

УДК 004.42+004.896

ББК 32.81

О 34

О 34 Овсяницкая, Л.Ю. Пропорциональное управление роботом Lego Mindstorms EV3 / Л.Ю. Овсяницкая, Д.Н. Овсяницкий, А.Д. Овсяницкий. – М.: Издательство «Перо», 2015. – 188 с.

ISBN 978-5-00086-947-5

В книге рассмотрено применение алгоритмов пропорционального управления роботом Lego Mindstorms EV3 с использованием различных датчиков: движение робота вдоль стены, слежение за движущимся предметом, следование за маячком, движение по линии и по заданному курсу, удержание тяжелого груза в требуемом положении, реализация плавного разгона и торможения, имитация эффекта гелиотропизма.

Отдельная глава посвящена реализации в среде Lego Mindstorms EV3 методики Циглера-Николса, позволяющей теоретически рассчитать оптимальные параметры управления, планировать и реализовывать научный эксперимент, позволяя раскрыть научный потенциал образовательной робототехники.

Все алгоритмы сопровождаются подробным объяснением и программным кодом. Книга будет полезна педагогам начального, среднего, высшего и дополнительного образования, учащимся, студентам и всем, интересующимся вопросами робототехники.

Рецензент:

доктор физико-математических наук, профессор А.Ф. Шориков.

Все права защищены. Любая часть этой книги не может быть воспроизведена без письменного разрешения владельцев авторских прав.

ISBN 978-5-00086-947-5

© Л.Ю. Овсяницкая, Д.Н. Овсяницкий, А.Д. Овсяницкий, 2015.

## СОДЕРЖАНИЕ

Введение.....	5
1. Адаптация алгоритмов пропорционального управления к среде Lego Mindstorms EV3 .....	9
1.1. Пропорциональный регулятор.....	9
1.2. Пропорциональный интегральный дифференциальный регулятор .....	16
2. Примеры пропорционального управления на основе различных датчиков.....	33
2.1. Применение датчиков ультразвука.....	33
2.1.1. Движение вдоль стены.....	33
2.1.2. Слежение за движущимся объектом. Проект «Сторожевая башня» .....	44
2.2. Применение датчика гироскопа. Движение робота по заданному курсу .....	53
2.3. Применение инфракрасного датчика. Движение робота за маячком. Проект «Робот-собачка».....	63
2.4. Применение датчика вращения моторов.....	75
2.4.1. Синхронизация моторов .....	76
2.4.2. Удержание тяжелого груза под заданным углом.....	82
2.4.3. Реализация плавного разгона и торможения робота.....	88
2.5. Применение датчика цвета.....	101
2.5.1. Движение робота по линии.....	102
2.5.2. Имитация гелиотропизма – поворота растения в сторону солнца. Проект «Робо-подсолнух».....	120
3. Расчет коэффициентов ПИД-регулятора методом Циглера-Николса .....	134

Заключение .....	149
Приложение 1. Сборка базовой конструкции робота.....	151
Приложение 2. Установка датчиков ультразвука.....	175
Приложение 3. Установка датчика гироскопа .....	180
Приложение 4. Установка инфракрасного датчика.....	184
Приложение 5. Установка датчиков цвета.....	185

## ВВЕДЕНИЕ

Дорогие друзья! В теории автоматического управления более 100 лет назад был разработан автоматический регулятор (лат. *regulo* – привожу в порядок, налаживаю) – устройство, с помощью которого осуществляется автоматическое регулирование различных процессов. С помощью датчиков измеряется регулируемая величина и, в соответствии с заданным законом, вырабатывается управляющее воздействие на различные системы робототехнического комплекса.

Благодаря простоте реализации алгоритма пропорционального управления в среде программирования Lego Mindstorms EV3, он приобрел заслуженную популярность.

Использование алгоритма возможно на основе любых датчиков, выходные значения которых изменяют свои показания пропорционально измеряемой величине. То есть можно работать со всеми датчиками, за исключением датчиков касания, и решать большое количество разноплановых задач.

Алгоритмы пропорционального управления способны реализовать более плавное движение по сравнению с дискретным управлением. Изучив книгу, вы научитесь применять алгоритмы пропорционального управления для решения задач с использованием

различных датчиков Lego Mindstorms EV3 и реализовывать следующие задачи:

- движение вдоль стены на заданном расстоянии;
- поиск и определение направления движущегося в зоне видимости робота предмета, слежение за ним;
- следование робота за маячком;
- движение робота по линии;
- движение робота по заданному курсу;
- удержание тяжелого груза в требуемом положении;
- реализация плавного разгона и торможения робота.

В книге подробно рассмотрен пример очень красивого и эффектного проекта «Робо-подсолнух», имитирующего эффект гелиотропизма – поворота растений в сторону солнца. На основе алгоритма пропорционального управления созданный робо-подсолнух будет поворачивать цветок в сторону более яркого света в горизонтальной и вертикальной плоскости.

Решение каждой задачи содержит подробный алгоритм и программный код. Приведены схемы сборки приведенных моделей роботов.

Третья глава, написанная совместно с кандидатом технических наук, доцентом А.Е. Васильевым, посвящена научному подходу к изучению элементов систем автоматического управления. В главе приведена полностью адаптированная к среде Lego Mindstorms EV3 методика Циглера-Николса, позволяющая теоретически

рассчитать параметры ПИД-регулятора и на практике увидеть полученные результаты.

Пошаговое руководство позволит запланировать и провести научный эксперимент, результатом которого будут оптимальные значения пропорционального, интегрального и дифференциального коэффициентов ПИД-регулятора.

Таким образом, знакомство с теорией управления, постановка исследовательского эксперимента, проведение измерений, анализ, визуализация и обоснованная интерпретация полученных результатов позволят раскрыть научный потенциал Lego-робототехники и обеспечить системное развитие инженерной культуры детей и подростков.

Переход на новый научно-исследовательский уровень изучения Lego-робототехники является связующим звеном между детским или подростковым увлечением и началом изучения серьезных технических дисциплин.

Данная книга предполагает начальные знания основ алгоритмизации и умений работы со средой программирования Lego Mindstorms EV3. Рекомендуем предварительно ознакомиться с книгой «Курс программирования робота Lego Mindstorms EV3 в среде EV3: основные подходы, практические примеры, секреты мастерства», авторы – Л.Ю. Овсяницкая, Д.Н. Овсяницкий, А.Д. Овсяницкий.

Работа является результатом многолетнего опыта как непосредственного участия в региональных,

всероссийских и международных состязаниях по робототехнике, так и педагогической деятельности, направленной на подготовку учеников, преподавателей и тренеров по данной тематике.

