

Разработка программного обеспечения для системы автоматического управления по выравниванию полуфабрикатов

М.В. Тарачков
О.В. Толстель
А.Л. Калабин

Ссылка для цитирования

Тарачков М.В., Толстель О.В., Калабин А.Л. Разработка программного обеспечения для системы автоматического управления по выравниванию полуфабрикатов // Программные продукты и системы. 2023. Т. 36. № 2. С. 327–333. doi: 10.15827/0236-235X.142.327-333

Информация о статье

Поступила в редакцию: 13.02.2023

После доработки: 20.03.2023

Принята к публикации: 22.03.2023

Аннотация. Для выравнивания полуфабрикатов на скоростной конвейерной линии разработана система автоматического управления, ключевое место в которой занимают робот-манипулятор ДР-1 и рабочий орган, осуществляющий надавливание и оказывающий вибрационное воздействие. Система создана для повышения эффективности линии и снижения доли брака. В ее состав входят бортовой микроконтроллер STM32F407VG (тактовая частота 168 МГц, 192 Кб ОЗУ, 1 Мб ПЗУ) и управляющий компьютер на базе процессора Intel Atom (4 Гб ОЗУ, 32 Гб ПЗУ). Для микроконтроллера разработано ПО на языке программирования C++14 в среде CubeIDE. Операционная система реального времени FreeRTOS используется для обработки микроконтроллером нескольких задач в псевдопараллельном режиме: управление электродвигателями, обработка данных с датчиков, взаимодействие с управляющим компьютером. Для управляющего компьютера реализовано ПО на языке программирования C++11, используется операционная система Linux Ubuntu 18.04 LTS, программная платформа ROS. Управляющий компьютер предоставляет оператору системы автоматического управления интерфейс, написанный на языке программирования Python с использованием библиотеки KivyMD. Однако наиболее интересная часть ПО – решение задач прямой и обратной кинематики для дельта-манипулятора и алгоритм выравнивания полуфабрикатов. Задачи прямой и обратной кинематики являются решенными в общем виде. Приводятся особенности их реализации для робота-манипулятора ДР-1. Для формализации алгоритма выравнивания полуфабрикатов проводились эксперименты. Было необходимо выполнить около 110 выравниваний в минуту, а это требовало поиска наиболее эффективной траектории перемещения рабочего органа. В статье приводятся результаты испытаний ПО системы автоматического управления. Рассматриваются различные случаи расположения коробок с полуфабрикатами. Оцениваются сохранность полуфабрикатов и упаковки и эффективность рассматриваемого алгоритма. Применение предложенного ПО системы автоматического управления позволило повысить производительность линии и сократить долю брака.

Ключевые слова: система автоматического управления, робот-манипулятор, промышленный робот, ПО, планирование движений

В данной статье описана разработка системы автоматического управления (САУ) на примере предприятия пищевой промышленности АПХ «Мираторг» для скоростной линии по упаковке полуфабрикатов.

Робот-манипулятор раскладывает пакеты с полуфабрикатами в коробки, которые затем упаковочная машина заклеивает. Проблема заключается в том, что пакет с полуфабрикатами может выступать за пределы коробки. В таком случае упаковочная машина работает некорректно. Чтобы избежать этого, необходимо предварительно выровнять полуфабрикаты.

В рамках исследования разработана САУ выравниванием полуфабрикатов, которая позволяет заменить труд людей, повысить производительность линии и снизить долю брака. Основным элементом САУ – робот-манипулятор со специально разработанным для решения задачи выравнивания рабочим органом. Он осу-

ществляет надавливающее и вибрационное воздействие на пакет с полуфабрикатами, в результате чего происходит выравнивание. В состав САУ также входят исполнительные механизмы и датчики. Разработанное ПО позволяет системе обрабатывать сигналы с датчиков, управлять исполнительными механизмами, осуществлять интеграцию с системами предприятия, обеспечивать интерфейс оператора. Реализация ПО велась с использованием *Robot Operating System (ROS)* – популярной программной платформы для программирования роботов [1]. В используемом ПО особого внимания заслуживают реализация планирования движений на основе задачи прямой и обратной кинематики для дельта-робота [2, 3] и реализация алгоритма выравнивания полуфабрикатов, так как разработка велась для конкретного робота-манипулятора и САУ, точного аналога которой в открытых источниках не найдено.

