

муниципальное бюджетное учреждение
дополнительного образования
станция юных техников
муниципального образования город-курорт Анапа

Методический сборник материалов для подготовки и проведения занятий
раздела движение по линии на основе образовательного набора «LEGO
MINDSTORMS EV3»

Автор составитель:
педагог дополнительного образования
Курганская Анастасия Дмитриевна

г. Анапа, 2024

Вводное занятие №1 Особенности конструкции робота для движения по линии

Сборка стандартного робота для занятия
<https://nsportal.ru/shkola/dopolnitelnoe-obrazovanie/library/2025/03/09/kalibrovka-datchikov-kalibrovka-v-nachale>

Для занятия мы используем стандартную сборку тележки с датчиками цвета направленными вниз. Теперь необходимо определиться с точностью показаний и прохождения черной линии. Следовать по линии можно с помощью:

1. одного датчика цвета, расположенного с одной стороны линии. Робот может двигаться вдоль одной стороны линии, проходить повороты, но не в состоянии обнаруживать перекрестки и инверсии. Тем не менее, на основе одного датчика могут быть реализованы алгоритмы любой сложности, включая ПИД - регулятор;
2. двух датчиков цвета, расположенных по обе стороны линии. Использование двух датчиков позволяет обнаруживать движение робота становится плавным, он практически никогда не «теряет» линию, даже на очень крутых поворотах; пересекаемые роботом перекрестки,
3. трех датчиков, два из которых расположены с обеих сторон линии, третий вынесен в сторону и немного вперед. Третий датчик в этом случае помогает обнаруживать и подсчитывать перекрестки, линии штрих-кода, повороты на острый угол и обнаруживать дополнительные препятствия;
4. трех датчиков, два из которых расположены с обеих сторон линии, а третий находится над линией и вынесен немного вперед. Такая конструкция позволяет предвидеть изменение траектории и заранее принимать меры по ее прохождению;
5. четырех датчиков. Четвертый датчик служит для реализации скоростных алгоритмов движения по сложным и насыщенным препятствиями трассам.

Определившись с количеством датчиков, необходимо правильно их разместить. Робот будет иметь устойчивую и сбалансированную конструкцию, если его масса равномерно распределена на все опоры, это необходимо учитывать в первую очередь. Конструкция робота может быть на основе переднего и заднего привода.

Для начала, собираем базовую модель стандартного учебного робота.

В дальнейшем нам понадобится обозначать датчики и моторы. Условимся смотреть на робота сверху по ходу движения. Не забываем подключить порты робота, теперь можно приступить к программированию.

Раздел №1 Простое движение по линии

Тема занятия №1: Простое движение по линии калибровка датчика цвета.

Запуская пробную программу можно столкнуться с тем, что наши роботы не всегда четко следуют по линии, и в некоторых случаях роботы блуждают. Это связано с тем, что мы не производили калибровку датчика цвета и он работает как придётся. Сегодня мы заставим его работать в четких рамках выверенного значения. Для этого проведем калибровку датчика цвета.

Калибровка датчиков

Что входит в понятие калибровка датчиков?

Калибровка датчика будет заключаться в том, что мы должны найти то состояние, к которому будет стремиться датчик, нахождение черного, белого, серого.

Что такое среднее значение серого?

Среднее значение серого – это значение, которое показывает датчик цвета (в режиме отраженного света), находясь на границе линии и поля.

Основные этапы алгоритма ручной калибровки датчика

1. Помещаем робота так, чтобы датчик цвета оказался над черной линией, и фиксируем его значение. Фиксировать значение можно на экране блока, или, если робот и компьютер связаны проводным или беспроводным соединением, непосредственно в среде EV3. Предположим, датчик показал значение 30.
2. Помещаем робота так, чтобы датчик цвета оказался над белой поверхностью поля, и фиксируем это значение. Предположим, датчик показал значение 70.
3. Вычисляем среднее значение: $(30+70)/2 = 50$. Значение 50 и будем называть среднее значение серого, оно будет фиксироваться датчиком тогда, когда он будет находиться над границей черной линии и белого поля.

Данный способ хорош при неизменном освещении, но такое условие почти не выполнимо и тележка будет съезжать с линии на резких поворотах и при изменении уровня освещенности и добавления нового источника освещения.