# 1. АДАПТАЦИЯ АЛГОРИТМОВ ПРОПОРЦИОНАЛЬНОГО УПРАВЛЕНИЯ К СРЕДЕ LEGO MINDSTORMS EV3

## 1.1. Пропорциональный регулятор

**Регулятор или управляющее устройство** – в теории управления устройство, которое следит за состоянием объекта управления как системы и вырабатывает для неё управляющие сигналы в соответствии с заданным законом управления. Согласно определению, робот с помощью датчиков измеряет регулируемую величину и вырабатывает требуемое воздействие на моторы.

Например, если задача заключается в движении робота по линии, то датчик цвета будет измерять текущее значение отраженного света, а алгоритм регулятора каждое полученное значение начнет сравнивать с требуемым, заранее установленным значением, и вырабатывать такие воздействия на моторы (тормозить один мотор и ускорять второй мотор), которые заставят робота изменить направление движения и достичь требуемого показания датчика.

При этом чем дальше робот отъехал от линии, то есть чем больше ошибка, тем с бóльшей скоростью он будет возвращаться на линию. Результатом становится приближение робота к линии.

Если необходимо осуществлять движение вдоль стены на одинаковом расстоянии, повторяя все ее



повороты, мы будем использовать датчик ультразвука. Регулятор будет сравнивать текущее расстояние до стены с требуемым и выработать воздействия на моторы, приближающие или удаляющие робота от стены, и одновременно сохраняющие его поступательное движение вперед.

Уравнение пропорционального регулятора (П-регулятора) имеет вид:

$$V(t) = k_p e(t), \text{ где}$$

$t$  – текущее время;

$V(t)$  – требуемая скорость мотора в момент времени  $t$ ;

$e(t)$  – ошибка рассогласования в момент времени  $t$ ;

$k_p$  – коэффициент П-регулятора.

Реализация регуляторов возможна на основе одного или двух датчиков. Если мы используем один датчик, то условимся называть ошибкой рассогласования разницу между реальным значением датчика и требуемым значением, к которому мы стремимся. Например, пусть при движении по линии мы хотим, чтобы датчик цвета располагался над границей линии и поля. Предварительно измерив значение отраженного света в данном положении робота, мы в каждый такт цикла будем вычитать это значение из текущего значения, определяемого датчиком цвета. Затем полученную ошибку рассогласования умножаем на коэффициент  $k_p$  и результат подаем в качестве управляющего воздействия на моторы.

## 2. ПРИМЕРЫ ПРОПОРЦИОНАЛЬНОГО УПРАВЛЕНИЯ НА ОСНОВЕ РАЗЛИЧНЫХ ДАТЧИКОВ

### 2.1. Применение датчиков ультразвука

#### 2.1.1. Движение вдоль стены

*Задача: робот должен двигаться на расстоянии 15 см вдоль стены, повторяя все ее изгибы и повороты.*

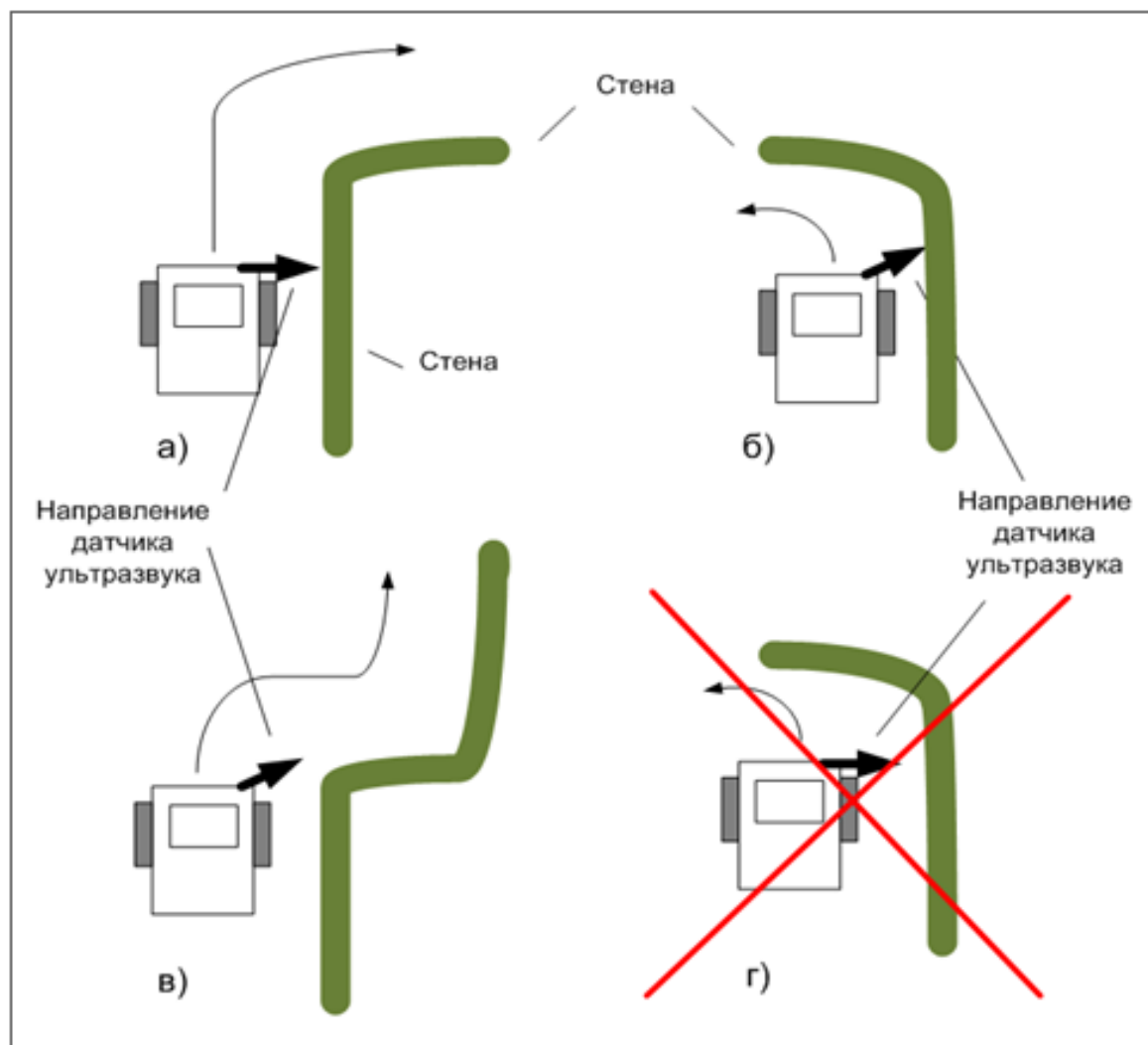
Для решения задачи соберите по приведенной схеме базовую конструкцию робота (Приложение 1) и установите с *правой* стороны робота датчик ультразвука (Приложение 2).

