

ИНФОРМАТИКА,
ВЫЧИСЛИТЕЛЬНАЯ ТЕХНИКА
И УПРАВЛЕНИЕ

INFORMATICS,
COMPUTER ENGINEERING
AND CONTROL

УДК 007.52

DOI: 10.17212/1814-1196-2018-1-103-116

Методика разработки программы управления роботом для робота-манипулятора Delta^{*}

Д.С. КОЛТЫГИН¹, И.А. СЕДЕЛЬНИКОВ²

¹ 665709, РФ, г. Братск, ул. Макаренко, 40, Братский государственный университет, кандидат технических наук, доцент. E-mail: kds@brstu.ru

² 665709, РФ, г. Братск, ул. Макаренко, 40, Братский государственный университет, доцент. E-mail: Ohtargil@yandex.ru

В статье рассмотрен вопрос разработки программного обеспечения для роботов-манипуляторов на примере робота-манипулятора Delta. Предложена методика и алгоритмы проектирования программного обеспечения для роботов-манипуляторов не требующая больших затрат по времени и дополнительного специализированного программного обеспечения. Обоснована причина создания данного программного обеспечения, описана его структура и структура его модулей. Представлены алгоритмы ручного и автоматического управления, а также алгоритм управления с помощью линейных пространственных координат вместо угловых координат робота.

В качестве основы для разработки выбран язык программирования Visual Basic (Visual Studio 2013) корпорации Microsoft. При создании программы использовалось как можно меньше специальных библиотек и других программных компонентов, что даёт большую универсальность при применении обновлений. В статье приведены обобщенная функциональная схема робототехнического комплекса, алгоритм разработки программы управления, структурная схема программного обеспечения Delta v. 1.0, алгоритмы ручного и автоматического управления, а также формы программы для различных режимов работы. В качестве примера показаны фрагменты кода таких функций, как работа с изображением, цикловая обработка, автоматическое управление. В результате были исправлены некоторые недостатки существующей программы, созданы новые режимы работы и новые команды, которые расширяют возможности управления роботами. Программа позволяет строить сложные задачи, включающие в себя циклы, множественные повторы и динамические переходы к различным частям управляющей программы. Получена база для создания новых функций и программ управления роботами.

Ключевые слова: программирование роботов, робот, робототехника, программа, манипулятор, система управления, методы, алгоритм

ВВЕДЕНИЕ

Современные роботы и робототехнические комплексы (РТК) продолжают совершенствоваться, происходит развитие всех их составляющих, к ним относятся в основном механика и кинематика, системы управления, программы управления (рис. 1).

^{*} Статья получена 29 сентября 2017 г.

Очевидно, что развитие систем и программ управления за последнее время намного существеннее, чем изменения в механике и кинематике современных роботов. За последние 20 лет механические роботы не потерпели особых изменений, чего нельзя сказать о системах управления и программном обеспечении (ПО) в данной области и в целом.

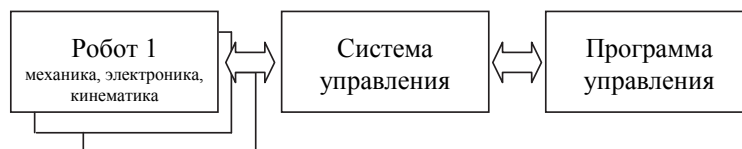


Рис. 1. Обобщенная функциональная схема РТК

Развитие микропроцессоров привело к минимизации, удешевлению, доступности, универсальности систем управления роботами, а соответственно и появлению новых программ управления. Появилась необходимость в разработке программного обеспечения к системам управления нового поколения и в модернизации существующих. Современные системы управления базируются на микропроцессорной основе, универсальны, поддерживают различные языки программирования и стандарты ввода-вывода. Поэтому разработка программ управления для таких систем перешла на новый уровень, где разработчик сам выбирает программные продукты и системы разработки. Тем более что существует широкий выбор программных продуктов для разработки программ управления, они достаточно развиты, включают весь необходимый инструмент, позволяют оперативно разрабатывать необходимый программный продукт. Это визуализация, удаленный доступ, ввод-вывод через порты, работа с видеокамерой и звуком, и т. д.

Методика разработки программ управления (выбор среды разработки, структура построения, алгоритмы функционирования программ управления) мало описана и является актуальной задачей.

1. ПОСТАНОВКА ЗАДАЧИ

Авторами предлагается методика разработки программы управления для РТК на базе робота-манипулятора Delta, результатом чего является более функциональная, расширяемая, универсальная программная среда управления по сравнению с существующей. Разработанная программа включает в себя локальное и удаленное управление, визуализацию с видеопотоком, ручное и командное управление (автоматическое), организацию циклов выполнения и т. д.

2. МЕТОДИКА РАЗРАБОТКИ ПРОГРАММЫ УПРАВЛЕНИЯ

Методика разработки включает в себя алгоритм (рис. 2), состоящий из следующих блоков: функции системы, структура, выбор программ и модулей разработки.

1. Функции, которые должна выполнять разрабатываемая система (локальное и удаленное управление, передача потокового видео, автоматическое управление и т. д.).

2. Разработка структуры программы, выделение основных модулей и разбивка их на отдельные блоки согласно требуемым функциям.

3. Для написания программы выбирается подходящий язык программирования, а также набор необходимых библиотек, драйверов и модулей требуемых для выполнения функций.

4. Если среда разработки полностью удовлетворяет требованиям, то можно приступать к созданию кода; если такого соответствия нет, то требуется осуществить поиск другой среды.