Запускаем роботов проводим пробные заезды, отмечаем ошибки в движении робота. Пробуем запуски при разном освещении отмечаем зависимость от постоянной ручной калибровки

Раздел №1 Простое движение по линии

Тема занятия №2: Простое движение по линии «Зигзаг»

Учебное пособие движение по линии «Зигзаг»
<https://nsportal.ru/node/6908542>

Калибровка датчиков является первым шагом реализации алгоритма движения по линии. Датчик цвета, работая в режиме измерения количества отраженного света, освещает поверхность и фиксирует по 100-значной шкале количество отраженного от нее света. Если 100% света, созданного датчиком, отразится, то датчик передаст значение 100. Если от поверхности ничего не отразится, то значение датчика будет равно 0.

Этот алгоритм является самым простым. Наша задача написать программу движения зигзаг, в приложении LEGO MINDSTORMS EV3.

Пишем алгоритм «Зигзаг» с одним датчиком цвета

- 1.Проводим калибровку датчика цвета, определяем среднее значение серого, полученное значение вносим в программу.
- 2.Выполняем программу в цикле:

- если значение датчика меньше среднего значения серого (то есть датчик оказался над черной линией), то левый мотор – 0, правый – 50;
- если значение датчика больше или равно среднему значению серого (датчик удаляется влево от черной линии), то левый мотор – 50, правый – 0.

В этом случае робота надо расположить так, чтобы датчик находился слева от линии.

Если мы хотим запустить робота с правой стороны от линии, то пункт 2 будет выглядеть так: если значение датчика меньше среднего значения серого, то левый мотор – 50, правый – 0; если значение больше или равно среднему значению серого, то левый мотор – 0, правый – 50.

Если движения робота очень резкие, то скорость движения робота можно снизить с 50 на меньшее.

Запуск роботов, проверка программы.

Алгоритм «Зигзаг» с двумя датчиками цвета

При наличии двух датчиков цвета робот практически никогда не «теряет» линию, даже при самых крутых поворотах. Кроме того, двумя датчиками возможно определять перекрестки и оказавшись на них, выполнять определенные действия, например, подсчитывать их количество или поворачивать в определенную сторону.

Алгоритм управления роботом:

1. Проводим калибровку одного датчика цвета, определяем среднее значение серого (пусть среднее значение равно 50).

2. Если значение левого датчика, подключенного в порт 1, больше среднего значения серого, то анализируем значение правого датчика, подключенного в порт 2:

- если значение правого датчика, подключенного в порт 2, меньше среднего значения серого, то левый мотор В - 50, правый мотор С - 0 (робот отклонился влево от линии);

- если значение правого датчика больше или равно среднему значению серого, то левый мотор - 50, правый мотор - 50 (робот находится на линии, датчики расположены по обе стороны от линии, робот едет вперед).

3. Если значение левого датчика, подключенного в порт 1, меньше или равно среднему значению серого, то анализируем значение правого датчика, подключенного в порт 2:

- если значение правого датчика меньше среднего значения серого, то левый мотор - 50, правый - 50 (оба датчика находятся на линии, значит перед нами перекресток, проезжаем его прямо);

- если значение правого датчика больше или равно среднему значению серого, то левый мотор - 0, правый - 50 (робот отклонился вправо от линии).

Выводы:

- алгоритм прост для понимания и воспроизведения;

- робот способен следовать вдоль линии с резкими поворотами или изгибающейся на любой угол;

- скорость прохождения трассы низкая.

Раздел №1 Простое движение по линии

Тема занятия №3: Простое движение по линии «Волна»

Учебное пособие движение по линии «Волна» <https://nsportal.ru/node/6908542>

Алгоритм «Волна» напоминает предыдущий алгоритм «Зигзаг», но предназначен для более быстрого и плавного прохождения дистанции.

Алгоритм работает также с одним и двумя датчиками цвета; отличается от предыдущего тем, что повороты робот выполняет не вокруг одного колеса, когда значение скорости одного мотора - 0, а второго – 50, а по дуге. Моторы работают со скоростями 60 и 10, 50 и 5, и т.д. Эти значения подбираются экспериментально в зависимости от кривизны и толщины линии, а также от конструкции робота.

Программа движения робота по левой стороне от линии с одним датчиком цвета, подключенного к порту 2.

Запуск роботов, проверка программы, проведение пробных запусков

Выводы:

- робот способен следовать вдоль линии с плавными поворотами, на крутых поворотах робот теряет линию;
- скорость прохождения трассы достаточно высокая.

Раздел №2 Движение по линии с регуляторами

Тема занятия №1: «Алгоритм пропорциональное регулирование»

Пропорциональное управление роботом. В теории автоматического управления более 100 лет назад был разработан автоматический регулятор (лат. Regulo - привожу в порядок, налаживаю) - устройство, посредством которого осуществляется автоматическое регулирование. С помощью чувствительного элемента (датчика) регулятор, в зависимости от принципа регулирования, измеряет регулируемую величину и при помощи преобразовательного или вычислительного устройства, и в соответствии с законом регулирования выбирает воздействие на регулируемый механизм.