В том случае, когда роботу необходимо следовать вдоль стены с *внешними поворотами* (рис. 2.1.1.1а), датчик ультразвука может быть установлен *перпендикулярно движению робота*. В этом случае движения будут более плавными, с меньшими колебаниями.

Однако если в пути предстоит преодолеть *внутренние углы* (рис. 2.1.1.1б), датчик ультразвука должен устанавливаться *под углом*, так как при перпендикулярном расположении датчик не увидит поворот, робот застрянет в углу, а моторы продолжат вращение (рис. 2.1.1.1г). Движение вдоль прямой будет происходить с бóльшими колебаниями из-за того, что

даже при небольшом отклонении от нужного курса диагональ увеличивается больше по сравнению с перпендикуляром. Соответственно, пропорциональное управление будет задавать большие управляющие воздействия на моторы, что вызывает резкие движения.

Если при движении встречаются внешние и внутренние углы, например, при проходе лабиринта, датчик ультразвука следует устанавливать под углом (рис. 2.1.1.1в).



*Рисунок 2.1.1.1 Схема расположения робота и датчика ультразвука в зависимости от типа поворота*

При любом расположении датчик ультразвука должен устанавливаться впереди робота на некотором расстоянии относительно колес. Это вызвано тем, что робот при управлении инерционен и любое воздействие на моторы отрабатывает с некоторым запаздыванием.

На рис. 2.1.1.2 показаны описанные выше варианты расположения датчика ультразвука на роботе.



*Рисунок 2.1.1.2. Варианты расположения датчика ультразвука на роботе*

Вы можете использовать как EV3, так и NXT датчик ультразвука. Единственным отличие датчиков заключается в точности определения расстояния (EV3-датчик измеряет расстояние с точностью +/- 1 см, NXT-датчик – с точностью +/- 3 см).

***Важно!***

*Датчик ультразвука корректно работает только при заряженной батарее. При низком заряде батареи датчик показывает неверные результаты измерения и сравнения!*

---

**Алгоритм решения задачи движения робота с левой стороны от стены на основе датчика ультразвука, установленного перпендикулярно движению**

1. Создаем цикл 01. Условие выхода из цикла – неограниченно.
2. Первый программный блок – «Ультразвуковой датчик» в режиме «Измерение» – «Расстояние в сантиметрах».
3. Результат измерений заводим в блок «Математика» в режиме «Дополнения (ADV)».
4. Задаем формулу управления:  $(a-b)*c$ .
5. Результат, вычисленный в блоке «Математика», подаем на вход «Рулевое управление» программного блока «Рулевое управление».
6. Задаем параметр «Мощность» блока «Рулевое управление» равным 30.

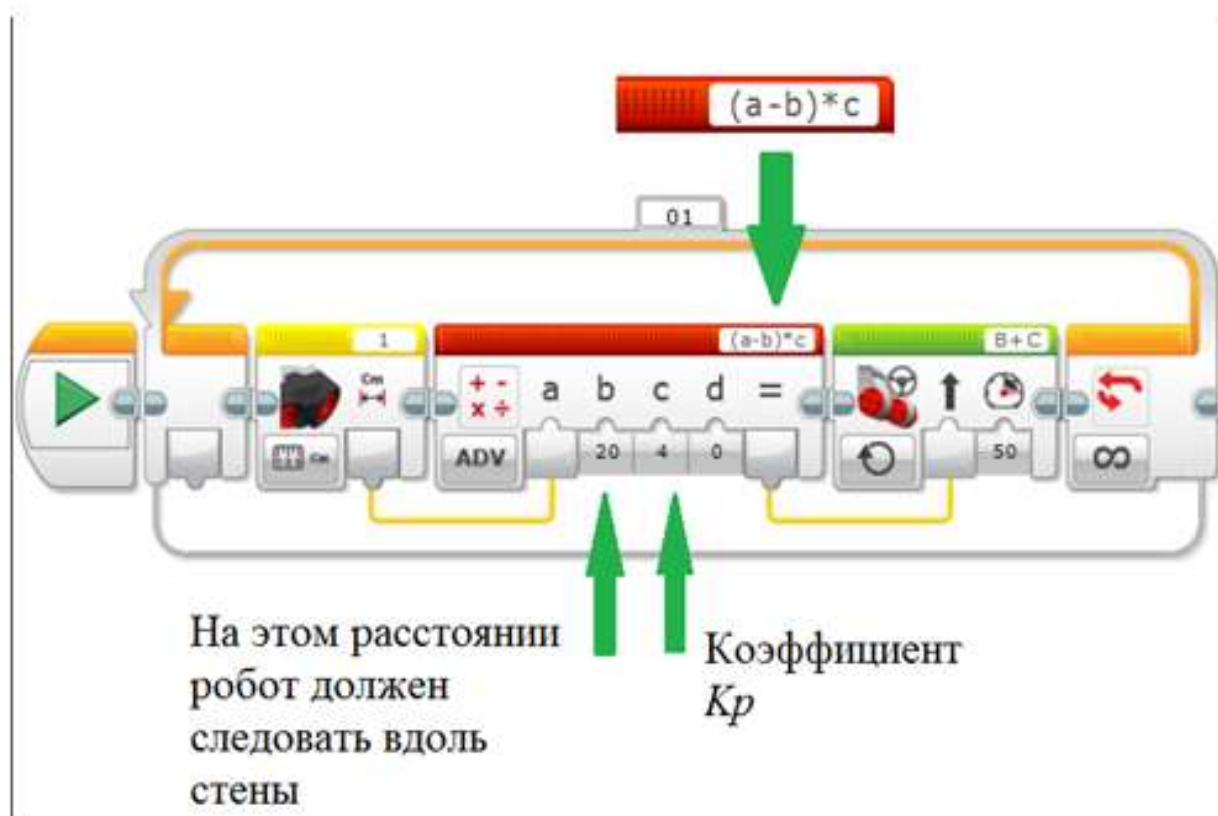
Пусть расстояние до стены становится больше 20 см. На блок рулевого управления подается положительное значение, мотор «В» начинает вращаться быстрее, пропорционально величине ошибки рассогласования. То есть, чем дальше робот отъехал от стены, тем с бóльшей скоростью он будет к ней подъезжать. Мотор «С» вращается с заданной скоростью 30 единиц.

В результате робот поворачивает направо в сторону стены, продолжая при этом движение вперед.

Если расстояние до стены становится меньше 20 см, то на блок рулевого управления подается отрицательное значение, мотор «В» продолжает вращаться с заданной скоростью 30 единиц, мотор «С» вращается быстрее пропорционально величине ошибки рассогласования. Чем ближе робот подъехал к стене, тем с бóльшей скоростью он будет от нее отъезжать.