Разработанность темы исследования.

На предприятии группы компаний «Мираторг» (г. Калининград) в производственном процессе используется большое количество средств автоматизации, например, роботы-упаковщики, которые раскладывают охлажденную продукцию в коробки. На рассматриваемом в данной работе участке осуществляется выравнивание полуфабрикатов. Инженерами компании уже была предложена система выравнивания, однако она работала менее эффективно, чем люди. Готовой системы в области автоматизации для выравнивания коробок не нашлось.

Тем не менее были найдены САУ для схожих процессов. Например, в [4] описана система, использующая дельта-манипулятор и техническое зрение для сортировки твердых бытовых отходов. Использование дельта-манипулятора обусловлено высокой скоростью конвейерной линии. Применение технического зрения позволяет определить координаты объекта. Однако скорость перемещения коробок существенно выше скорости сортировки твердых бытовых отходов. Кроме того, коробки с полуфабрикатами одинаковые, что дает возможность использовать детектор прерывания луча для определения их положения. Это быстрее и дешевле.

В статье [5] рассматривается ПО робота-манипулятора, в котором организуется взаимодействие с бортовым микроконтроллером робота, решаются задачи прямой и обратной кинематики, представлен интерфейс пользователя. Приводятся алгоритмы работы ПО. Ключевым отличием являются различная кинематика роботов, набор оборудования и задача, для решения которой используется робот.

В исследовании [6] описывается использование программной платформы ROS для управления роботом-манипулятором с семью степенями свободы. Особое внимание уделено функционированию всей системы в режиме реального времени, что важно при работе на больших скоростях. Результаты эксперимента показывают, что ROS может эффективно управлять оборудованием робота с высокой производительностью системы управления в режиме реального времени.

Аппаратная часть САУ. Состоит из робота-манипулятора с дельта-кинематикой, рабочего органа с пневматическим вибратором, детектора коробок с полуфабрикатами, работающего по принципу прерывания луча, датчика скорости конвейера, щита управления, рамы для подвешивания робота-манипулятора над конвейерной лентой (рис. 1).

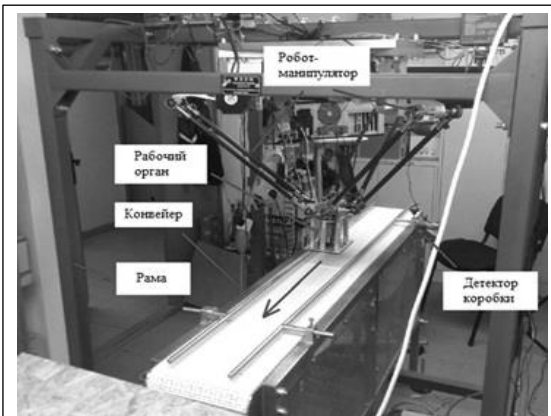


Рис. 1. Общий вид САУ

Fig. 1. A general view of the ACS

Все компоненты системы управляются при помощи одной платы. На рисунке 2 изображена схема сигнальной и силовой частей системы автоматизации. Питание осуществляется от двух блоков, которые преобразуют 220 В в 60 В для питания электродвигателей и 12 В для питания платы управления. В свою очередь, плата управления распределяет питание на концевые датчики, детектор коробки, реле управления электромагнитным клапаном. Блоки управления электродвигателями подключаются к плате управления через интерфейс STEP/DIR/ENABLE на выходы типа открытый коллектор. Они же передают управляющее воздействие на обмотки электродвигателей. Питание блоков управления осуществляется от блока питания 60 В, 5 А. Концевые датчики получают питание от платы управления и подключены на ее дискретные входы. Для управления электромагнитным клапаном используется симисторный ключ. Электромагнитный клапан управляет подачей сжатого воздуха на пневматический вибратор.

Планирование движений. Схематично дельта-манипулятор можно представить в виде двух параллельных треугольников, соединенных рычагами (рис. 3).