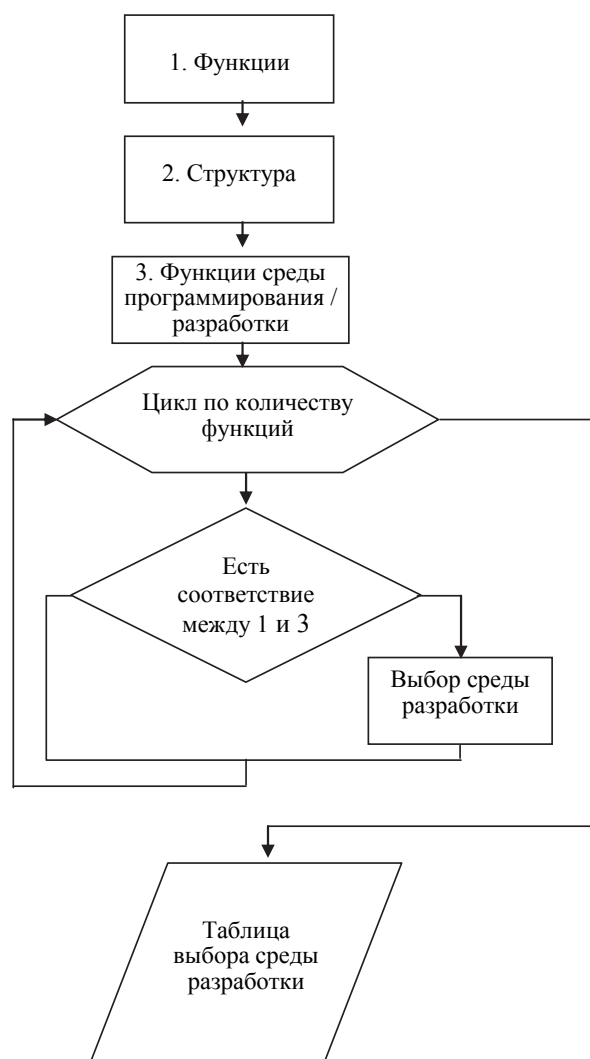


Рис. 2. Алгоритм разработки программы управления

Программа Delta v 1.0 включает в себя несколько модулей. При создании программы использовалось как можно меньше специальных библиотек и других программных компонентов, что дает большую универсальность при применении обновлений.

Программный код написан на языке Visual Basic (Visual Studio 2013). Использован NET.Framework 3.5. В разных версиях использованы разные

библиотеки System, соответственно по-разному организована работа с портами либо она отсутствует полностью, поэтому он должен быть установлен на ПК, подключенном к управляющему устройству.

Драйверы CDM от FTDI обеспечивают функциональность D2XX и VCP через единый пакет драйверов на компьютерах под управлением операционных систем Windows. Данный пакет драйверов служит для работы с устройством, подключенным по USB, как с COM-портом.

МОДУЛЬ ПОДКЛЮЧЕНИЯ К УПРАВЛЯЮЩЕМУ УСТРОЙСТВУ

Управляющее устройство подключено к ПК с помощью USB-порта, а подача команд происходит через COM-порт (особенности управляющего устройства). Для корректного выполнения сопряжения используется специальный драйвер, производящий эмуляцию локального COM порта и передачу данных через него на USB. В программе для работы с COM портом подключается библиотека System.IOPorts. Ошибки подключения обрабатываются системным обработчиком ошибок.

МОДУЛЬ ВЫВОДА ИЗОБРАЖЕНИЯ

Отображение состояния робота-манипулятора осуществляется с помощью USB-камеры, подключенной к этому же ПК. Для работы с графикой необходимо использовать NET.Framework 3.5. Видео организовано заполнением специального окна статичными картинками, получаемыми с камеры, с заданным интервалом, что позволяет использовать изображение с камеры и варьировать частоту кадров под задачи и разгрузить каналы передачи и ресурсы ПК. При необходимости возможно организовать подключение нескольких камер.

МОДУЛЬ ВЫВОДА

Модуль вывода тесно связан с модулем подключения и использует те же библиотеки. Вывод осуществляется отправкой сформированной команды на COM-порт. Интерпретация и обработка команд происходит в устройстве управления. Там же происходит обработка ошибок.

МОДУЛЬ ПОДГОТОВКИ И ФОРМИРОВАНИЯ КОМАНД

На подготовительном этапе каждой переменной, соответствующей звену робота, присваиваются значения в виде требуемого количества шагов двигателя с учетом передаточных чисел редуктора. Для этого производятся все необходимые расчеты, такие как пересчет значений treckbar и перевод текстовых значений из textbox в числовые значения, и т. д. Формирование команды заключается в заполнении шаблона, отформатированного под требования управляющего устройства, с подготовленными значениями.