В результате робот поворачивает налево от стены, продолжая при этом движение вперед.

Программа движения робота на заданном расстоянии от стены с использованием алгоритма пропорционального управления на основе одного датчика ультразвука представлена на рис. 2.1.1.3.

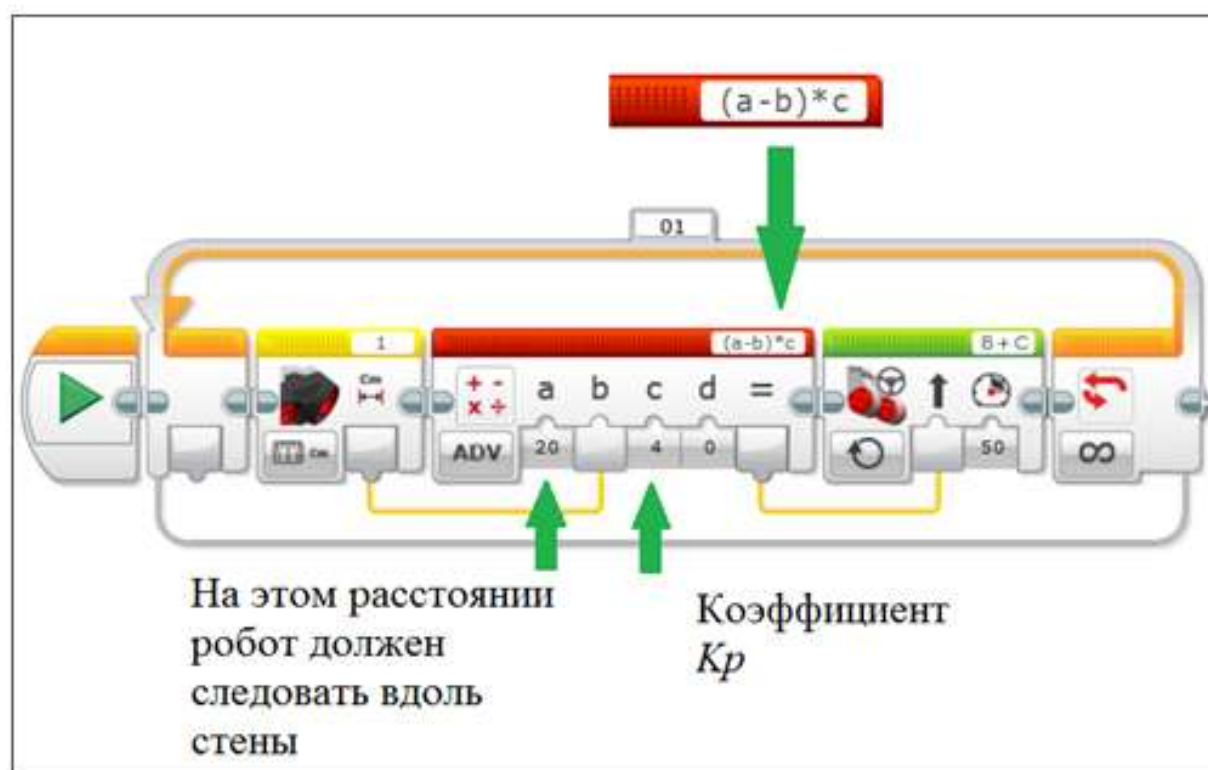


*Рисунок 2.1.1.3. Программа движения робота на заданном расстоянии слева от стены на основе одного датчика ультразвука*

Если необходимо следовать с **правой стороны** от стены, то управляющее воздействие должно быть обратным. Это можно реализовать так, как показано на рис. 2.1.1.4. Второй вариант реализации обратного управления – изменить формулу в программном блоке «Математика»:  $(b-a)*c$ .

Коэффициент  $k_p$  (параметр «с» программного блока «Математика») необходимо подбирать с учетом конструкции робота, покрытия поля и отражающей способности стены. Если стена будет покрыта мягким или пористым материалом (мех, вата, синтепон и др.), то

ультразвук может не увидеть стену или показать значения, значительно превышающие истинные!



*Рисунок 2.1.1.4. Программа движения робота на заданном расстоянии справа от стены на основе одного датчика ультразвука*

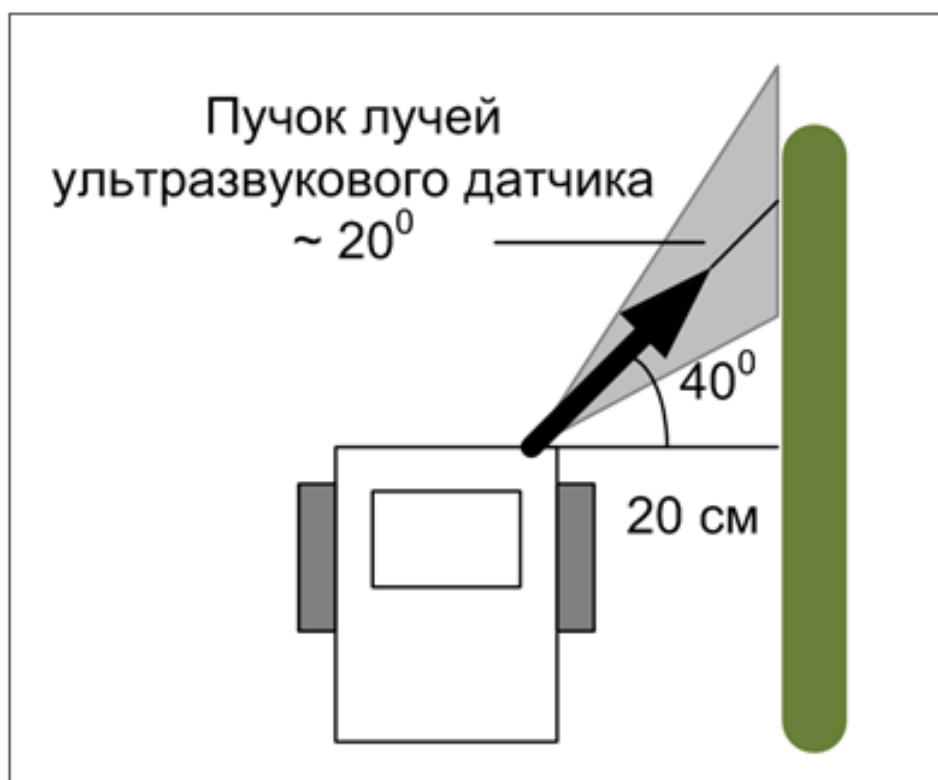
Чем меньше коэффициент, тем более плавными будут движения, но повороты робот будет проходить под большими углами.