Точка $E_0(x_0, y_0, z_0)$ – это положение рабочего органа в системе координат робота. Углы $\theta_1, \theta_2, \theta_3$ определяют положение валов электродвигателей. Задачи прямой и обратной кинематики можно формализовать следующим образом:

$f(\theta_1, \theta_2, \theta_3) \rightarrow (x_0, y_0, z_0)$ – задача прямой кинематики, где по заданным углам поворота валов электродвигателей рассчитывается положение рабочего органа;

$f(x_0, y_0, z_0) \rightarrow (\theta_1, \theta_2, \theta_3)$ – задача обратной кинематики, где по заданному положению ра-

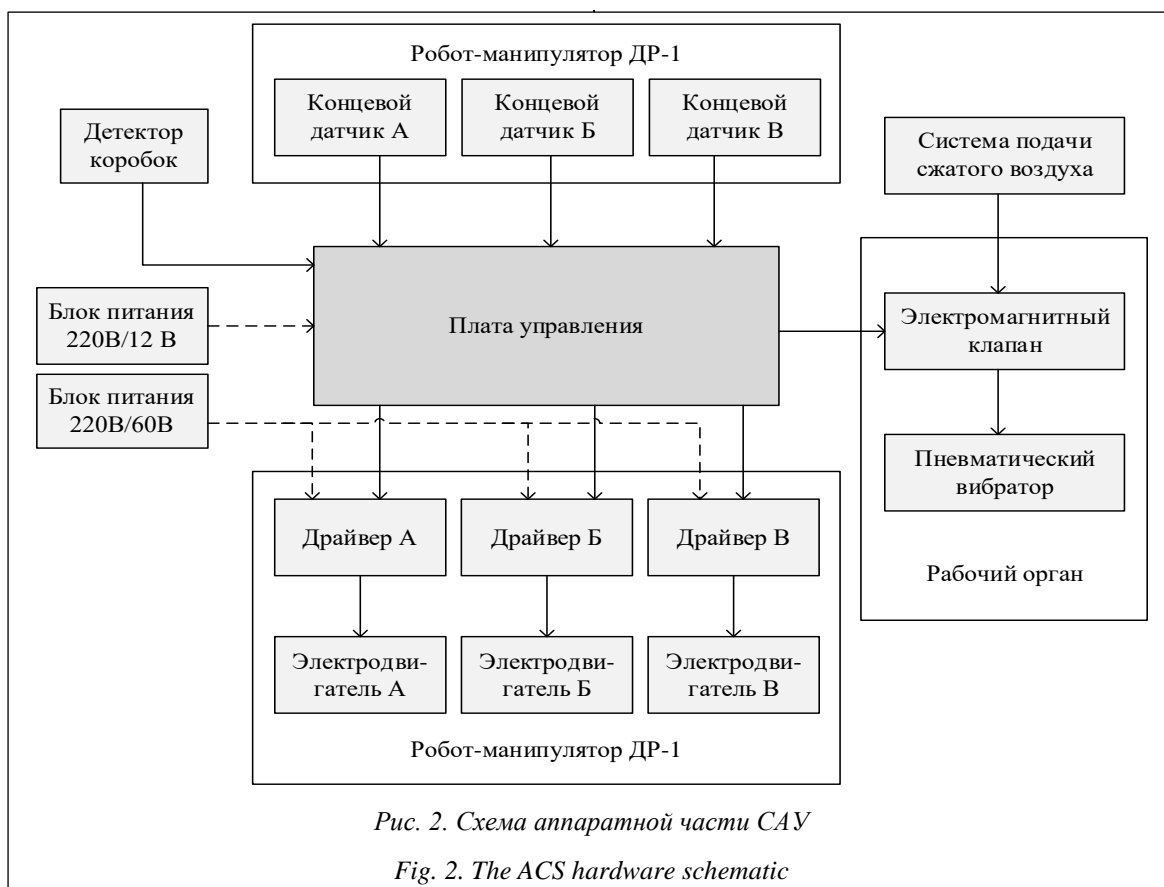


Рис. 2. Схема аппаратной части САУ

Fig. 2. The ACS hardware schematic

бочего органа вычисляются углы, на которые необходимо повернуть валы электродвигателей.

В концепции ROS реализован узел планирования движений, он обменивается сообщениями [7] с узлом связи с микроконтроллером платы управления.

Алгоритм выравнивания полуфабрикатов. На рисунке 4 САУ представлена схематично. Введена система координат, центр которой находится в точке M_0 , ось Z направлена вниз, ось X составляет угол α с центральной линией конвейера. Ось Y дополняет правую тройку, точка M_0 – геометрический центр робота.