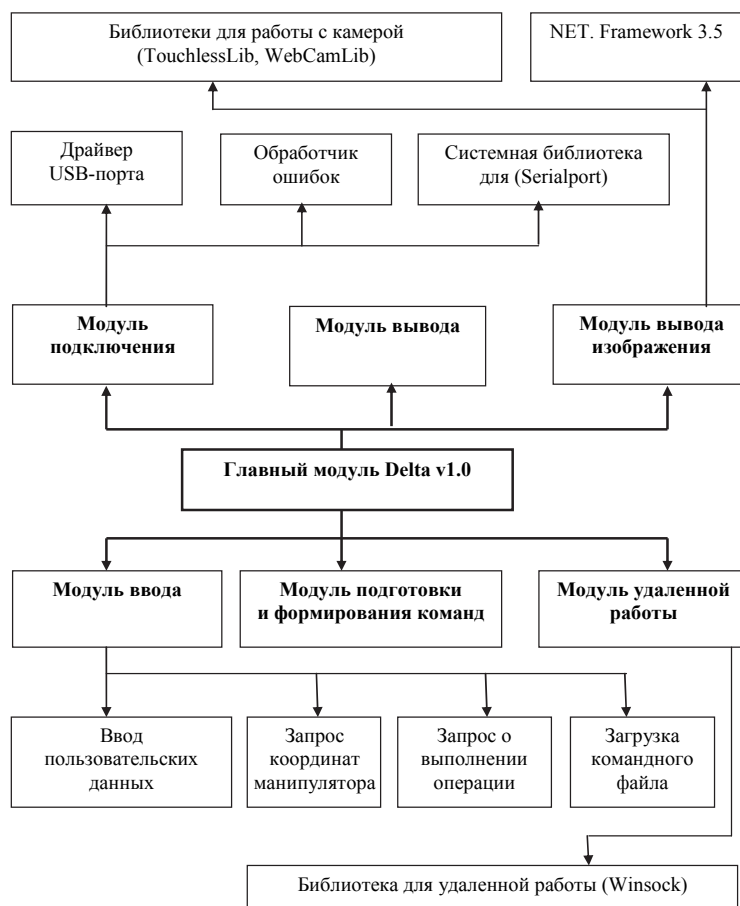


Рис. 3. Структурная схема ПО Delta v. 1.0

МОДУЛЬ ВВОДА

Поскольку используется большое количество входных данных, то и применяется несколько вариантов их получения, таких как ввод текстовой информации в специальные поля (Textbox, RichTextBox), получение информации из объекта TrackBar путем вычисления значения в зависимости от положения бегунка, заполнение многострочного текстового поля (RichTextBox) информацией из файла. Для данных операций не используется никакое специальное ПО. При получении информации с управляющего устройства робота используются те же библиотеки, что и для вывода.

МОДУЛЬ УДАЛЕННОЙ РАБОТЫ

Удаленное управление является обязательным условием для любой системы управления роботом. В данной программе реализована клиент-серверная архитектура в едином приложении. Для осуществления подключения и передачи используется библиотека Winsock. Для передачи используется два отдельных потока: TCP для управления манипулятором и UDP для передачи изображения.

3. ФОРМЫ И ФУНКЦИИ ПРОГРАМНОГО ОБЕСПЕЧЕНИЯ

Программа служит для управления роботом-манипулятором Delta и поддерживает три режима: 1) ручное управление, 2) управление по координатам, 3) автоматическое управление, а также два вида работы:

1) локальное управление;

2) удаленное управление.

ПО Delta v.1.0 включает в себя четыре формы.

Главная форма программы (рис. 4) служит для выбора режима управления манипулятором.

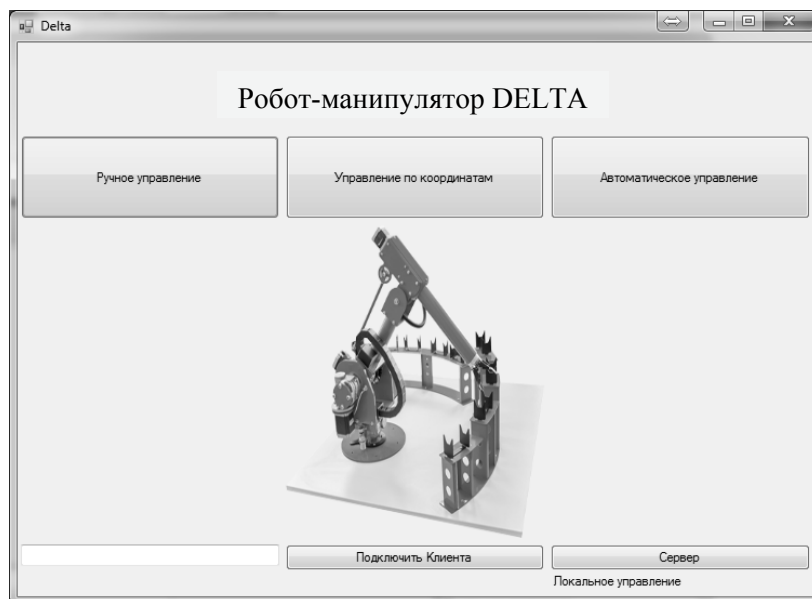


Рис. 4. Главная форма программы Delta v.1.0

Удаленное управление включает в себя два режима работы:

1) сервер (компьютер, подключенный к управляющему устройству);

2) при нажатии на кнопку «Сервер» программа переходит в режим прослушивания порта для ожидания подключения.

Клиент (удаленный компьютер)

Для подключения в режиме клиента в текстовое поле вводится адрес сервера.

При отсутствии удаленного подключения на форме отображается надпись «Локальное управление» без выделения, при удаленном – «Подключен в режиме Сервер» и «Подключен в режиме Клиент» соответственно. На других формах также будет отображаться текущее состояние.

При подключении в режиме сервера управляющие кнопки будут неактивными для исключения возможности вмешательства в процесс.

Наблюдение за движением робота происходит с помощью USB-камеры, что позволяет очень точно позиционировать схват манипулятора.

Загрузка видео с камеры состоит из двух частей.