Если необходимо следовать вдоль стены на определенном расстоянии от нее и преодолевать внутренние углы с расположенным под углом датчиком ультразвука, то необходимо опытным путем определить длину диагонали, измеряемую датчиком ультразвука. Из-за разброса пучка лучей ультразвукового датчика при необходимости движения на расстоянии 20 см от стены



параметр «b» программного блока «Математика» нужно задавать как минимум вдвое больше (рис. 2.1.1.5)!

Из-за этой особенности пропорциональный коэффициент невозможно рассчитать математически, как будет показано в п. 2.2.1 при рассмотрении движения по заданному курсу с помощью датчика гироскопа, а только опытным путем. Действительно, мы не можем сказать, какое расстояние вернет датчик ультразвука при отклонении робота, например, на 2 см от стены, поскольку датчик окажется под углом.



*Рисунок 2.1.1.5. Определение расстояния при диагональном расположении датчика ультразвука*

Описанный алгоритм следования роботом на определенном расстоянии вдоль стены может применяться при прохождении лабиринтов.

### 2.1.2. Слежение за движущимся объектом. Проект «Сторожевая башня»

*Задача: робот должен найти движущийся объект с любой стороны и следить за ним. Как только расстояние до объекта станет меньше 30 см, произвести выстрел и продолжить поиск других объектов (рис. 2.1.2.1).*



*Рисунок 2.1.2.1. Конструкция робота, способного обнаружить объект, следить за ним и поражать шариками на заданном расстоянии*

## 2.2. Применение датчика гироскопа. Движение робота по заданному курсу

Цифровой датчик гироскопа предназначен для измерения угла и направления вращения, а также скорости вращения робота.

Датчик гироскопа определяет движение вокруг одной оси вращения. Это направление указано стрелкой на корпусе датчика (рис. 2.2.1). Угол и направление вращения могут быть положительными или отрицательными. Вращение по часовой стрелке считается положительным, против часовой стрелки – отрицательным. Единица измерения скорости – градусы в секунду.

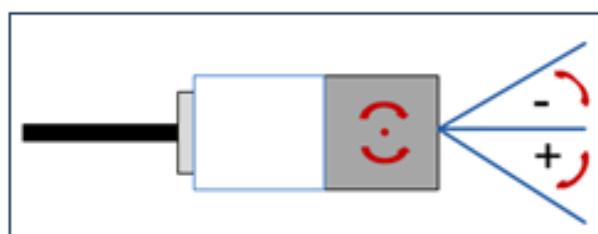


Рисунок 2.2.1. Направления вращения

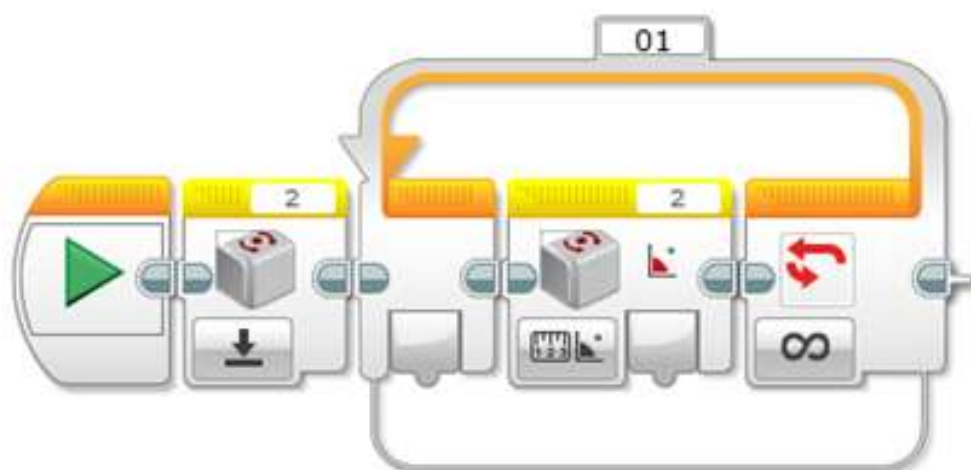
Отличие датчика гироскопа от других датчиков состоит в том, что, будучи включенным при длительном нахождении робота в неподвижном состоянии, значение угла не остается постоянным, а «дрейфует». Поэтому чем больше времени проходит от начала первого обращения к датчику гироскопа до чтения показаний, тем менее точными становятся результаты за счет систематического накопления ошибки. Поэтому перед началом каждого

измерения всегда необходимо производить обнуление угла при помощи режима «Reset».

***Важно!***

*Иногда (достаточно часто!) при работе с датчиком гироскопа можно наблюдать следующее: при запущенной программе робот находится в неподвижном состоянии, а значения постоянно увеличиваются, скорость увеличения составляет более 1 градуса в секунду!*

*Если ваш робот, управляемый значениями датчика гироскопа, ведет себя неадекватно, запустите приведенную ниже программу. При этом робот должен оставаться подключенным к компьютеру посредством любой связи (USB, Bluetooth или WiFi):*



*Обратите внимание на показания датчика гироскопа в правом нижнем углу экрана компьютера:*

### 2.4.3. Реализация плавного разгона и торможения робота

При выполнении плавного разгона или торможения робот совершает равноускоренное или равнозамедленное движение, скорость которого описывается известной формулой:

$$V=V_0+at, \quad (1)$$

где  $V$  – текущая скорость робота;

$V_0$  – начальная скорость робота;

$a$  – ускорение робота;

$t$  – время, прошедшее с момента начала движения.

Если посмотреть на приведенную формулу с точки зрения пропорционального управления, мы видим стандартную многократно описанную выше зависимость, в которой мощность управляющего воздействия на моторы зависит от времени движения, а в качестве пропорционального коэффициента служит величина ускорения робота.

Мы приведем решение двух задач, позволяющих роботу плавно начать движение (или торможение) за заданное время или на заданном расстоянии.

*Задача 1: роботу необходимо совершить плавный разгон со скорости (мощности) 0 единиц до скорости (мощности) 50 единиц за 2 секунды.*

### 2.5.2. Имитация гелиотропизма – поворота растения в сторону солнца. Проект «Робо-подсолнух»

Гелиотропизм – это способность растений принимать определенное положение под влиянием солнечного света. Впервые данное явление было описано Леонардо да Винчи в его ботанических исследованиях.

В большей или меньшей степени почти все цветы являются гелиотропными, но сильнее всего этот эффект прослеживается у подсолнуха, изменение положения которого в течение дня особенно заметно.

Попробуем и мы создать умный робо-подсолнух, который будет постоянно отслеживать окружающую освещенность и поворачиваться в сторону наиболее яркого света.