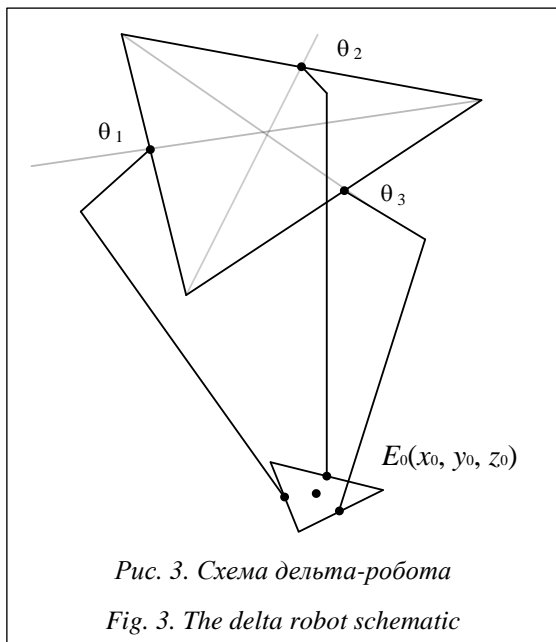
Угол α появляется из-за того, что робот-манипулятор закреплен на раме таким образом, чтобы при движении не мог ее коснуться. Поэтому центральная ось конвейера не сонаправлена ни с одной из осей системы координат робота. Угол α постоянный.

В ходе экспериментов по отладке системы планирования движений робота-манипулятора ДР-1 выяснено, что наибольшую скорость работы можно получить, если задавать траекторию движения минимальным количеством точек. Это связано с тем, что робот-манипулятор

в заданной точке должен остановиться, поэтому при выполнении операции выравнивания полуфабрикатов использована траектория в форме треугольника.

На рисунке 4 синий прямоугольник – коробка с полуфабрикатами, точка M_k – центр ее верхней грани. Коробка появляется в зоне видимости робота-манипулятора в момент прохождения через детектор (точка M_d). При этом M_0 – начало системы координат робота-манипулятора. Точка M_1 – координата центра нижней пластины рабочего органа в положении ожидания. Точка M_2 – координата первого соприкосновения нижней пластины робота-манипулятора с коробкой. В этот момент осуществляется воздействие на полуфабрикаты в коробке самим роботом-манипулятором и пневматическим вибратором, размещенным на рабочем органе. Такое воздействие продолжается до точки M_3 ; рабочий орган сопровождает коробку. В точке M_3 рабочий орган прекращает воздействие на коробку и начинает движение в положение ожидания – точку M_1 .

Высоты точек M_1, M_2, M_3 подобраны опытным путем. От них зависят качество выравнивания полуфабрикатов, сохранность упаковки и самих полуфабрикатов. В ходе испытаний



было замечено, что наиболее эффективно выравнивание полуфабрикатов при установке точек M_2 и M_3 на высоте верхней грани коробки. При этом такая высота минимизирует воздействие рабочего органа на коробку. Высота точки M_1 выбирается на несколько сантиметров меньше (ось Z направлена вниз), чем M_2 и M_3 . Этого достаточно, чтобы при перемещении рабочего органа не зацепить коробку и невыровненные полуфабрикаты в ней.

Алгоритм разделен на два блока, работающих параллельно.

Первый блок отвечает за обработку сигналов от детектора коробок и записи времени обнаружения коробки на границе рабочей области t_{0i} в кольцевую очередь.

В случае прихода сигнала от детектора коробок нужно сделать следующее. Если очередь пуста или разница между текущим временем t и последним элементом очереди больше либо

равна Δt , то добавить текущее время t в конец очереди.

Пустая очередь означает, что еще ни одна коробка не прошла через детектор. Разницу времен нужно определить, чтобы разделить две коробки, которые идут друг за другом.

Второй блок отвечает за расчет положения коробки, время t_0 которой записано в начало очереди, и отправку управляющих команд.

1. Установить сигнал «Работа разрешена» в ЛОЖЬ.

2. Если сигнал «Работа разрешена» в значении ЛОЖЬ, перейти к п. 3, иначе – к п. 5.

