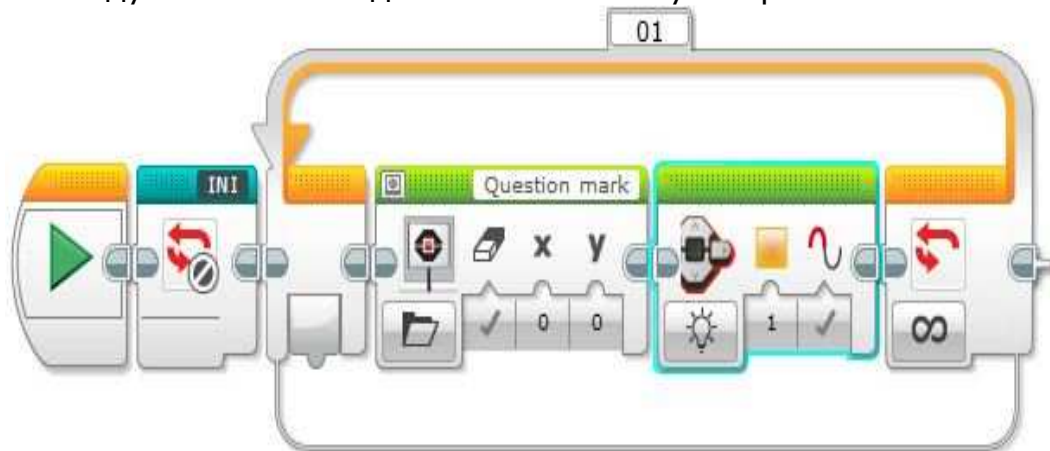




Рис. 7. Добавление блока «Индикатор состояния модуля»

Открываем вкладку «Управление операторами»  и выбираем блок «Ожидание»  и добавляем к блоку «Индикатор состояния модуля». Редактируем блок. Нажимаем на кнопку «Выбор режима» и выбираем «Кнопки управления модулем» - «Сравнение»- «Кнопки управления модулем» Рис. 8.

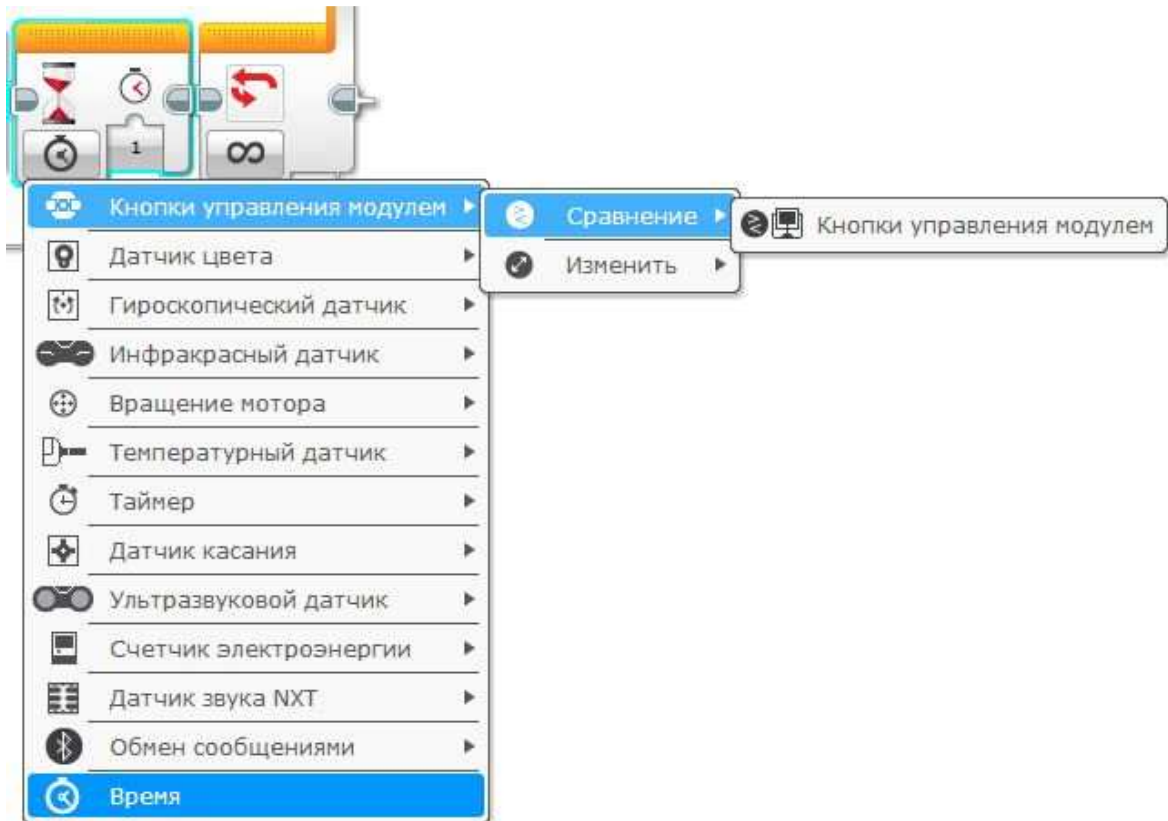


Рис. 8. Редактирование блока

Далее нажимаем на «Набор идентификаторов кнопок модуля» и выбираем кнопки 4 и 5 Рис. 9.

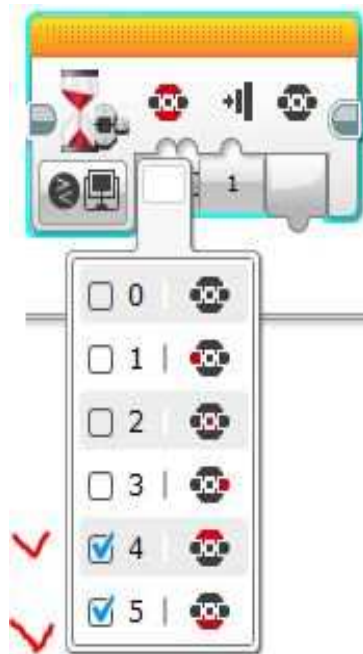



Рис. 9. Выбор кнопок реагирования

Кнопка «Состояние» остается со значением «1». Во вкладке

«Действие»  выбираем блок «Индикатор состояния модуля» и добавляем к блоку «Ожидание». Редактируем блок. Нажимаем на кнопку «Выбор режима» и выбираем режим «Сброс» Рис. 10.

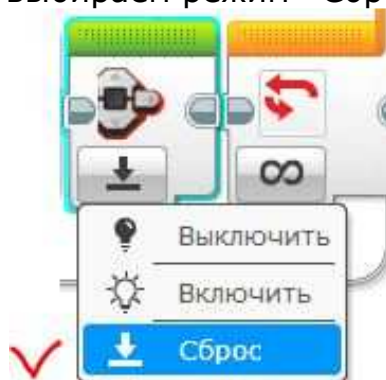




Рис. 10. Выбор режима работы блока

Во вкладке «Действие»  выбираем блок «Звук»  и добавляем к блоку «Индикатор состояния модуля». Редактируем блок. Нажимаем на строку и из списка выбираем звук «Air release» Рис. 11.