*Задача: создать робота, имитирующего эффект гелиотропизма, т.е. изменяющего положение в направлении наиболее яркого света.*

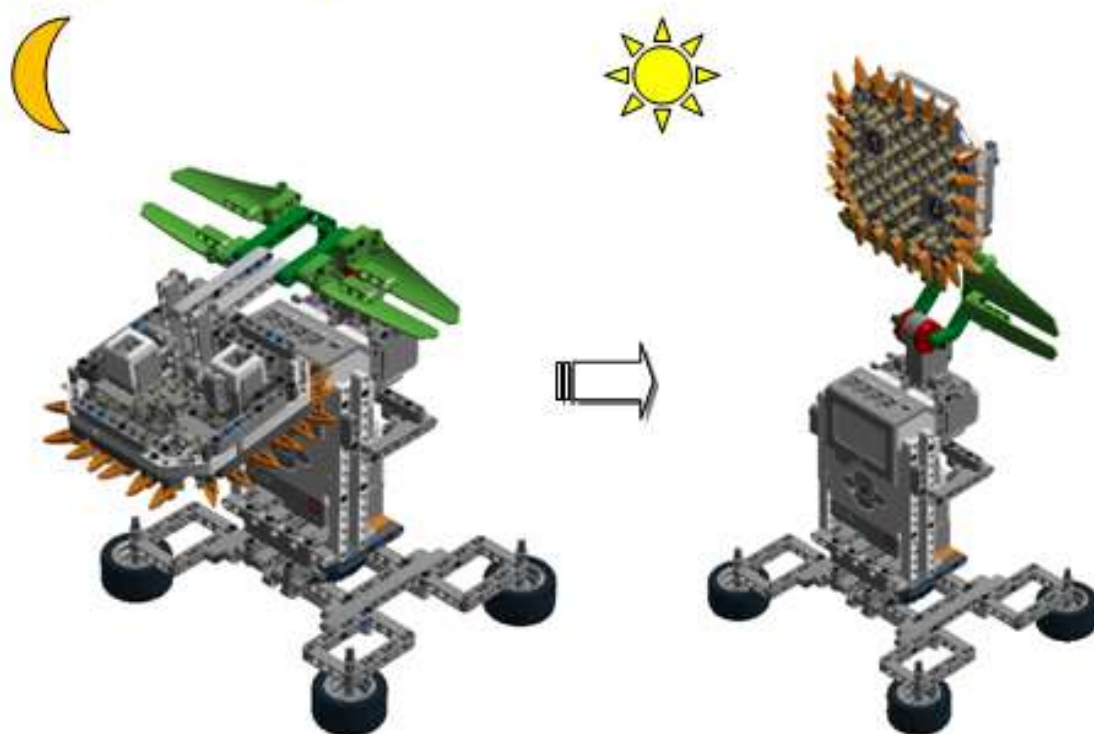
В качестве «солнца» используйте фонарь.

Для решения задачи соберите конструкцию робота, приведенную на сайте [www.ev3book.ucoz.ru](http://www.ev3book.ucoz.ru). На рис. 2.5.2.1 приведено изображение робота-подсолнуха.

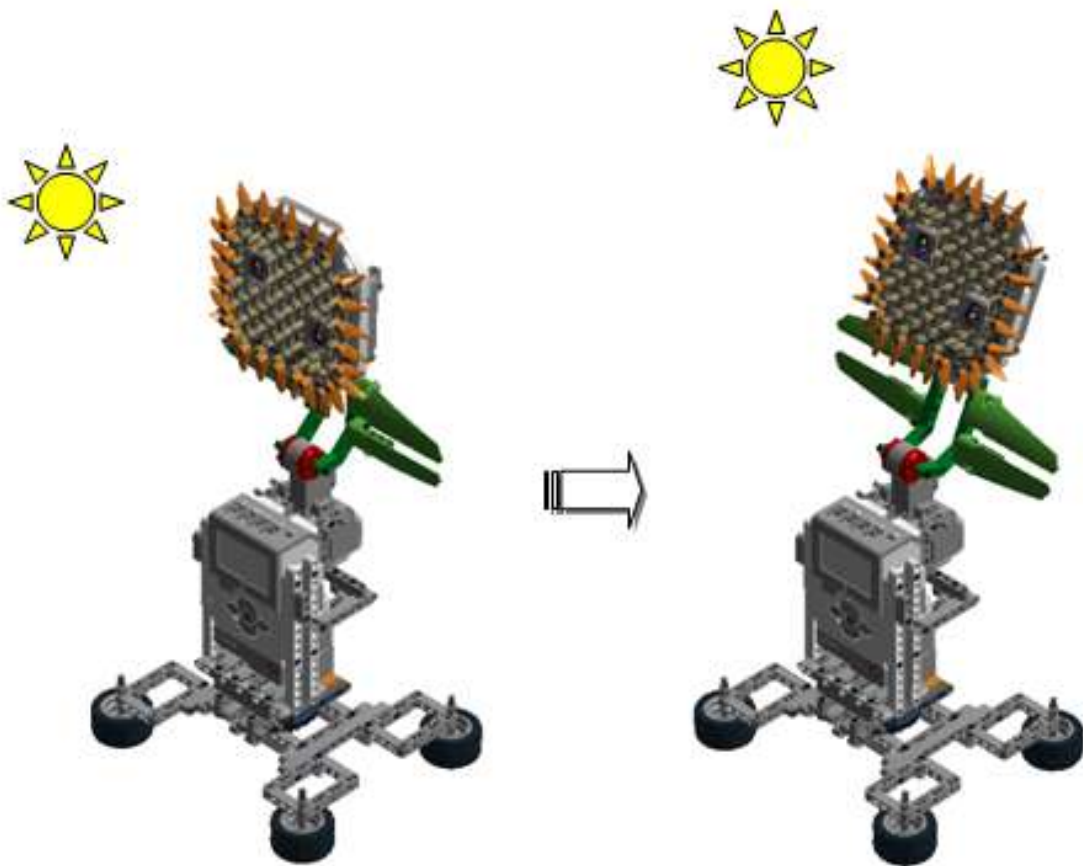
очень яркого освещения – 100. Таким образом, выходной диапазон полностью удовлетворяет требованиям для реализации пропорционального управления.

Представим два варианта решений задачи. В первом случае цветок будет поворачиваться в сторону более яркого света только в одном направлении по горизонтали (в природе подсолнух в течение дня поворачивает соцветие с востока на запад). Во втором случае мы добавим небольшие движения и в вертикальной плоскости для того, чтобы цветок мог поднимать соцветие в зависимости от высоты солнца над горизонтом.