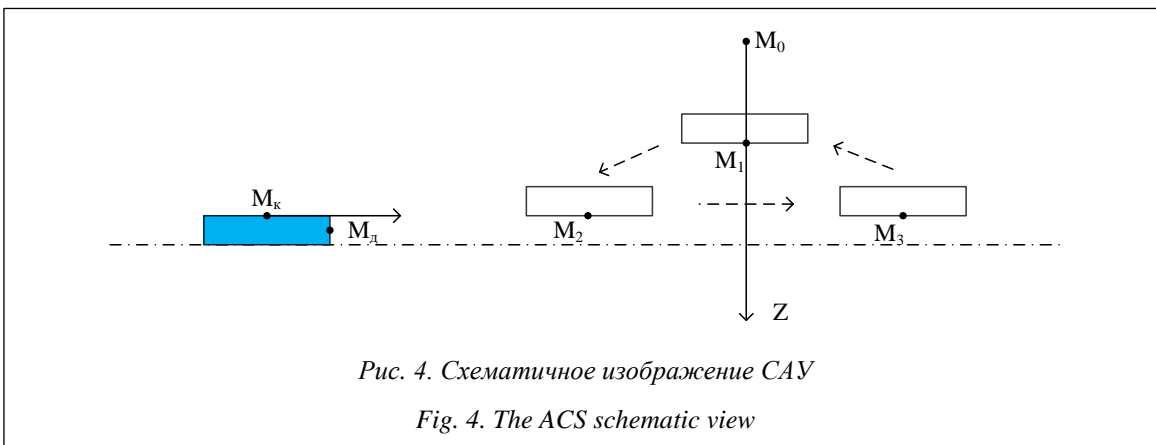
3. Если размер очереди больше 0, то рассчитать текущее положение коробки, перейти к п. 4, иначе – к п. 2.

4. Если текущее положение коробки находится в пределах рабочей области робота, то установить сигнал «Работа разрешена» в ИСТИНА и удалить первый элемент из очереди, перейти к п. 2.

5. Выполнить последовательное перемещение из точки M_1 в M_2 , затем из M_2 в M_3 и из M_3 в M_1 . Установить сигнал «Работа разрешена» в ЛОЖЬ, перейти к п. 2.

Реализация ПО. В состав САУ входит бортовой микроконтроллер STM32F407VG, который размещен на плате управления. Его характеристики: процессор ARM Cortex M4, 168 МГц, 192 Кб ОЗУ, 1 Мб ПЗУ. Программирование осуществлялось на языке C++14 в CubeIDE. Использовалась операционная система реального времени FreeRTOS для того, чтобы несколько задач работали в псевдопараллельном режиме. Задачи, решаемые ПО микроконтроллера:

- управление электродвигателями через интерфейс STEP/DIR/ENABLE;
- опрос концевых датчиков для калибровки робота;



- опрос датчика скорости конвейерной ленты;
- опрос детектора коробок;
- взаимодействие с существующей системой предприятия через дискретные входы-выходы;
- индикация состояния САУ;
- обмен данными с управляющим компьютером через TCP.

Разработанное ПО управляющего компьютера реализовано на языке программирования C++11. Используется операционная система Linux Ubuntu 18.04 LTS, программная платформа ROS. Характеристики управляющего компьютера: процессор Intel Atom, 4 Гб ОЗУ, 32 Гб ПЗУ. Управляющий компьютер решает следующие задачи:

- расчет положения коробки и траектории робота-манипулятора для выравнивания полуфабрикатов;
- обмен данными с бортовым микроконтроллером платы управления;
- обеспечение интерфейса оператора для отображения текущего состояния САУ и задания параметров ее работы;
- логирование действия САУ.

Интерфейс реализован на языке программирования Python 3 с использованием библиотеки KivyMD и представлен в виде узла в концепции ROS, который обменивается сообщениями с другими частями программы.

Особенностью реализации программного комплекса является его быстродействие. Именно поэтому используется язык программирования C++. Важно было минимизировать количество данных при обмене между бортовым микроконтроллером и управляющим компьютером, чтобы не переполнялся буфер обмена. В результате удалось добиться минимального времени обработки сигнала от детектора коробок, составляющего 10 мс. Этого достаточно, чтобы корректно производить выравнивание полуфабрикатов около 110 раз в минуту.

На рисунке 5 представлена структурная схема ПО. Она разделена на две части, которые связаны при помощи TCP-сервера и TCP-клиента. Очередь необходима для обмена данными между блоками управляющего компьютера, так как они реализованы как отдельные программы.