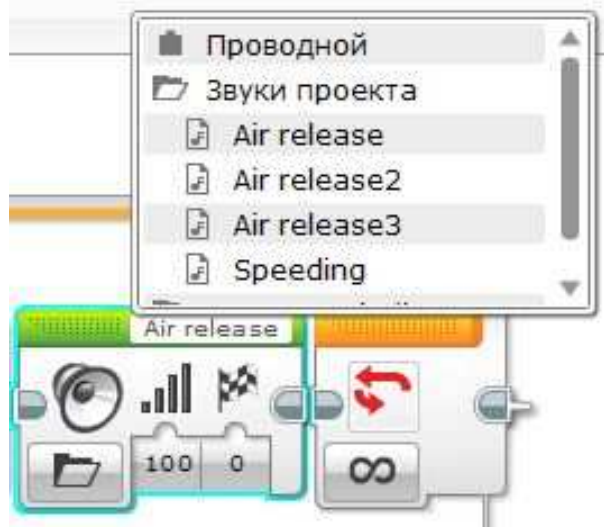


Рис. 11. Выбор звукового сигнала

После этого во вкладке «Действие»  выбираем блок «Экран»



и добавляем его к блоку «Звук». Редактируем блок. Нажимаем на строку и из списка выбираем изображение «Асцепт» Рис. 12.

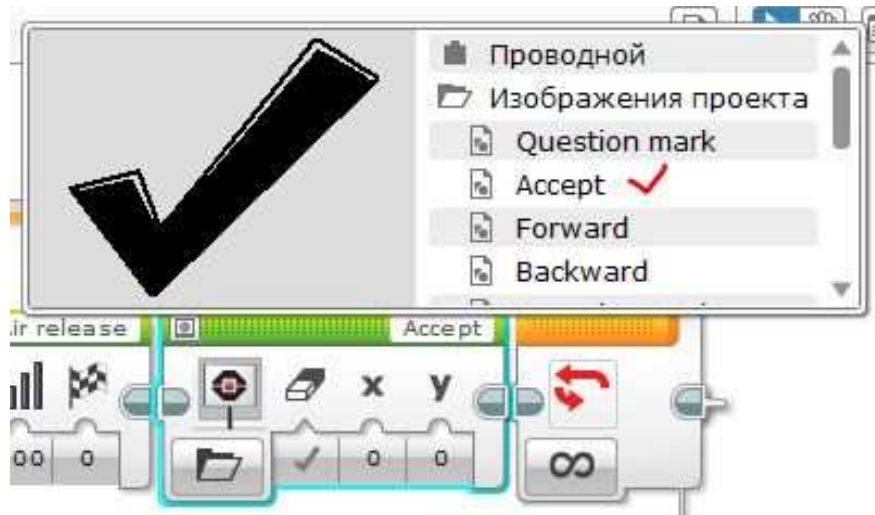




Рис. 12. Выбираем тип изображения

Открываем вкладку «Управление операторами»  и выбираем

блок «Ожидание»  и добавляем к блоку «Экран». Редактируем блок. Нажимаем на кнопку «Выбор режима» и выбираем «Кнопки управления модулем»- «Сравнение»- «Кнопки управления модулем» Рис. 13.

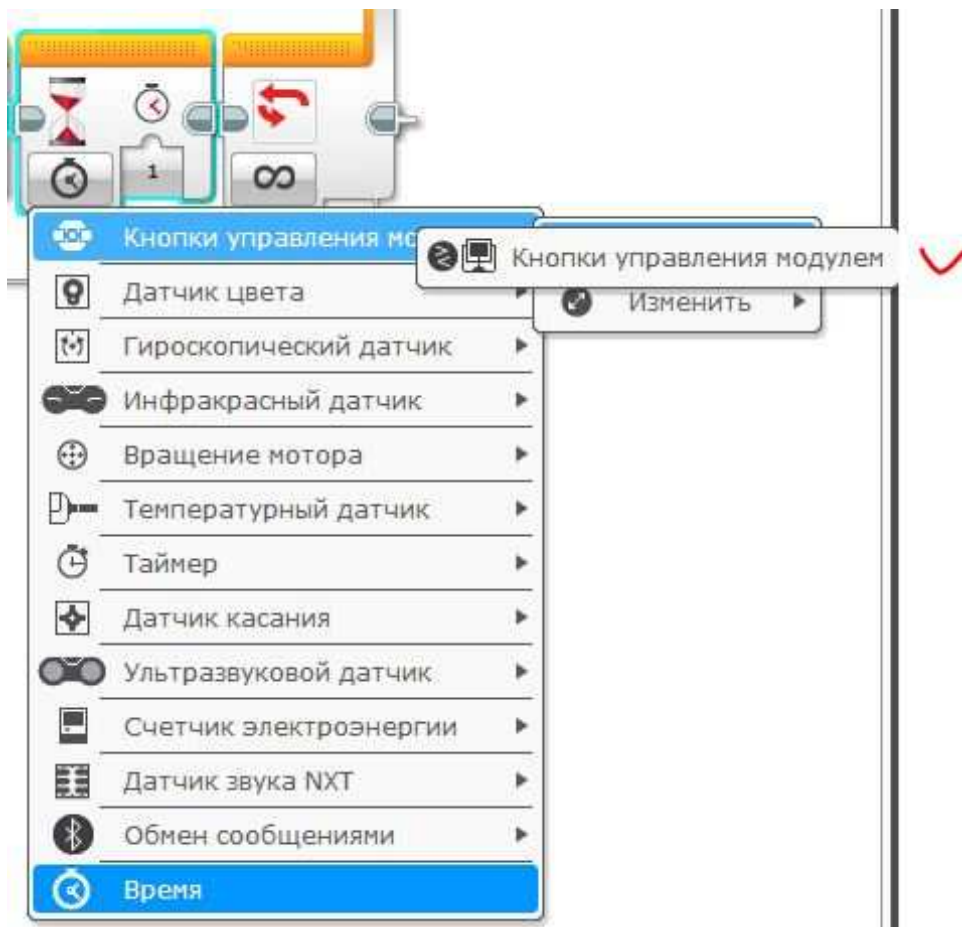


Рис. 13. Редактирование блока

Далее нажимаем на «Набор идентификаторов кнопок модуля» и выбираем кнопку 2 Рис. 14.

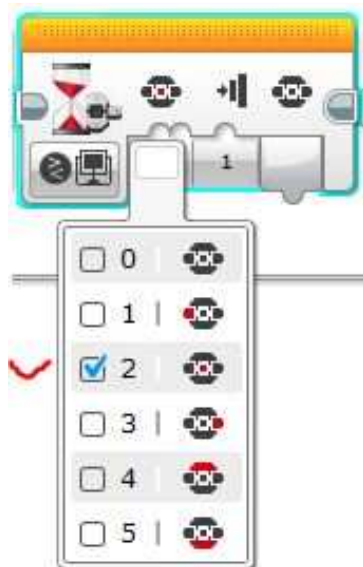


Рис. 14. Выбор кнопок реагирования

Кнопка «Состояние» остается со значением «1». Далее во вкладке «Управление операторами» выбираем блок



«Переключатель» и добавляем его в цикл к последнему блоку «Ожидание» Рис. 15.

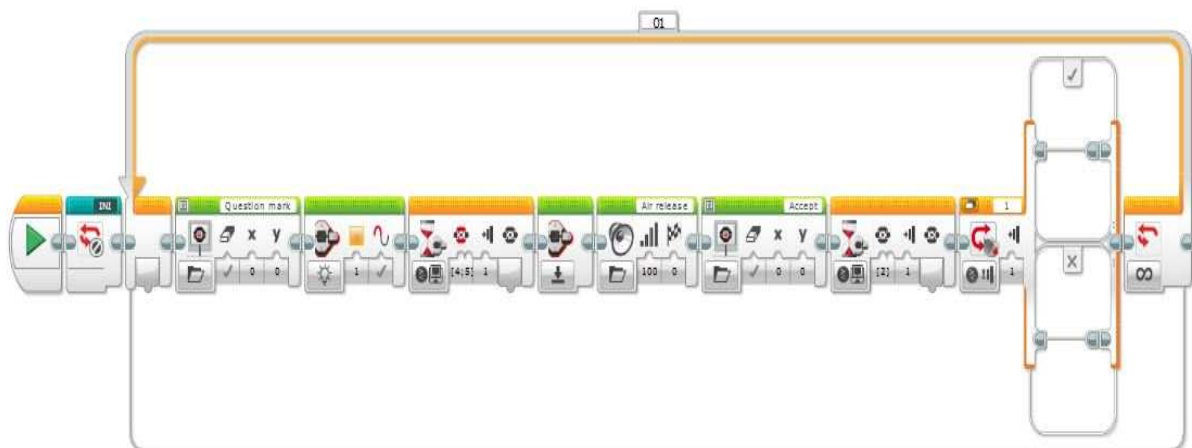


Рис. 15. Добавляем блок «Переключатель» в программу

Далее редактируем блок «Переключатель». Нажимаем на кнопку «Режим» и выбираем «Числовое значение» Рис. 16.