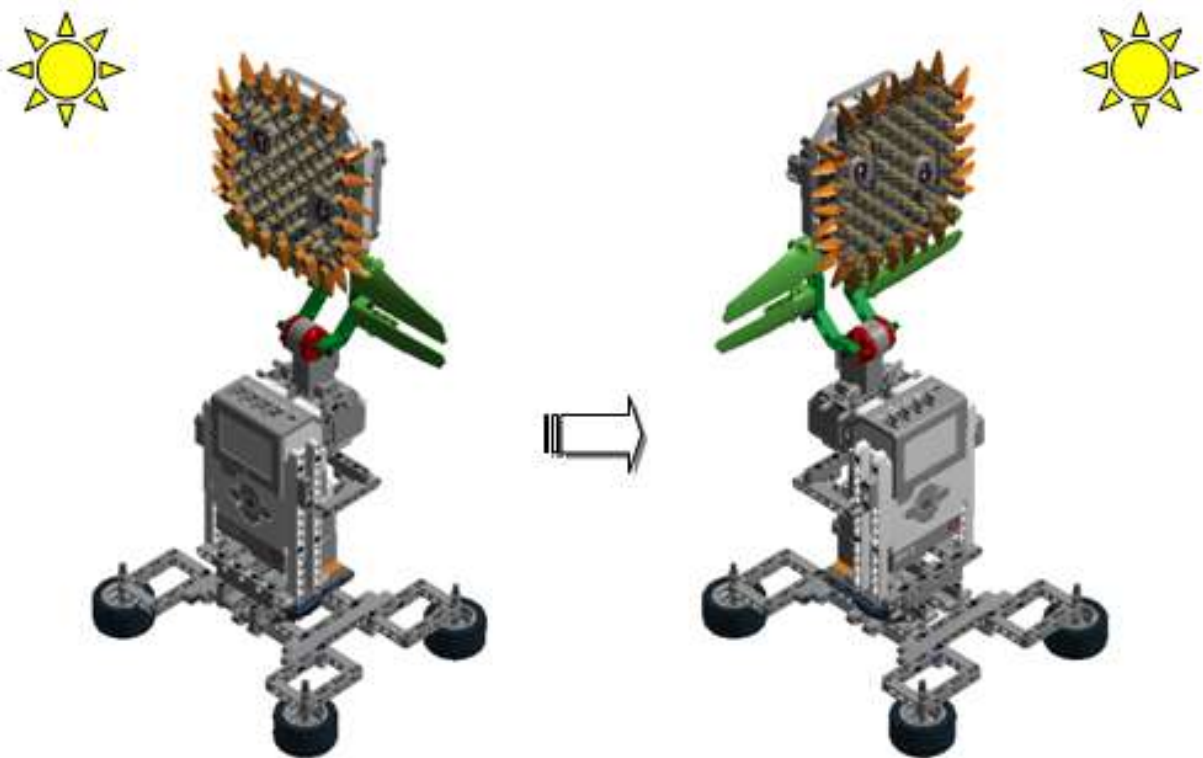
В обоих вариантах мы воспользуемся готовым решением, позволяющим удерживать тяжелое соцветие в вертикальном положении благодаря программе пропорционального управления на основе датчика вращения (см. п. 2.4.2).



*Утро. Робо-подсолнух просыпается*



*День. Робо-подсолнух поднимает соцветие к солнцу*



*Вечер. Робо-подсолнух повернулся в другую сторону*



### 3. РАСЧЕТ КОЭФФИЦИЕНТОВ ПИД-РЕГУЛЯТОРА МЕТОДОМ ЦИГЛЕРА-НИКОЛСА<sup>3</sup>

Одна из самых серьезных задач, которую необходимо решать при создании ПИД-регулятора – выбор пропорционального, дифференциального и интегрального коэффициентов. От них зависит качество движения робота. В предыдущих разделах мы указывали рекомендованные коэффициенты для каждого вида задачи и типа датчика, описывая способы их экспериментального определения.

Однако существует методики теоретического расчёта параметров ПИД-регулятора, одна из которых предложена Циглером и Николсом (J.G. Ziegler, N.B. Nichols) в 1942 году, она получила название метода Циглера-Николса. Эта методика, благодаря своей простоте, до сих пор часто используется на практике, хотя в настоящее время появилось множество более точных методов.

Циглер и Николс предложили два метода настройки ПИД-регуляторов. Один из них основан на параметрах отклика объекта на единичный скачок; второй метод основан на частотных характеристиках объекта управления. Математически оба метода эквивалентны, но

---

<sup>3</sup> Глава написана совместно с А.Е. Васильевым, кандидатом технических наук, доцентом кафедры электроники и микроэлектроники, старшим научным сотрудником научно-исследовательского сектора ФГБОУ ВПО «МГТУ им. Г.И. Носова».

Удаляем предыдущее содержание файла "Test". Если этого блока не будет, при каждом запуске программы новые данные будут записываться в конец старых данных. Обнуляем таймер.

Проводим корректировку показаний датчиков: в нашем случае датчик, подключенный к порту 2, всегда показывает результат на 3 единицы меньше датчика, подключенного к порту 3.