- Установка параметров изображения

```
Public Touchless As New TouchlessLib.TouchlessMgr
```

```
Public Webka As TouchlessLib.Camera = Touchless.Cameras.Item(0)
```

```
Dim command As String
```

```
Delegate Sub SetTextCallback(ByVal [text] As String)
```

```
Private Sub Form1_Load(sender As System.Object, e As System.EventArgs)
```

```
Handles MyBase.Load
```

```
If Touchless.Cameras.Count <> 0 Then
```

```
    Webka = Touchless.Cameras.Item(0)
```

```
    Touchless.CurrentCamera = Webka
```

```
    Touchless.CurrentCamera.CaptureHeight = 480
```

```
    Touchless.CurrentCamera.CaptureWidth = 640
```

```
End If
```

- Получение изображения

```
Private Sub Timer1_Tick(sender As Object, e As EventArgs) Handles Timer1.Tick
```

```
    PictureBox1.Image = Touchless.CurrentCamera.GetCurrentImage
```

Ручное управление, алгоритм функционирования которого представлен на рис. 5, а форма – на рис. 6, осуществляется перемещением бегунков, соответствующих требуемым координатам, кнопками «Старт / Стоп» и «Сжать / Разжать» для задания состояния схвата.

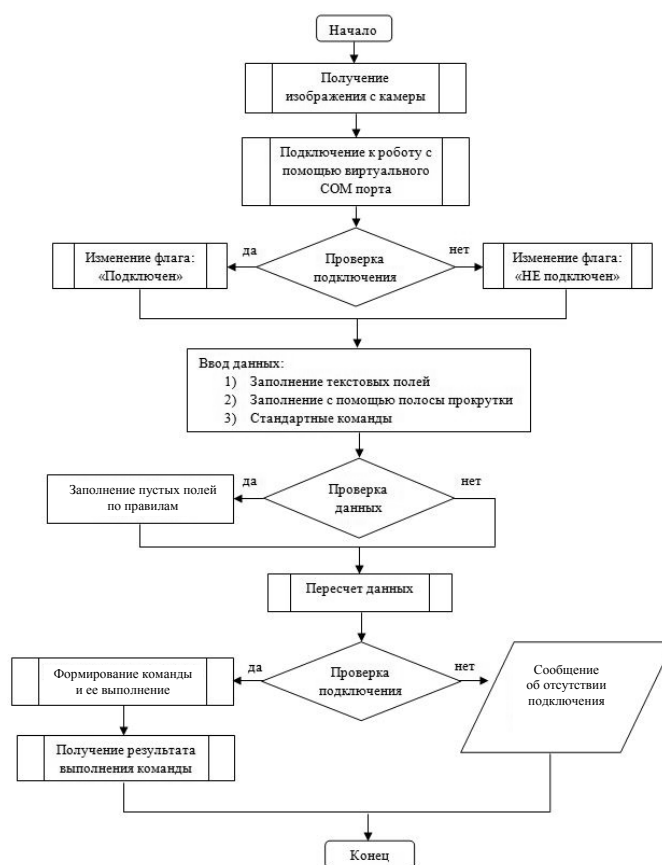


Рис. 5. Алгоритм ручного управления

Создано выпадающее меню начальных положений как по всем координатам, так и по отдельным. В специальном окне выводятся результаты выполнения действий для отслеживания состояния манипулятора.

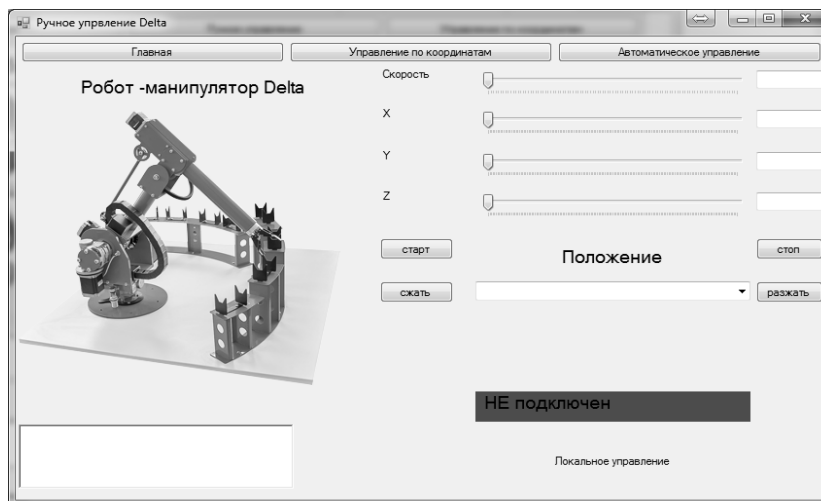


Рис. 6. Форма ручного управления программы Delta v.1.0

Режим управления манипулятором по координатам является одним из важнейших в задаче автоматизации. Он позволяет задавать перемещение не с помощью угловых координат робота, а с помощью линейных координат в декартовом пространстве, что значительно упрощает управление, так как не требуется подбирать комбинации наклона звеньев для точной установки схвата. Алгоритм перехода к линейным координатам и математическое описание приведены в статье «Аналитический и численный методы решения обратной задачи кинематики для робота Delta» [1], там же приведен пример программы.

На форме (рис. 7) после ввода координат объекта будут показаны координаты робота-манипулятора, и произойдет перемещение звена.