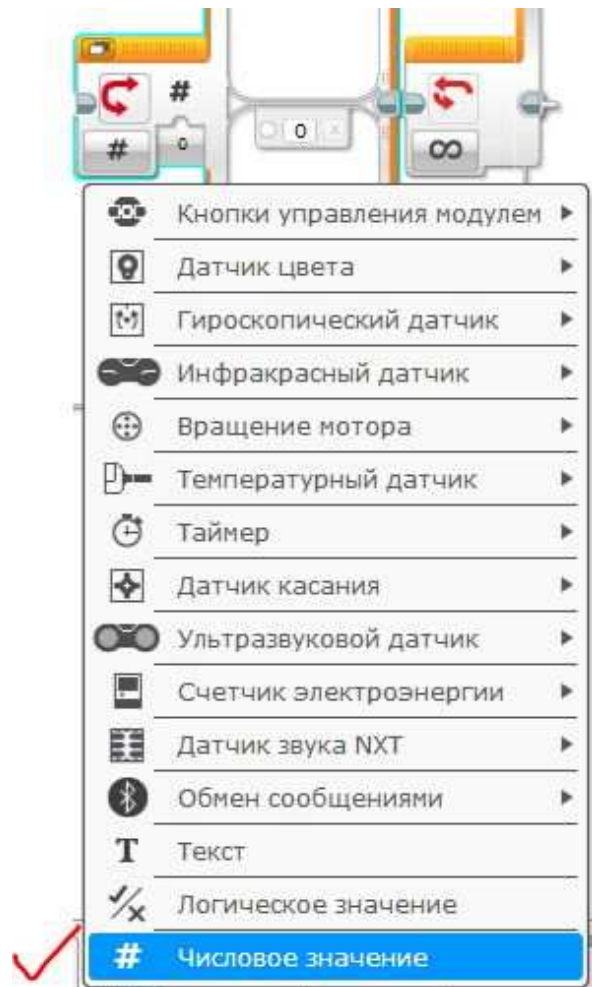


Рис. 16. Редактирование блока

После этого в верхнюю ячейку добавляем блоки «Экран» и «Большой мотор» Рис. 17.

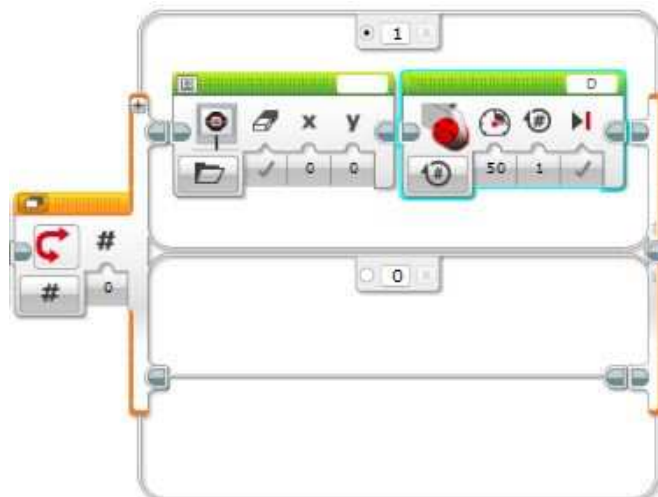


Рис. 17. Редактирование блока «Переключатель»

Редактируем блок «Экран». Выбираем изображение «Forward» Рис. 18.

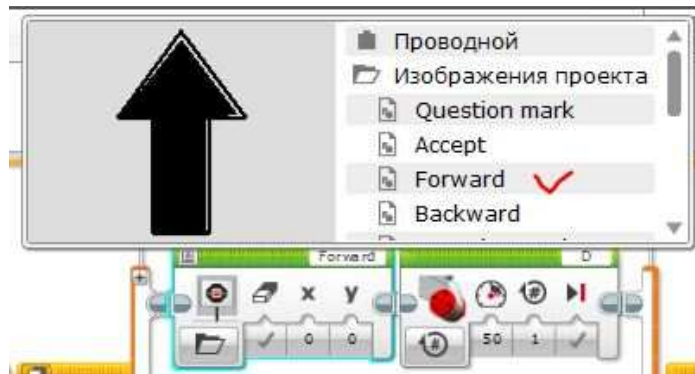


Рис. 18. Выбираем тип изображения

Далее редактируем блок большого мотора. Нажимаем на «Режим работы» и выбираем «Включить на количество оборотов» Рис. 19.

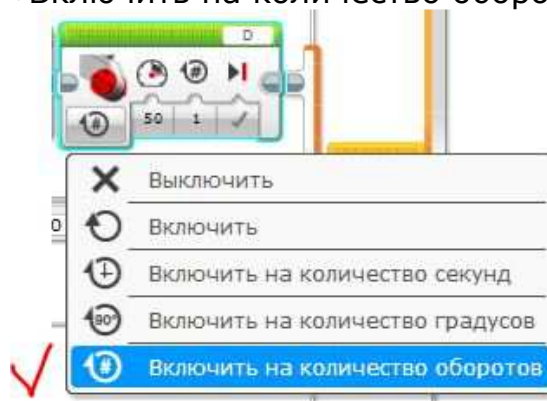


Рис. 19. Выбираем режим работы

Далее «Мощность» ставим 65, а количество оборотов 0,865. Порт соединения «С» Рис. 20.

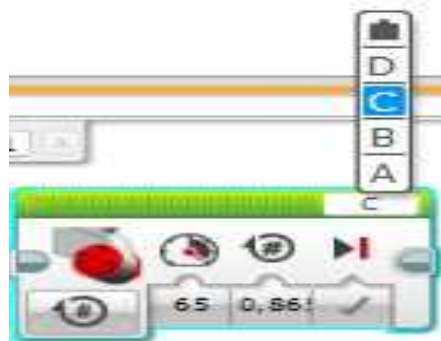


Рис. 20. Выбираем порт соединения

Далее в верхней строке ячейки ставим «4» Рис. 21.



Рис. 21. Редактирование блока

После этого в нижнюю ячейку добавляем блоки «Экран» и «Большой мотор» Рис. 22.

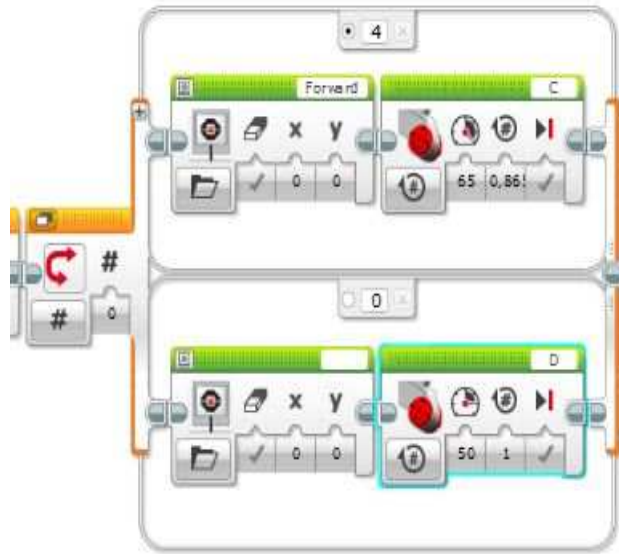


Рис. 22. Добавление блоков в нижнюю ячейку блока «Переключатель»

Редактируем блок «Экран». Нажимаем на строку и выбираем изображение «Backward» Рис. 23.