Испытания ПО. В первую очередь измерялась скорость прохождения сигнала от момента детектирования коробки до оказания

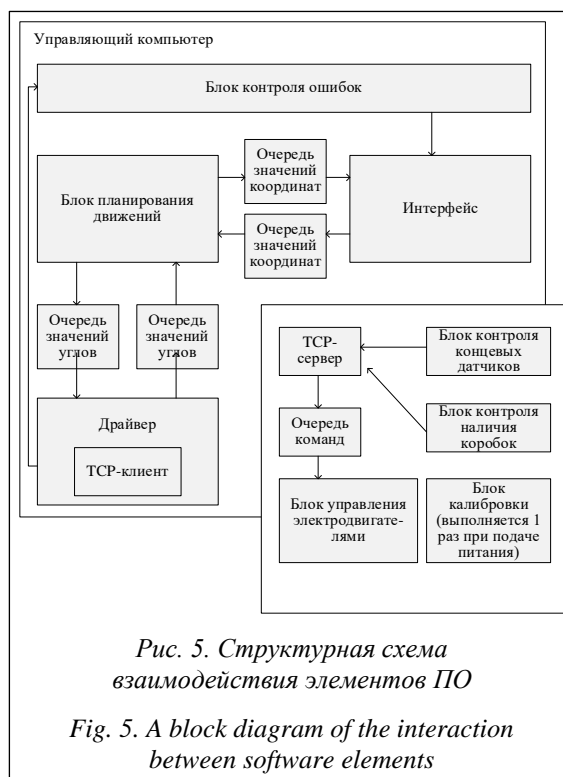


Рис. 5. Структурная схема взаимодействия элементов ПО
 Fig. 5. A block diagram of the interaction between software elements

управляющего воздействия на него. Время составило не более 10 мс.

Была проверена корректность обработки ПО различных ситуаций, которые могут произойти с САУ: нестандартные положения робота-манипулятора, застревание коробки с полуфабрикатами около детектора коробок и блокирование его работы. Особое внимание уделялось интерфейсу.

В ходе испытаний были использованы три коробки с выступающими за их верхнюю границу наггетсами. Цель испытания заключалась в том, чтобы проверить, в скольких случаях после выравнивания автоматизированной системой упаковка с наггетсами будет выступать за пределы коробки. Было важно проверить целостность упаковки полуфабрикатов после выравнивания. Коробки запускались на конвейерную ленту в нескольких последовательностях: одна коробка, две коробки, между которыми было расстояние, две рядом стоящие коробки, три рядом стоящие коробки. Наиболее сложный случай, которому уделялось особое внимание, – это три подряд идущие коробки.

Было выполнено 20 экспериментов. Коробки стояли друг за другом без промежутков. Автоматизированная система выполняла выравнивание, после чего оценивался результат (<http://www.swsys.ru/uploaded/image/2023-2/2023-2-dop/20.jpg>). Во всех случаях пакеты с

нагдетсами в коробках были выровнены. В некоторых коробках были обнаружены повреждения. Следует отметить, что это результат нескольких экспериментов. В реальности выравнивание будет осуществляться только один раз.

Результаты. В ходе разработки ПО САУ для выравнивания полуфабрикатов удалось достичь требуемых показателей скорости работы и доли брака. Особое внимание уделено со-

хранности упаковки и самих полуфабрикатов. За счет подбора параметров для алгоритма выравнивания не повреждаются ни коробка, ни полуфабрикаты.

Использование современных языков программирования и программных продуктов для реализации ПО САУ дает большой потенциал для расширения возможностей САУ. Таким образом, цель разработки можно считать достигнутой.