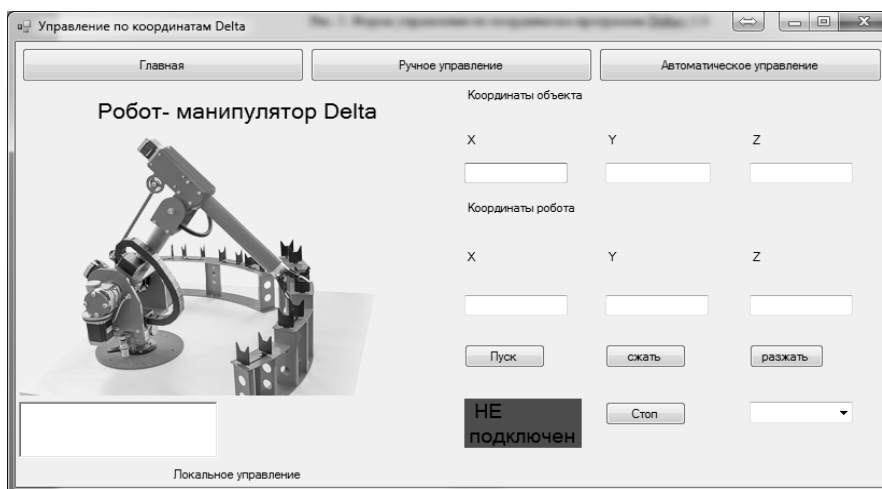


Рис. 7. Форма управления по координатам программы Delta v.1.0

Автоматическое управление, алгоритм функционирования которого представлен на рис. 8, а форма – на рис. 9, представляет собой управление по заранее написанной программе, которую можно загрузить из файла или вписать в текстовое окно.