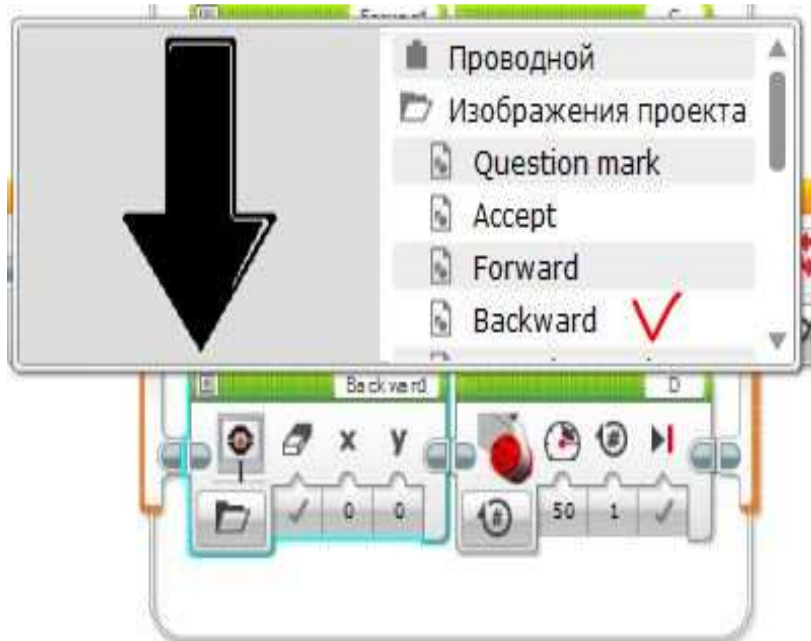


Рис. 23. Выбираем тип изображения

Редактируем блок «Большой мотор». Нажимаем на «Режим работы» и выбираем «Включить на количество оборотов». Далее «Мощность» ставим -65, а количество оборотов 0,85. Порт соединения «С» Рис. 24.

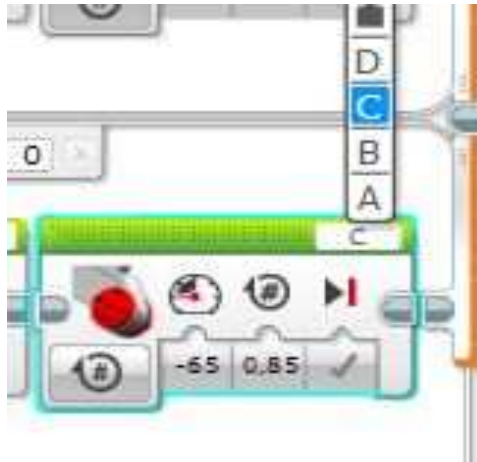


Рис. 24. Настройка блока большого мотора

Далее в верхней строке ячейки ставим «5» Рис. 25.

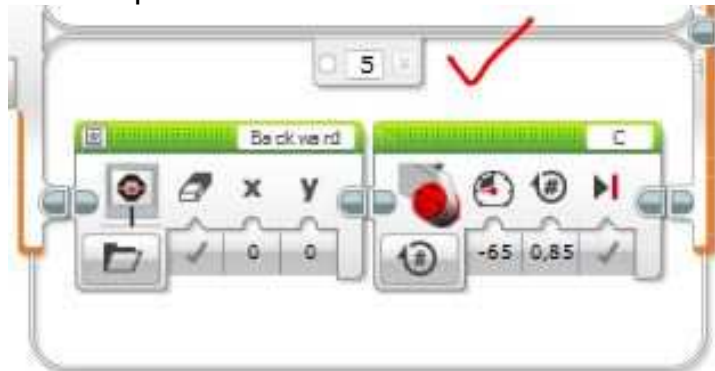


Рис. 25. Редактирование нижней ячейки блока

После этого нам необходимо соединить блок «Ожидание», отмеченный на Рис. 26 цифрой «1» и блок «Переключатель», отмеченный цифрой «2».

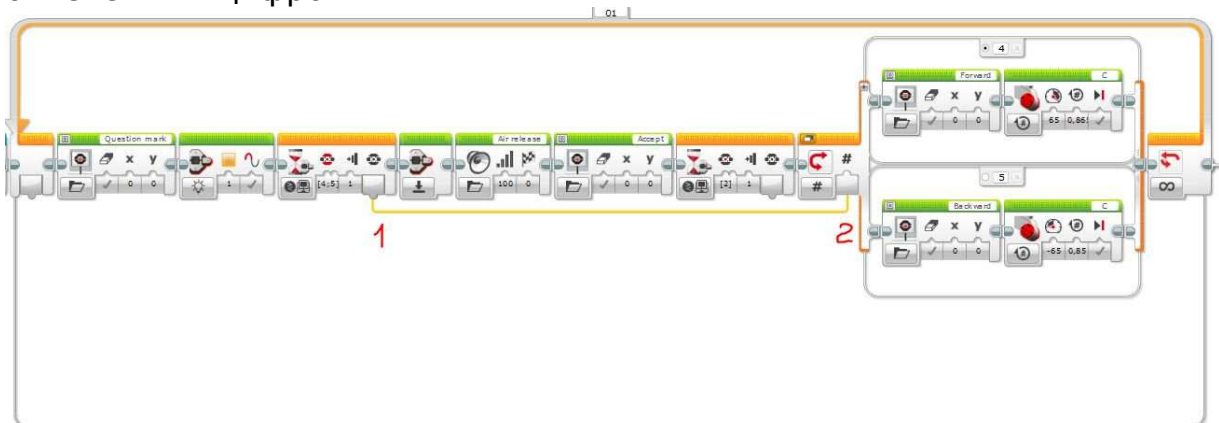


Рис. 26. Устанавливаем соединение между блоками

Далее выбираем блок «Большой мотор» и добавляем его в цикл к блоку «Переключатель» Рис. 27.

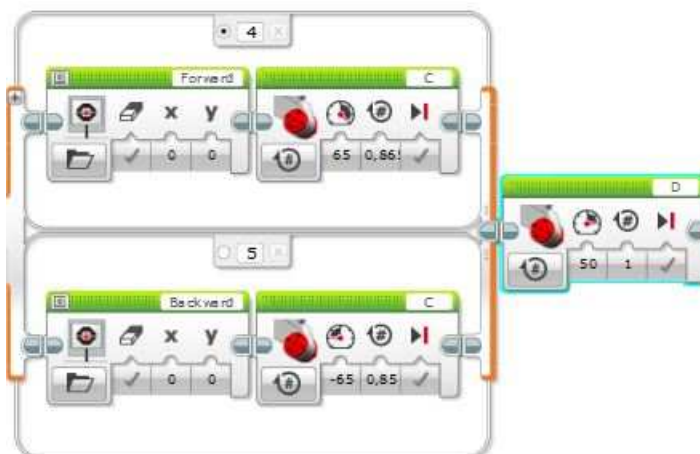


Рис. 27. Добавления блока большого мотора

Редактируем блок. Режим работы- «Включить на количество градусов» Рис. 28.