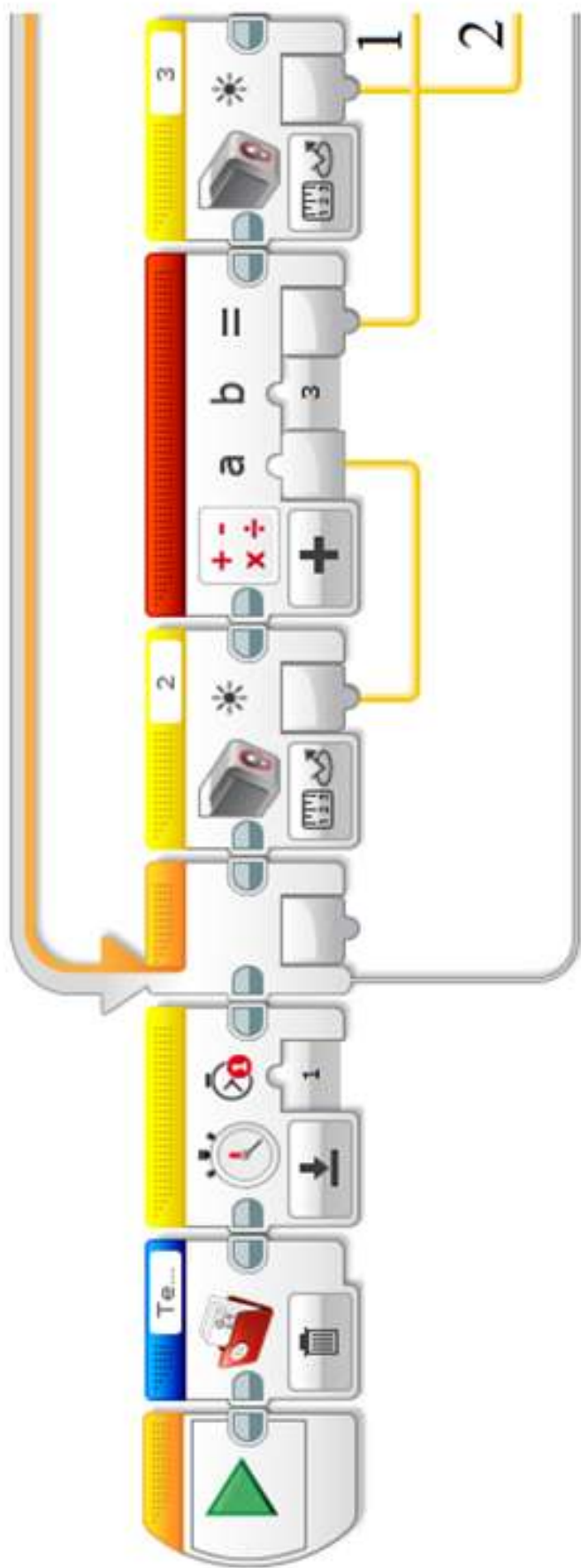


Рисунок 3.2 (начало). Программа для фиксирования реакции управляющего воздействия на моторы в момент подачи единичного возмущающего воздействия

Время	Управление	Управляющие нормализованные значения
0,015	0	0,00
0,023	0	0,00
0,032	0	0,00
0,041	0	0,00
0,054	2	0,03
0,064	4	0,06
0,074	3	0,05
0,085	5	0,08
0,094	19	0,30
0,105	43	0,67
0,115	56	0,88
0,127	60	0,94
0,138	64	1,00
0,150	63	0,98
0,160	63	0,98
0,169	61	0,95
0,178	61	0,95
0,200	62	0,97

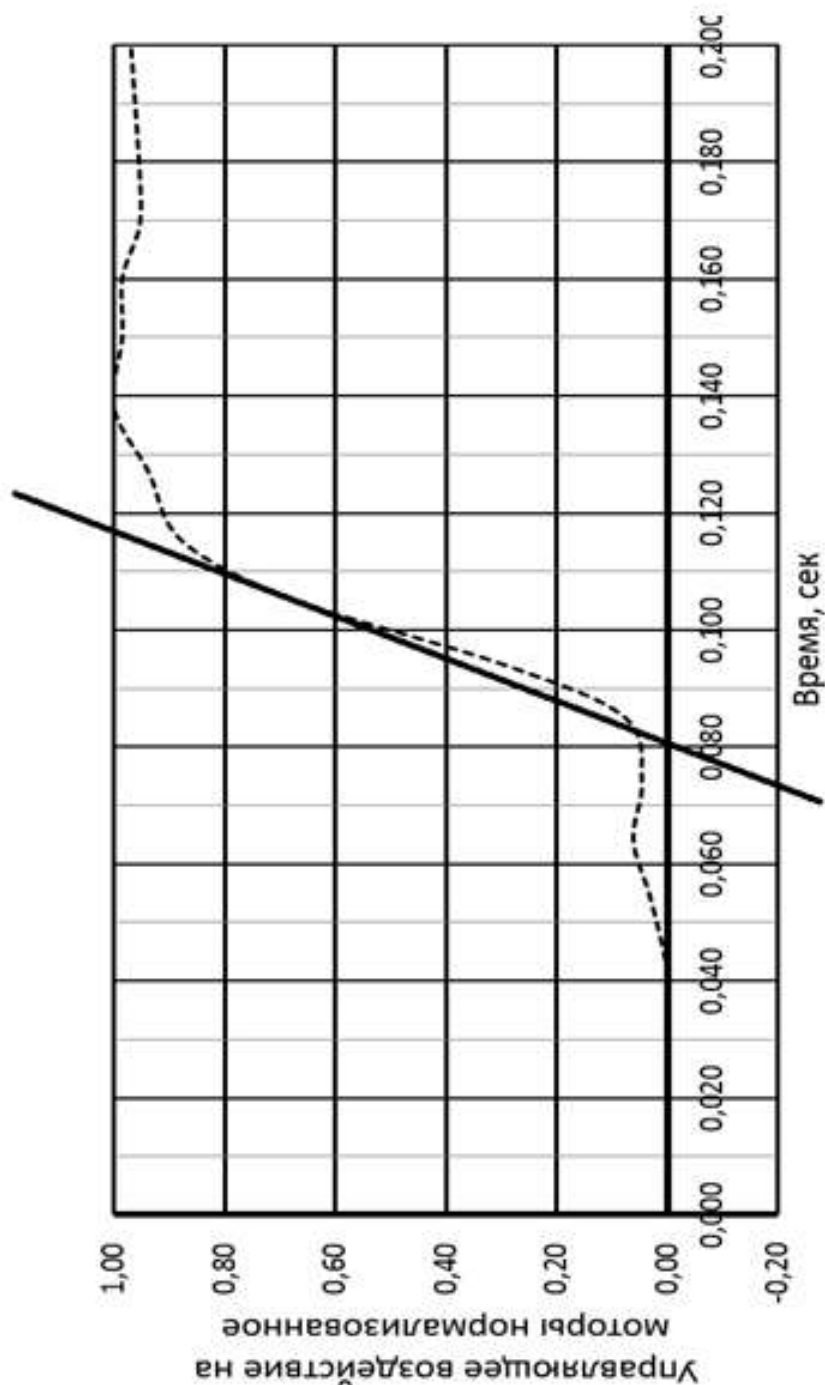


Рисунок 3.5. Нормализованные значения управления

В табл. 3.2 представлены полученные нами значения.

Таблица 3.2.

Полученные значения коэффициентов и времени интегрирования и дифференцирования регулятора по методу Циглера-Николса

Параметр	$k_p$	$T_i, мс$	$T_d, мс$
Значение	1,7	50	28,2

Таблица 3.3.

Формулы для расчета интегральных и дифференциальных коэффициентов по методу Циглера-Николса

Параметр	$k_p$	$k_i$	$k_d$
Значение	$1,2/a$	$k_p/T_i$	$T_d \times k_p$

В табл. 3.4 представлены полученные нами значения.

Таблица 3.4.

Полученные значения интегральных и дифференциальных коэффициентов по методу Циглера-Николса

Параметр	$k_p$	$k_i$	$k_d$
Значение	1,7	0,03	47,9

Подставим полученные коэффициенты в программу движения по линии (рис. 3.6) и запустим работа.