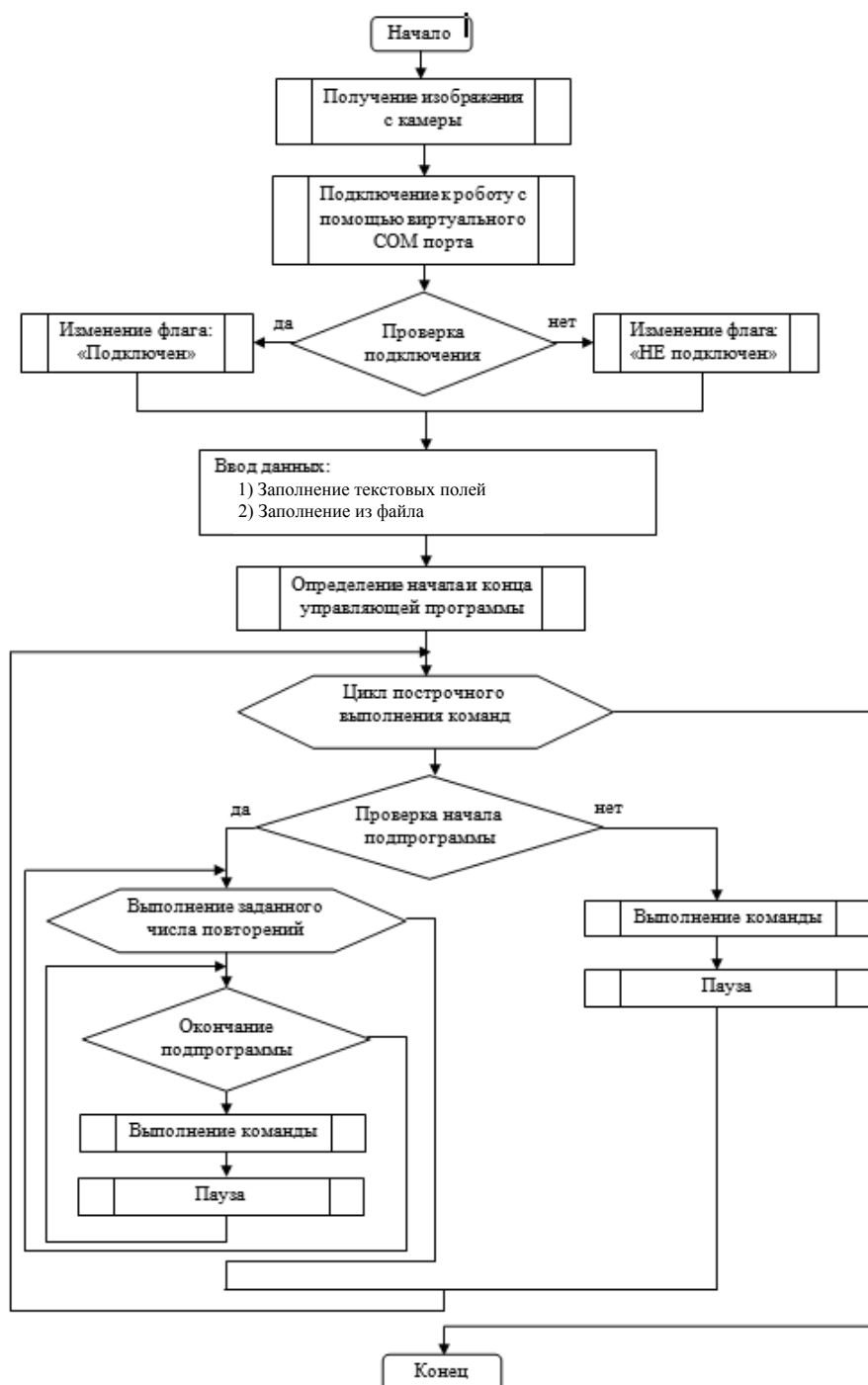


Рис. 8. Алгоритм автоматического управления

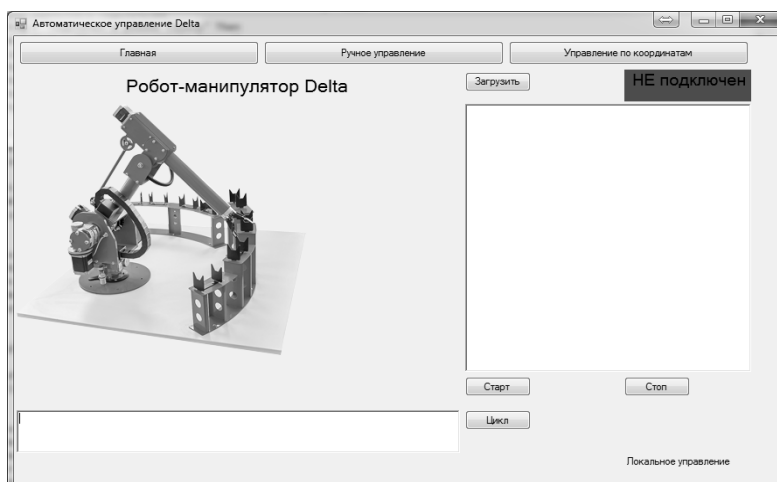


Рис. 9. Форма автоматического управления программы Delta v.1.0

Программа позволяет строить сложные задачи, включающие в себя циклы, множественные повторы и динамические переходы к различным частям управляющей программы.

Ниже приведены ключевые слова, используемые при формировании управляющей программы:

"BEGIN" – начало программы
 "FINISH" – конец программы
 "DO: " – начало подпрограммы
 "END_DO" – конец подпрограммы

Приведен пример программного кода, осуществляющий автоматическое управление:

```
Dim q As String
Dim i, b, k, r, t, m As Integer
b = RichTextBox2.GetLineFromCharIndex(RichTextBox2.Find("BEGIN"))
k = RichTextBox2.GetLineFromCharIndex(RichTextBox2.Find("FINISH"))
For i = b + 1 To k - 1
  'начало
  If (Microsoft.VisualBasic.Left(RichTextBox2.Lines(i), 3) = "DO:") Then
    r = CInt(Microsoft.VisualBasic.Right(RichTextBox2.Lines(i),
Len(RichTextBox2.Lines(i)) - 3))
    For t = 0 To r - 1
      m = 0
      Do While RichTextBox2.Lines(i + m) <> "END_DO"
        m += 1
      If RichTextBox2.Lines(i + m) <> "END_DO" Then
        q = RichTextBox2.Lines(i + m)
        SerialPort1.Write(q & vbCr)
        SleepVB(20)
      End If
    Loop
  Next
  i += m + 1
```

Цикловое управление требует задержки на выполнение команды (в программе задержка установлена в размере 20 с). Задержка осуществлена отдельной функцией. Для сокращения времени ожидания в функции паузы осуществляется проверка конца операции. После окончания выполнения каждой операции в специальное текстовое поле выводится текст, содержащий ответ порта (положение манипулятора и т. д.). Ключевое слово, определяющее завершение операции "READYR:", используется для окончания функции паузы:

```
Sub SleepVB(Seconds)
Dim Start As Double
Start = Microsoft.VisualBasic.DateAndTime.Timer
Do While Microsoft.VisualBasic.DateAndTime.Timer <
Start + Seconds
Application.DoEvents()
If RichTextBox1.Text.ToLower.Contains("READYR:")
Then
RichTextBox1.Text = ""
Exit Do
End If
Loop
End Sub
```

Пример кода управляющей программы:

```
«BEGIN ' начало программы
HOME ' переход в начальное положение
M10 ' схват разжать
DO:3 ' начало цикла. повторить 3 раза
G01X00000Y00000Z00500F00000
G01X00000Y00000Z00000F00000
END DO 'Конец цикла повторов
M11 ' схват сжать
FINISH ' конец программы»
```

ЗАКЛЮЧЕНИЕ

Разработанная программа полностью соответствует поставленной задаче. Предлагаемая методика проектирования ПО для роботов является эффективной по времени и не требует специального дополнительного ПО. Такие алгоритмы функционирования программ управления можно использовать для проектирования программного обеспечения роботов различных типов. В режиме ручного управления повышена точность задания координат, добавлен режим управления по координатам, графическая модель заменена на видео, предусмотрен контроль состояния и положения манипулятора. В режиме автоматического управления разработан новый алгоритм выполнения управляющей программы с добавлением вложенных циклов. Переход на видеоизображение открывает дополнительные возможности модернизации (реализация компьютерного зрения). Организация удаленной работы с роботом-манипулятором расширяет круг решаемых задач и повышает безопасность работы за счет отстранения от зоны движения робота. В созданном ПО воз-

можно реализовать большое количество функций: условные переходы, актуальные при подключении датчиков, полное заикливание, запись выполненных команд в файл для применения их в автоматическом режиме и т.д.