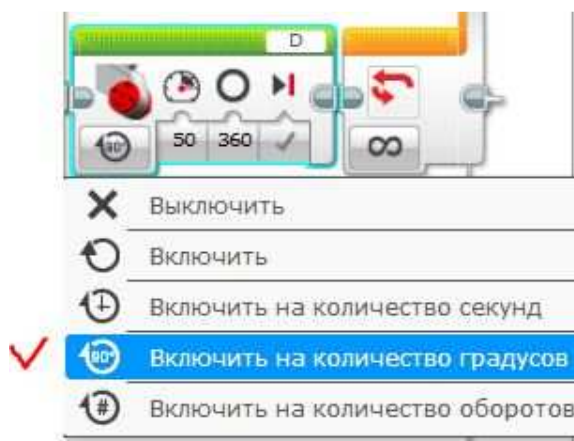


Рис. 28. Выбираем режим работы блока большого мотора

Далее «Мощность» ставим 20, а «Градусы» 275. Порт подключения «В» Рис. 29.

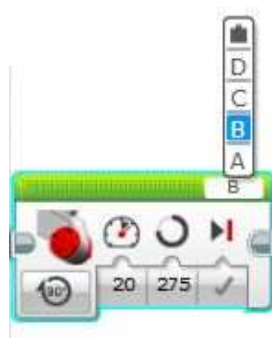


Рис. 29. Редактирование блока


Далее добавляем блок «Средний мотор»  и добавляем к последнему блоку Рис. 30.



Рис. 30. Добавление блока среднего мотора

Редактируем блок. Режим работы- «Включить на количество секунд». «Мощность» ставим 30, а «Секунды» ставим 1. Порт подключения «А» Рис. 31.

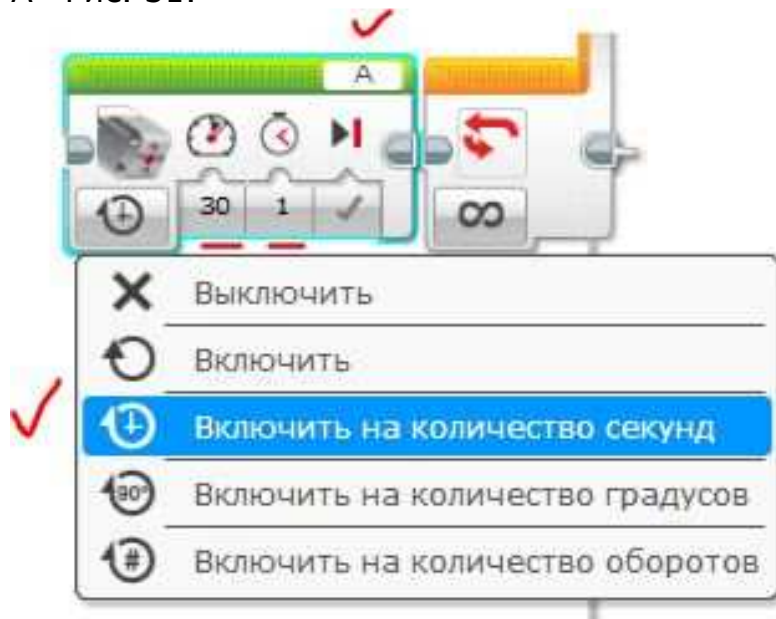


Рис. 31. Ставим режим работы среднего мотора

Далее добавляем блок «Большой мотор». Редактируем блок. Режим работы- «Включить». «Мощность» ставим -55. Порт подключения «В» Рис. 32.

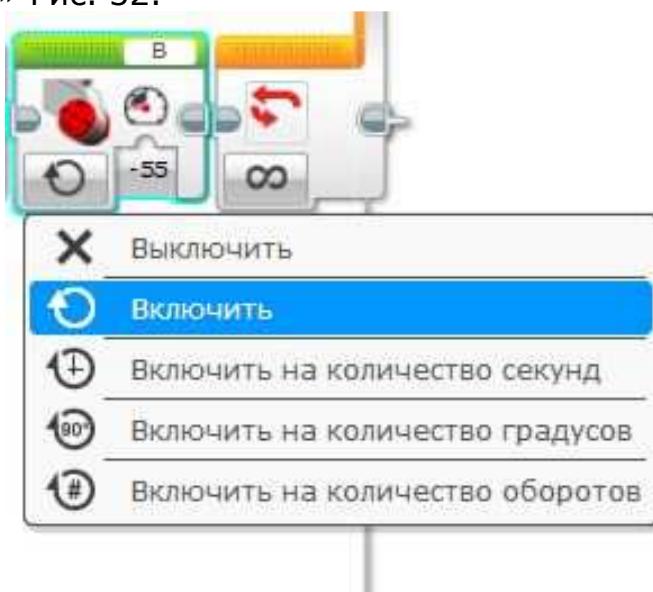


Рис. 32. Выбираем режим работы

Далее добавляем датчик «Ожидание». Редактируем блок. Нажимаем на «Режим работы» и выбираем «Датчик цвета»- «Сравнение»- «Яркость отраженного света» Рис. 33.

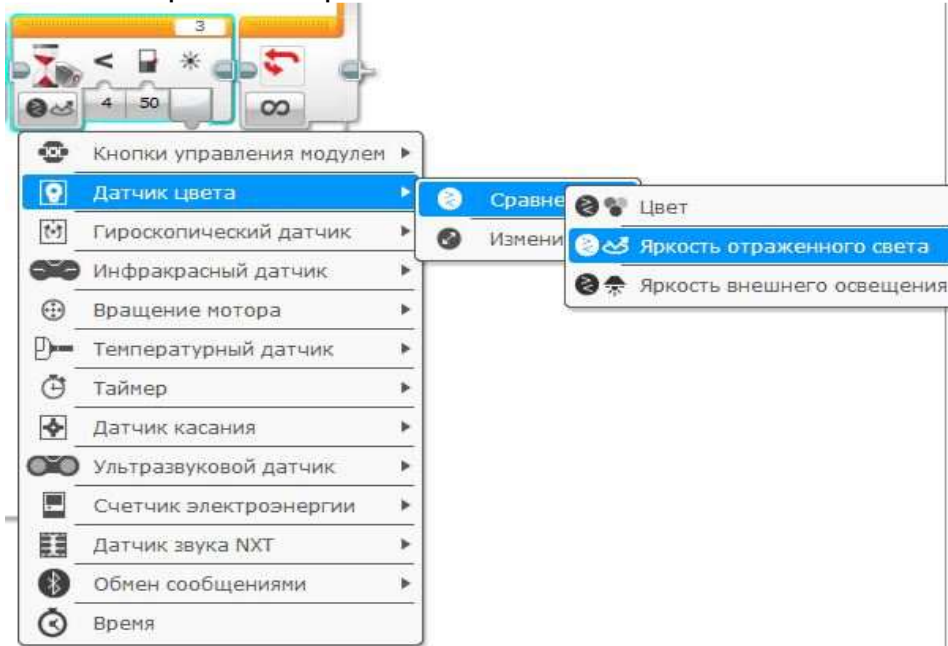


Рис. 33. Редактирование блока

«Тип сравнения» ставим «2», а «Пороговое значение». Порт подключения «3» Рис. 34.

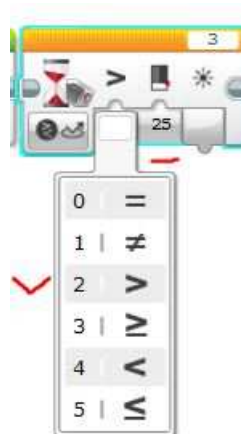


Рис. 34. Настройки блока ожидания

Далее добавляем блок «Большой мотор». Редактируем блок. Режим работы- «Выключить». Порт подключения «В» Рис. 35.



Рис. 35. Настройка блока большого мотора

Далее во вкладке «Управление операторами»  выбираем

блок «Переключатель»



и добавляем его к последнему блоку Рис. 36.