В настоящее время наиболее распространен пропорционально-интегрально-дифференцирующий регулятор или ПИД - регулятор. Алгоритм ПИД - регулятора, адаптированный для управления роботом Lego Mindstorms EV3, позволяет оценивать положение датчика (датчиков) цвета относительно следуемой линии и, в зависимости от результата, рассчитывать управляющее воздействие на моторы.

В разобранных выше дискретных алгоритмах скорость поворота робота при выполнении зигзага или волны постоянна и не зависит от того, насколько далеко робот отъехал от линии. Это вызывает неоптимальное прохождение пути, так как даже на прямых участках робот будет выполнять ненужные движения.

ПИД - регулятор содержит три составляющие. Рассмотрим каждую из них.

1. Пропорциональная составляющая (П) позволяет задавать управляющие воздействия на моторы (тормозить один мотор и ускорять второй мотор) с силой, пропорциональной величине отклонения робота от следуемой линии. Скорости моторов рассчитываются как разность требуемого и текущего положения, умноженная на неотрицательный коэффициент. Чем больший коэффициент будет задан, тем быстрее будет происходить возвращение робота на требуемую траекторию, но движения при этом станут более резкими и зигзагообразными. Пропорциональная составляющая оказывает самое сильное управляющее воздействие на моторы.

2. Интегральная составляющая (И) учитывает информацию о положении робота на предыдущих шагах и задает дополнительные воздействия на моторы в зависимости от полученного результата.

3. Дифференциальная составляющая (Д) прогнозирует положение робота на следующем шаге и задает дополнительные управляющие воздействия на моторы в зависимости от полученного результата

Уравнение ПИД - регулятора имеет вид:

$$V(t) = P + I + D = k_p e(t) + k_i \int e(t) dt + k_d \frac{de(t)}{dt}$$

где

t - период дискретизации;

$V(t)$ - требуемая скорость мотора;

$e(t)$ - ошибка, указывающая, на сколько далеко датчик цвета находится от среднего значения серого;

P, I, D - пропорциональная, интегральная и дифференциальная составляющие соответственно;

k_p, k_i, k_d - коэффициенты ПИД регулятора.

В приведенном виде ПИД - регулятор невозможно реализовать на компьютере, поэтому дифференцирование заменим разностью, а интегрирование - суммой:

$$V(t) = P(t) + I(t) + D(t);$$

$$P(t) = k_p * e(t);$$

$$I(t) = I(t-T) + k_i * e(t);$$

$$D(t) = k_d * (e(t) - e(t-T)).$$

Реализация регуляторов возможна на основе одного или двух датчиков цвета. Существует возможность использования некоторых составляющих регулятора отдельно. Благодаря простоте реализации очень распространенным является П - регулятор. Интегральная и дифференциальная составляющие предназначены для более тонкого регулирования.

Пропорциональный регулятор очень популярен для управления движением по линии из-за стабильной работы и простоты реализации в среде EV3. Управление на основе П-регулятора возможно на основе одного или двух датчиков цвета в режиме отраженного света. Приведем два алгоритма и примеры их реализации для управления движением робота в среде Lego Mindstorms EV3.

Робот может двигаться по алгоритму пропорционального управления на основе одного датчика с левой или правой стороны линии. Это может быть важно при проезде перекрестков или других препятствий. Различие заключается в том, что управления, подаваемые на моторы, должны быть противоположны: при движении робота слева от линии и при приближении к линии робот должен поворачивать налево, при движении робота справа от линии - поворачивать направо.

Алгоритм движения робота:

В непрерывном цикле опрашиваем показание датчика цвета, в режиме измерения отраженного света и:

- в случае движения робота с левой стороны от линии вычитаем из полученного значения среднее значение серого (40);

-

в случае движения робота с правой стороны от линии вычитаем из среднего значения серого (40) полученное значение;

- результат умножаем на коэффициент $k_p = 1,5$;

- в блоке математика пишем формулу пропорционального коэффициента $(A-B)*C$ точное значение находим эмпирически, подбором значений C от 1-1,5, в зависимости от условий освещения.

Запуск роботов, проверка, делаем выводы:

- реализация П-регулятора на основе блока рулевого управления проста и доступна для начинающих знакомство с пропорциональным управлением;

- алгоритм показывает хорошие результаты на плавных трассах;

- для прохождения крутых поворотов требуется увеличение значения коэффициента k_p и/или снижение мощности, подаваемой на моторы, что приводит к медленному зигзагообразному движению.