СПИСОК ЛИТЕРАТУРЫ

1. Колтыгин Д.С., Седельников И.А., Петухов Н.В. Аналитический и численный методы решения обратной задачи кинематики для робота Delta // Вестник Иркутского государственного технического университета. – 2017. – Т. 21, № 5 (124). – С. 87–95.
2. Колтыгин Д.С., Седельников И.А. Алгоритм определения обобщенных координат для робота Delta // Труды Братского государственного университета. Серия: Естественные и инженерные науки. – 2016. – Т. 2. – С. 127–130.
3. Колтыгин Д.С., Седельников И.А. Алгоритмы и программы робототехнического комплекса // Труды Братского государственного университета. Серия: Естественные и инженерные науки. – 2015. – Т. 1. – С. 93–96.
4. Колтыгин Д.С., Седельников И.А. Алгоритмы и программы робототехнического комплекса // Труды Братского государственного университета. Серия: Естественные и инженерные науки. – 2014. – Т. 1. – С. 281–284.
5. Робот PASCAL DELTA 1-3X-USB+: учебное пособие. – Челябинск: ООО «Уральские станки», 2015. – 47 с.
6. Иконников А.В. Микропроцессорная система управления параллельной работой двух промышленных роботов // XXII Региональная конференция молодых ученых Волгоградской области: тезисы докладов / редкол.: А.В. Навроцкий (отв. ред.) и др. – Волгоград, 2017. – С. 80–81.
7. Zimin G.A., Mordvinov D.A. Visual dataflow language for educational robots programming // Труды Института системного программирования РАН. – 2016. – Т. 28, № 2. – С. 45–62.
8. Афонин В.Л. Интеллектуальные технологические системы. – Saarbrücken: LAP LAMBERT, 2015.
9. Клековкин А.В. Управление учебным высокоманевренным мобильным роботом с омниколесами // Молодые ученые – ускорению научно-технического прогресса в XXI веке: сборник материалов III Всероссийской научно-технической конференции аспирантов, магистрантов и молодых ученых с международным участием. – Ижевск, 2015. – С. 181–186.
10. Вакуленко А.С., Сартаков В.Д. Алгоритмы и программы микроконтроллерного управления промышленным роботом // Вестник Иркутского государственного технического университета. – 2015. – № 5 (100). – С. 12–17.
11. Павловский В.Е., Павловский В.В. Математическая модель двумерной гомогенной стаи роботов // Искусственный интеллект и принятие решений. – 2015. – № 4. – С. 62–71.
12. Давыдов О.И., Платонов А.К. Алгоритм управления дифференциальным приводом мобильного робота РБ-2 // Препринты ИПМ им. М.В. Келдыша. – 2015. – № 25. – С. 1–16.
13. Стебулянин М.М., Сеницын А.Г. Контурное управление манипуляционным роботом в режиме априорной неопределенности закона движения // Известия высших учебных заведений. Машиностроение. – 2011. – № 8. – С. 44–50.
14. Герасин П.В., Редько Ю.С., Мартынов В.В. Визуализация движения мехатронно-модульного робота // Вестник Донского государственного технического университета. – 2011. – Т. 11, № 5 (56). – С. 776–778.
15. Колтыгин Д.С., Седельников И.А., Павлюк Е.Ю. Определение точности позиционирования роботов-манипуляторов DELTA и OMEGA // Труды Братского государственного университета. Серия: Естественные и инженерные науки. – 2016. – Т. 2. – С. 121–126.

Колтыгин Дмитрий Станиславович, кандидат технических наук, доцент кафедры управления в технических системах факультета энергетики и автоматизации Братского государственного университета. Основное направление научных исследований – робототехника, программирование. Имеет более 20 публикаций. E-mail: kds@brstu.ru

Седельников Илья Андреевич, доцент кафедры управления в технических системах факультета энергетики и автоматизации Братского государственного университета. Основное направление научных исследований – робототехника, программирование. Имеет более 8 публикаций. E-mail: Ohtargil@yandex.ru

The methodology for developing a robot control program for the Delta robot manipulator*

D.S. KOLTYGIN¹, I.A. SEDELNIKOV²

¹ Bratsk State University, 40, Makarenko Prospekt, Bratsk, 665709, Russian Federation, PhD (Eng.), associate professor. E-mail: kds@brstu.ru

² Bratsk State University, 40, Makarenko Prospekt, Bratsk, 665709, Russian Federation, associate professor. E-mail: ohtargil@yandex.ru

The article considers the development of software for robots - manipulators in the case of the Delta robot – manipulator. The methodology and algorithms for designing software are proposed for robots which are manipulators that do not require much time and additional specialized software. The reason for creating this software, its structure and the structure of its modules is grounded. Algorithms for manual and automatic control are presented as well as a control algorithm based on linear spatial coordinates instead of the robot angular coordinates.

The Microsoft Visual Basic (Visual Studio 2013) programming language is chosen as a basis for the development. When creating the program, as few special libraries and other software components as possible were used, which makes it more versatile when updates are used. The article contains a generalized functional diagram of a robotic complex, an algorithm for developing a control program, a block diagram of the Delta v software. 1.0., algorithms for manual and automatic control, as well as program forms for different modes of operation. As an example, fragments of the code of such functions as, image processing, cyclic processing, and automatic control are shown. As a result, a number of shortcomings of the existing program were corrected, new operating modes and new commands were created that expand the robot control capabilities. The program allows you to build complex tasks, including cycles, multiple repetitions and dynamic transitions to different parts of the control program. The basis for creating new functions and robot control programs has been obtained.

Keywords: Robot programming, robot, robotics, program, manipulator, control system, methods, algorithm