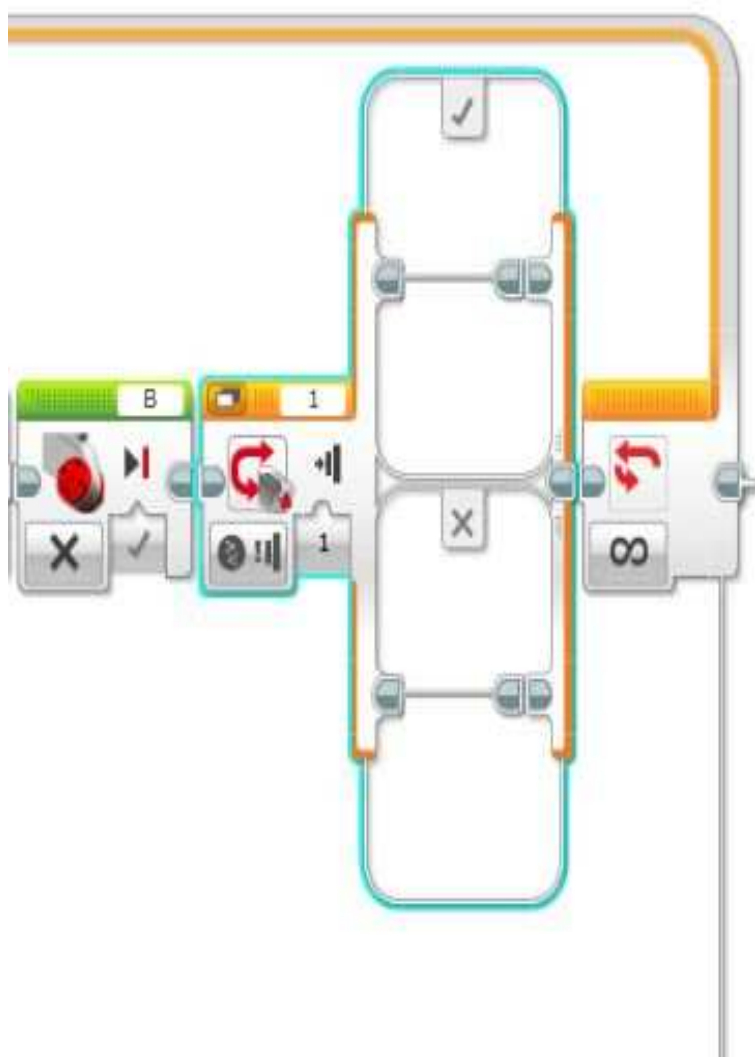


Рис. 36. Добавление блока «Переключатель»

Далее редактируем блок «Переключатель». Нажимаем на кнопку «Режим» и выбираем «Числовое значение». В верхнюю ячейку добавляем блок «Большой мотор» и редактируем его. Режим работы- «Включить на количество оборотов». «Мощность» ставим -65, а «Обороты» ставим 0,865. Порт подключения «С». Нажимаем на верхнюю строку и ставим 4.

В нижнюю ячейку добавляем блок «Большой мотор» и редактируем его. Режим работы- «Включить на количество оборотов». «Мощность» ставим 65, а «Обороты» ставим 0,85. Порт подключения «С». Нажимаем на верхнюю строку и ставим 5. Готовый блок «Переключатель» показан на Рис. 37.

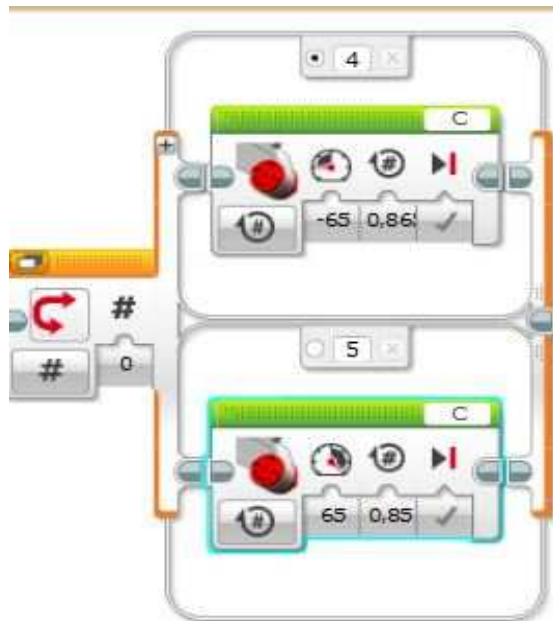


Рис. 37. Редактирование блока «Переключатель»

После этого нам необходимо соединить блок «Ожидание», отмеченный на Рис. 38 цифрой «1» и блок «Переключатель», отмеченный цифрой «2».

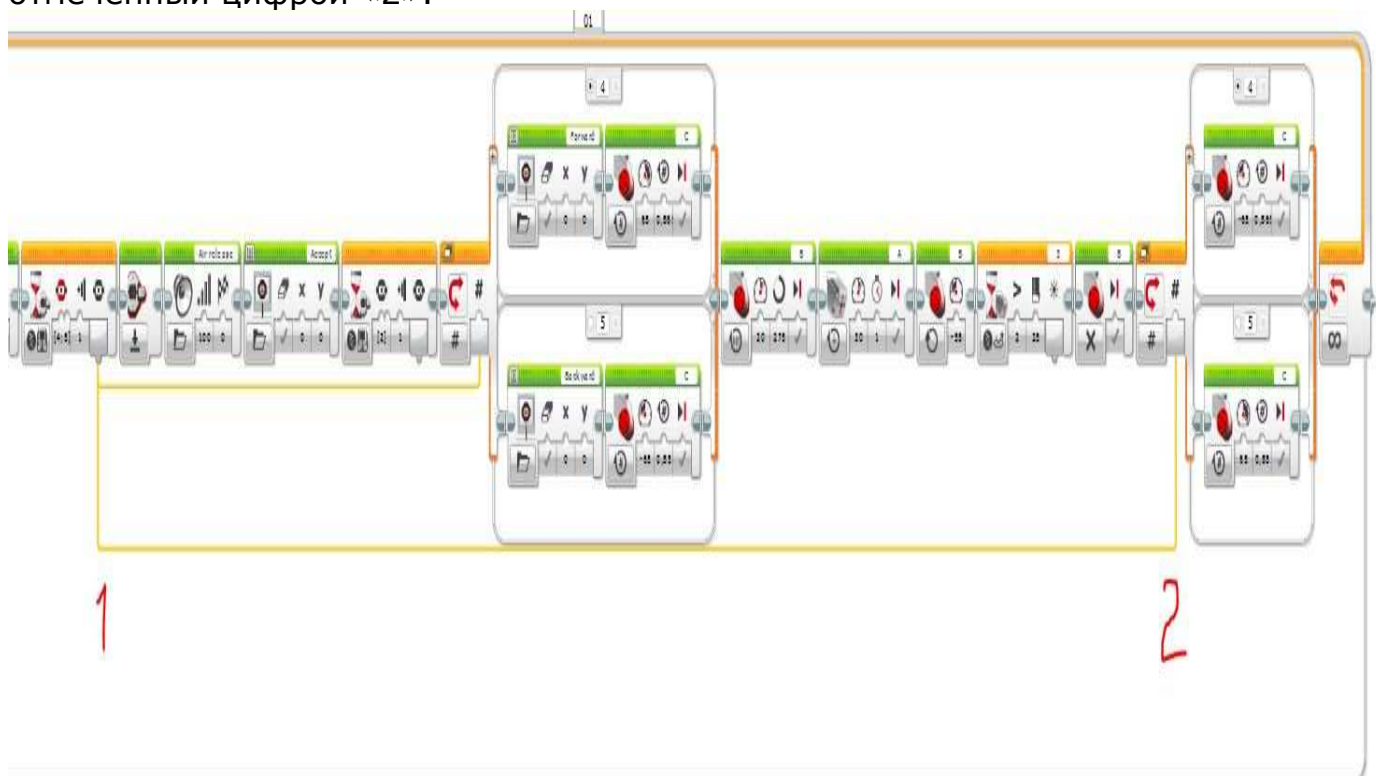


Рис. 38. Установка соединения между блоками

Далее выбираем блок «Большой мотор» и добавляем его в цикл к блоку «Переключатель» Рис. 39.