Раздел №2 Движение по линии с регуляторами

Тема занятия №2: «Автоматическая калибровка в движении»

Учебное пособие алгоритм автоматической калибровки датчика
<https://nsportal.ru/node/6908548>

Автоматическая калибровка — это функция, которая позволяет роботу самостоятельно выполнять калибровку в процессе движения без необходимости использования внешних вычислений.

Приступаем к написанию программы в LEGO MINDSTORMS EV3 для автоматической калибровки

1. добавляем блоки ожидания датчика касания измерения датчик цвета звук ожидание датчика касание измерения цвета и звук
2. добавляем блок математика
3. тянем шину данных от блоков измерение цвета последовательно в значениях А и В
4. далее выбираем параметр своё уравнение в шапку блока записываем простую формулу для расчета среднего значения $(A + B) / 2$, эта формула посчитает нам среднее значение, которое измерили датчики.
5. далее полученное равенство уравнение шины данные протягиваем переключателю и подключаем к числовому значению.
6. далее добавляем датчик касания для запуска тележки.
7. проверяем написанное
8. выделяем блоки для создания собственного блока
9. переходим в Панель инструменты выбираем конструктор мой блок
10. подписываем имя блоку вставляем описание добавляем описание и имя пишется на латинице.
11. добавляем иконку.
12. выбираем настраиваемый параметр добавляем ему имя.
13. выбираем значение параметра кликаем на иконку после чего нажимаем кнопку Завершить.
14. видим, что наши блоки преобразовались в один блок.

15. для дальнейшей работы с этим блоком его можно сохранить так как автоматическая необходимо нам и в других программах. Запускаем роботов по линии.

Автоматическую калибровку можно использовать с простыми алгоритмами движения по линии «Зигзаг» и «Волна» с 2-мя датчиками.

Выводы:

Использование, автоматической упрощает работу с расчетом среднего серого делает ее автоматической, не зависимой от освещения и цвета поверхности.

Список литературы и интернет-источников:

1. Алгоритмизация и программирование [Текст] / И.Н. Фалина, И.С. Гущин, Т.С. Богомолова и др. – М.: Кудиц-Пресс, 2007. – 276 с.
2. Алгоритмы и программы движения по линии робота LEGO MINDSTORMS EV3 [Текст]/ Овсяницкая Л. Ю. . — М.: Издательство «Перо», 2015.с. 23
3. Соревновательная робототехника: приёмы программирования в среде EV3. Учебно-практическое пособие. [Текст] Вязовов С. М., Калягина О. Ю., Слезин К. А. — М.: Издательство «Перо», 2014. с.111
4. Робототехника для детей и родителей. [Текст] Филиппов С.А. – СПб.: Наука, 2013. с.319
5. Методические рекомендации по использованию образовательного набора LEGO MINDSTORMS EV3 в образовательном процессе[Текст] Сидоренко, И. В. , Издательство "КноРус", Г.2020,с. 200
6. Применение LEGO MINDSTORMS EV3 в образовательной деятельности [Текст] Золотухина, Н. В., Издательство "Просвещение", г. 2019, с.150
7. Технологии программирования в образовательном процессе с LEGO MINDSTORMS EV3 [Текст] Иванов, К. Ф., Современные образовательные технологии,г. 2022,с. 180
8. Создание образовательных проектов с LEGO MINDSTORMS EV3[Текст] Тарасова, О. С. , Физматлит, г. 2023, с.210
9. LEGO MINDSTORMS EV3 как средство обеспечения STEAM-образования [Текст] Максимова, Е. В., Издательство "Белый ветер", г.2018, с.160
10. Робототехника и программирование [Электронный ресурс]. -URL: <https://www.youtube.com/channel/UCsEf9ACflVM9fIYcUpHXkQQ>
11. Официальный сайт соревнований Робофест [Электронный ресурс]. - URL: <https://www.robofest.ru/>
12. Innopolis open Robotics [Электронный ресурс]. -URL: <http://robolymp.ru/>
13. Федерация спортивной и образовательной робототехники [Электронный ресурс]. -URL: <https://sportrobotics.ru/>
14. Региональные центры «Инженерные кадры России» - ИКаР и ИКаРёнок [Электронный ресурс]. -URL: <https://икар.фгос.пф/regionalnyeresursnye-tsentry-inzhenernye-kadry-rossii>
15. Образовательная социальная сеть nsportal.ru [Электронный ресурс]. - URL: <https://nsportal.ru/node/6908542>,