REFERENCES

1. Koltygin D.S., Sedel'nikov I.A., Petukhov N.V. Analiticheskii i chislennyy metody resheniya obratnoi zadachi kinematiki dlya robota Delta [Analytical and numerical methods for solving the inverse kinematics problem for the Delta robot]. *Vestnik Irkutskogo gosudarstvennogo tekhnicheskogo universiteta – Bulletin of Irkutsk State Technical University*, 2017, vol. 21, no. 5 (124), pp. 87–95.
2. Koltygin D.S., Sedel'nikov I.A. Algoritmy opredeleniya obobshchennykh koordinat dlya robota Delta [Algorithm for determining generalized coordinates for a Delta robot]. *Trudy Bratskogo gosudarstvennogo universiteta. Seriya: Estestvennye i inzhenernye nauki – Proceedings of the Bratsk State University. Series: Natural and Engineering Sciences*, 2016, vol. 2, pp. 127–130.
3. Koltygin D.S., Sedel'nikov I.A. Algoritmy i programmy robototekhnicheskogo kompleksa [Algorithms and programs of the robotic complex]. *Trudy Bratskogo gosudarstvennogo universiteta. Seriya: Estestvennye i inzhenernye nauki – Proceedings of the Bratsk State University. Series: Natural and Engineering Sciences*, 2015, vol. 1, pp. 93–96.
4. Koltygin D.S., Sedel'nikov I.A. Algoritmy i programmy robototekhnicheskogo kompleksa [Algorithms and programs of the robotic complex]. *Trudy Bratskogo gosudarstvennogo universiteta. Seriya: Estestvennye i inzhenernye nauki – Proceedings of the Bratsk State University. Series: Natural and Engineering Sciences*, 2014, vol. 1, pp. 281–284.
5. *Robot PASCAL DELTA 1-3X-USB+*. Chelyabinsk, Ural'skie stanki Publ., 2015. 47 p. (In Russian).
6. Ikonnikov A.V. [Microprocessor control system for parallel operation of two industrial robots]. *XXII Regional'naya konferentsiya molodykh uchenykh Volgogradskoi oblasti: tezisy dokladov*

* Recerved 19 September 2017.

[XXII Regional Conference of Young Scientists of the Volgograd Region Abstracts]. Ed. by A.B. Navrotsky et al. Volgograd, 2017, pp. 80–81. (In Russian).

7. Zimin G.A., Mordvinov D.A. Visual dataflow language for educational robots programming. *Trudy Instituta sistemnogo programmirovaniya RAN – Proceedings of the Institute of System Programming of the RAS*, 2016, vol. 28, no. 2, pp. 45–62.

8. Afonin V.L. *Intellektual'nye tekhnologicheskie sistemy* [Intellectual technological systems]. Saarbrücken, LAP LAMBERT, 2015.

9. Klekovkin A.V. [Management of a highly mobile mobile robot with omnicholes training]. *Molodye uchenye – uskoreniyu nauchno-tekhnicheskogo progressa v XXI veke: sbornik materialov III Vserossiiskoi nauchno-tekhnicheskoi konferentsii aspirantov, magistrantov i molodykh uchenykh s mezhdunarodnym uchastiem* [Young scientists – acceleration of scientific and technical progress in the 21st century collection of materials of the III All-Russian Scientific and Technical Conference of postgraduate students, undergraduates and young scientists with international participation]. Izhevsk, 2015, pp. 181–186. (In Russian).

10. Vakulenko A.S., Sartakov V.D. Algoritmy i programmy mikrokontrolernogo upravleniya promyshlennym robotom [Algorithms and programs of microcontroller control of industrial robot]. *Vestnik Irkutskogo gosudarstvennogo tekhnicheskogo universiteta – Bulletin of Irkutsk State Technical University*, 2015, no. 5 (100), pp. 12–17.

11. Pavlovskii V.E., Pavlovskii V.V. Matematicheskaya model' dvumernoi gomogennoi stai robotov [Mathematical model of a two-dimensional homogeneous flock of robots]. *Iskusstvennyi intellekt i prinyatie reshenii – Artificial Intelligence and Decision Making*, 2015, no. 4, pp. 62–71.

12. Davydov O.I., Platonov A.K. Algoritmy upravleniya differentsial'nym privodom mobil'nogo robota RB-2 [A control algorithm for a differential drive of RB-2 mobile robot]. *Preprinty IPM im. M.V. Keldysha – Keldysh Institute preprints*, 2015, no. 25, pp. 1–16.

13. Stebulyanin M.M., Sinitsyn A.G. Konturnoe upravlenie manipulyatsionnym robotom v rezhime apriorno neopredelennosti zakona dvizheniya [Contouring control of a manipulation robot in the regime of a priori uncertainty of the law of motion]. *Izvestiya vysshikh uchebnykh zavedenii. Mashinostroyeniye – Proceedings of Higher Educational Institutions. Machine Building*, 2011, no. 8, pp. 44–50.

14. Gerasin P.V., Red'ko Yu.S., Martynov V.V. Vizualizatsiya dvizheniya mekhatronno-modul'nogo robota [Visualization of the motion of the mechatronic-modular robot]. *Vestnik Donskogo gosudarstvennogo tekhnicheskogo universiteta – Vestnik of Don State Technical University*, 2011, vol. 11, no. 5 (56), pp. 776–778.

15. Koltygin D.S., Sedelnikov I.A., Pavlyuk E.Yu. Opredelenie tochnosti pozitsionirovaniya robotov-manipulyatorov DELTA i OMEGA [Determining the accuracy of positioning robots robots manipulators DELTA and OMEGA]. *Trudy Bratskogo gosudarstvennogo universiteta. Seriya: Estestvennye i inzhenernye nauki – Proceedings of the Bratsk State University. Series: Natural and Engineering Sciences*, 2016, vol. 2, pp. 121–126.

Для цитирования:

Колтыгин Д.С., Седельников И.А. Методика разработки программы управления роботом для робота-манипулятора Delta // Научный вестник НГТУ. – 2018. – № 1 (70). – С. 103–116. – doi: 10.17212/1814-1196-2018-1-103-116.

For citation:

Koltygin D.S., Sedelnikov I.A. Metodika razrabotki programmy upravleniya robotom, dlya robota-manipulyatora Delta [The methodology for developing a robot control program for the Delta robot manipulator]. *Nauchnyi vestnik Novosibirskogo gosudarstvennogo tekhnicheskogo universiteta – Science bulletin of the Novosibirsk state technical university*, 2018, no. 1 (70), pp. 103–116. doi: 10.17212/1814-1196-2018-1-103-116.