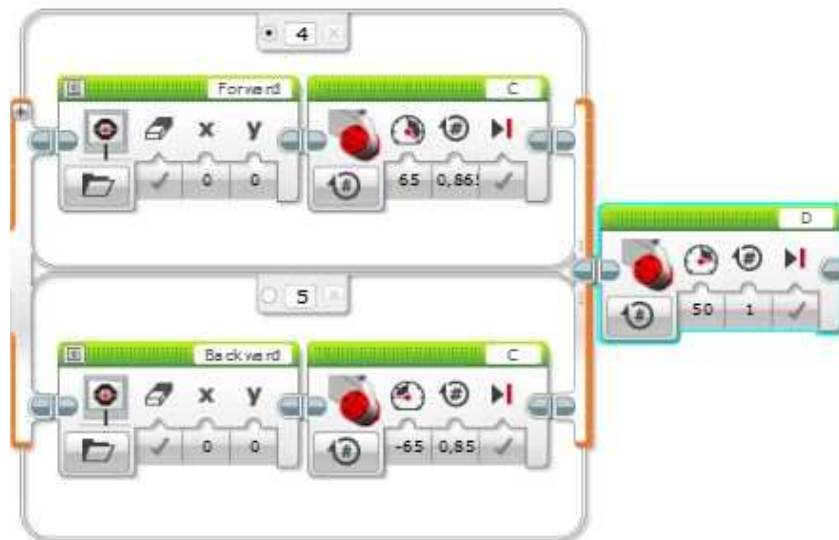


Рис. 39. Добавление блока большого мотора

Редактируем блок. Режим работы- «Включить на количество градусов» Рис. 40.

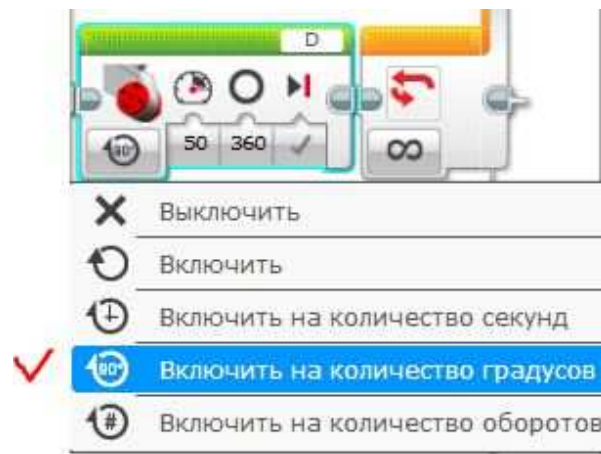


Рис. 40. Установка режима работы мотора

Далее «Мощность» ставим 20, а «Градусы» 275. Порт подключения «В» Рис. 41.

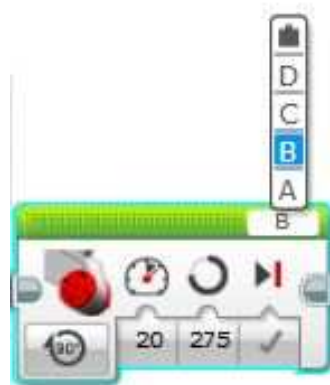


Рис. 41. Редактирование блока


Далее добавляем блок «Средний мотор»  и добавляем к последнему блоку Рис. 42.



Рис. 42. Добавление блока среднего мотора

Редактируем блок. Режим работы- «Включить на количество градусов». «Мощность» ставим -30, а «Градусы» ставим 90. Порт подключения «А» Рис. 43.

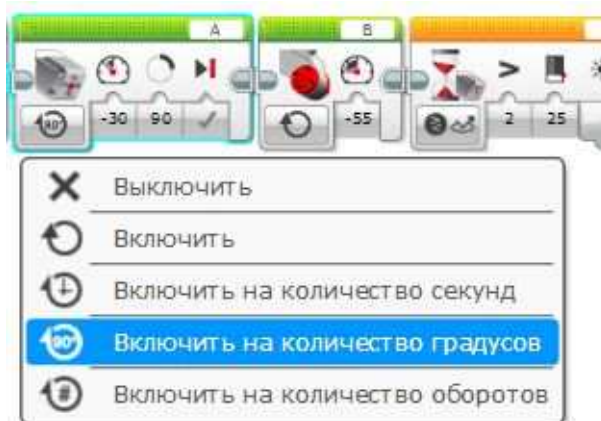


Рис. 43. Выбор режима работы

Далее добавляем блок «Большой мотор». Редактируем блок. Режим работы- «Включить». «Мощность» ставим -55. Порт подключения «В» Рис. 44.

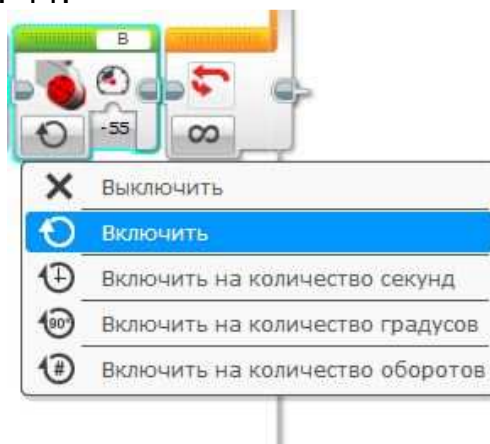


Рис. 44. Редактирование блока

Далее добавляем датчик «Ожидание». Редактируем блок. Нажимаем на «Режим работы» и выбираем «Датчик цвета» - «Сравнение»- «Яркость отраженного света» Рис. 45.

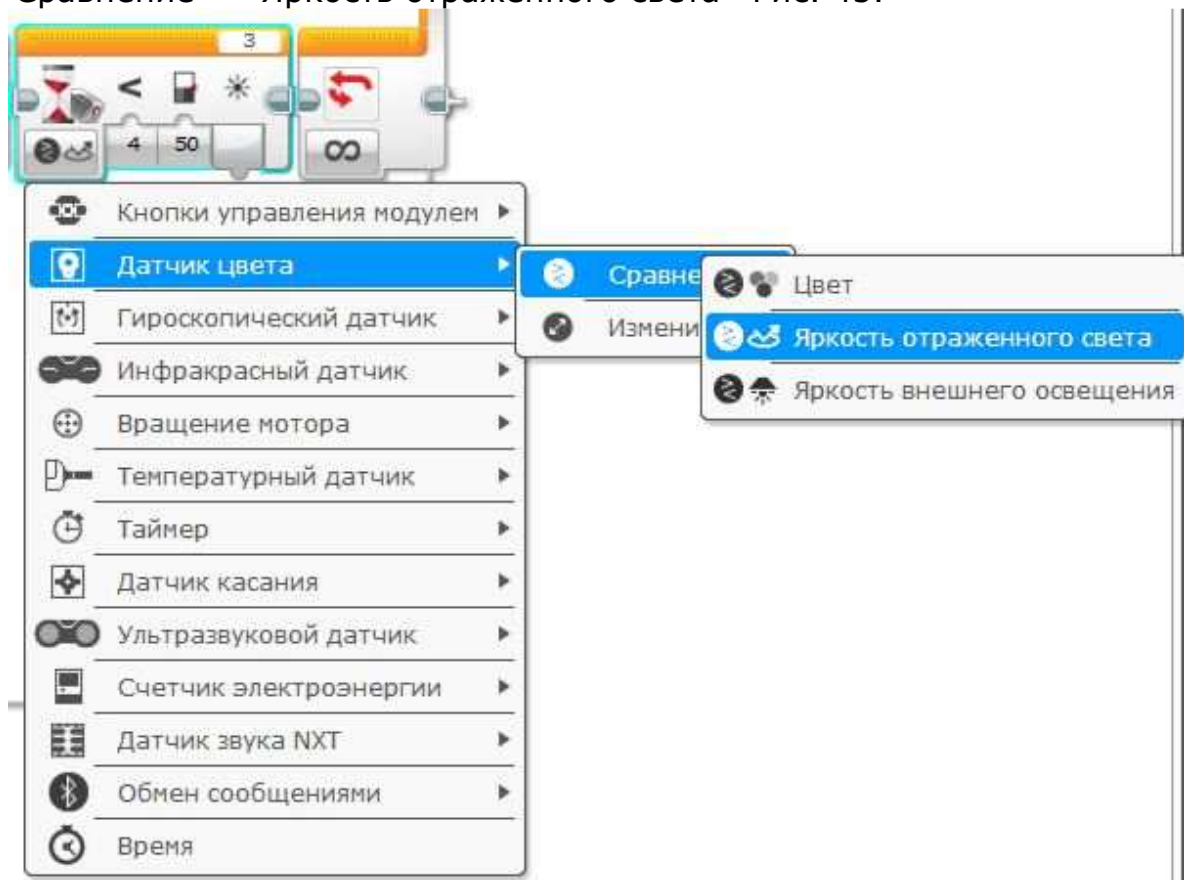


Рис. 45. Редактирование блока

«Тип сравнения» ставим «2», а «Пороговое значение». Порт подключения «3» Рис. 46.

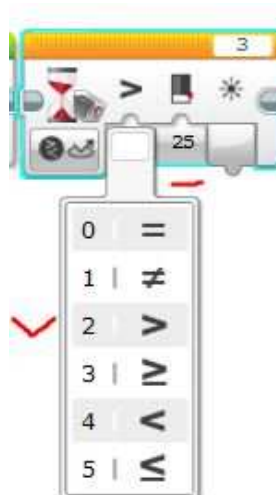


Рис. 46. Настройки блока

Далее добавляем блок «Большой мотор». Редактируем блок. Режим работы- «Выключить». Порт подключения «B» Рис. 47.



Рис. 47. Редактирование блока

Готовая программа продемонстрирована на Рис. 48.

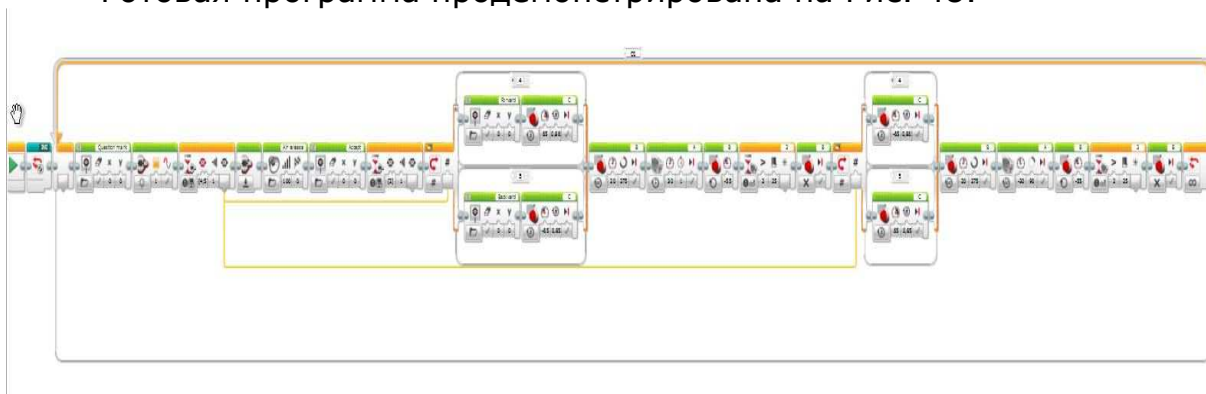


Рис. 48. Готовая программа

Варианты для самостоятельной работы

Вариант №1.

Редактируйте программу «Перемещение предметов». Измените в программе все звуковые сигналы на тип сигнала «Good». Изображение первого блока «Экран» Question mark3.

Вариант №2.

Редактируйте программу «Перемещение предметов». Измените в программе все звуковые сигналы на тип сигнала «Good job». Изображение первого блока «Экран» Question mark2.

Вариант №3.

Редактируйте программу «Перемещение предметов». Измените в программе все звуковые сигналы на тип сигнала «Go». Изображение второго блока «Экран» Accept.

Вариант №4.

Редактируйте программу «Перемещение предметов». Измените в программе все звуковые сигналы на тип сигнала «Good». Изображение второго блока «Экран» Accept2.

Вариант №5.

Редактируйте программу «Перемещение предметов». Измените в программе все звуковые сигналы на тип сигнала «Okay». Изображение второго блока «Экран» Accept3.

Вариант №6.

Редактируйте программу «Перемещение предметов». Измените в программе все звуковые сигналы на тип сигнала «Yes». Изображение второго блока «Экран» Accept.

Контрольные вопросы

1. Что такое блок «Средний мотор»?
2. Что такое блок «Индикатор состояния модуля»?
3. Что такое блок «INI»?
4. Что такое блок «Звук»?
5. Для чего используются блоки переключения в программе «Перемещение предметов»?
6. Какими блоками осуществляется процесс выбора роботом рабочей области в программе «Перемещение предметов»?
7. Для чего мы в программе «Перемещение предметов» использовали блок «INI»?
8. Какие блоки были использованы для создания общего блока-программы «INI»?

Список литературы

1. Руководство пользователя. LEGO MINDSTORMS Education EV3. The LEGO GROUP. 2013.
2. Веб-сайт: <http://www.lego.com/ru-ru/>
3. Остроух А.В., Николаев А.Б. Интеллектуальные системы в науке и производстве /Учебно-методическое пособие. ISBN 978 3 659 98006 0 – Saarbrucken, Germany: Palmarium Academic Publishing, 2012. - 312 с.
4. Николаев А.Б., Остроух А.В. Интеллектуальные системы: учебное пособие / // - М.: МАДИ, 2012. – 271 с.
5. Остроух А.В. Основы построения систем искусственного интеллекта для промышленных и строительных предприятий. Монография. М.: ООО «Техполиграфцентр». 2008. - 280 с.
6. Электронный учебный курс «Системы искусственного интеллекта» <http://www.cdomadi.ru>.
7. Электронные учебные курсы по интеллектуальным системам <http://www.intuit.ru/>

Оглавление

Введение	3
Лабораторная работа №1.....	4
Знакомство со средой разработки LEGO MINDSTORMS Education EV3. Конструирование базовой модели робота	4
Лабораторная работа №2	7
Программирование робота с использованием контроллера	7
Лабораторная работа №3.....	18
Программирование робота на выполнение простых и сложных действий. Многозадачность	18
Лабораторная работа №4.....	35
Шины данных. Регистрация собранных данных.....	35
Лабораторная работа №5.....	47
Использование блока «Переключатель». Программирование графиков	47
Лабораторная работа № 6.	65
Сборка робота- манипулятора.....	65
Лабораторная работа № 7.	70
Программирование робота- манипулятора.....	70
Список литературы	93