Список литературы

1. Pyo Y., Cho H., Jung R., Lim T. ROS Robot Programming. ROBOTIS Publ., 2017, 487 p.
2. Тывес Л.И. Механизмы робототехники. Концепция развязок в кинематике, динамике и планировании движений. М.: Ленанд, 2018. С. 204.
3. Zsombor-Murray P. Descriptive Geometric Kinematic Analysis of Clavel's "Delta" Robot. 2004, pp. 1–5. URL: <https://www.cim.mcgill.ca/~paul/clavdelt.pdf> (дата обращения: 12.02.2023).
4. Гемуев Ш.Ш., Воротников С.А. Роботизированный комплекс сортировки твердых бытовых отходов // Прикаспийский журнал: управление и высокие технологии. 2019. № 2. С. 194–207. doi: 10.21672/2074-1707.2019.46.2.194-207.
5. Колтыгин Д.С., Седельников И.А. Методика разработки программы управления роботом для робота-манипулятора Delta* // Науч. вестн. НГТУ. 2018. Т. 70. № 1. С. 103–116. doi: 10.17212/1814-1196-2018-1-103-116.
6. Zhang G., Li Z., Ni F., Liu H. A real-time robot control framework using ROS Control for 7-DoF Light-weight robot. Proc. IEEE/ASME Int. Conf. AIM, 2019, pp. 1754–1579. doi: 10.1109/AIM.2019.8868488.
7. Joseph L. Mastering ROS for Robotics Programming. Packt Publ., 2018, 689 p.

Software & Systems

doi: 10.15827/0236-235X.142.327-333

2023, vol. 36, no. 2, pp. 327–333

Developing software for automatic control system for aligning semi-finished products

Mikhail V. Tarachkov
Oleg V. Tolstel
Alexandr L. Kalabin

For citation

Tarachkov, M.V., Tolstel, O.V., Kalabin, A.L. (2023) 'Developing software for automatic control system for aligning semi-finished products', *Software & Systems*, 36(2), pp. 327–333 (in Russ.). doi: 10.15827/0236-235X.142.327-333

Article info

Received: 13.02.2023

After revision: 20.03.2023

Accepted: 22.03.2023

Abstract. There is a developed automatic control system (ACS) to align semi-finished products on a high-speed conveyor line. It has a DR-1 robot manipulator and a working body that pressures and exerts a vibration impact. The system was created to increase the line efficiency and reduce the reject rate. The ACS includes an on-board microcontroller STM32F407VG (clock frequency 168 MHz, 192 KB RAM, 1 MB ROM) and a control computer based on an Intel Atom processor, 4 GB RAM, 32 GB ROM. The microcontroller software was developed in the C++14 programming language in the CubeIDE development platform. The FreeRTOS real-time operating system is used to process several tasks in a pseudo-parallel mode by a microcontroller: electric motor control, processing data from sensors, interaction with the control computer. The control computer has an implemented software in the C++11 programming language, the Linux Ubuntu 18.04 LTS operating system, the Robot Operating System (ROS) software platform. The control computer provides the ACS operator with an interface written in the Python programming language using the KivyMD library. However, the most interesting part of the software is solving the problems of direct and inverse kinematics for the delta manipulator and the algorithm for aligning semi-finished products. The problems of direct and inverse kinematics are solved in general terms. The paper gives the features of their implementation for the robotic manipulator DR-1. The authors carried out experiments

to formalize the alignment algorithm for semi-finished products. It was necessary to perform about 110 alignments per minute, and this required finding the most efficient trajectory for moving a working body. The paper gives the results of testing the ACS software; considers various cases of arrangement of boxes with semi-finished products; assesses the safety of semi-finished products and packaging. The efficiency of the considered algorithm is also estimated.

The use of the proposed ACS software made it possible to increase the line productivity and reduce the reject rate.

Keywords: automatic control system, robotic arm, robot operating system, industrial robot, software, motion planning

Reference List

1. Pyo, Y., Cho, H., Jung, R., Lim, T. (2017) *ROS Robot Programming*, ROBOTIS Publ., 487 p.
2. Tyves, L.I. (2018) *Mechanisms of Robotics. The Concept of Interchanges in Kinematics, Dynamics and Motion Planning*, Moscow, 204 p. (in Russ.).
3. Zsombor-Murray, P. (2004) *Descriptive Geometric Kinematic Analysis of Clavel's "Delta" Robot*, pp. 1–5, available at: <https://www.cim.mcgill.ca/~paul/clavdelt.pdf> (accessed February 12, 2023).
4. Gemuev, Sh.Sh., Vorotnikov, S.A. (2019) 'Robotic complex for sorting of municipal solid waste', *Caspian J.: Management and High Tech.*, (2), pp. 194–207. doi: 10.21672/2074-1707.2019.46.2.194-207 (in Russ.).
5. Koltygin, D.S., Sedelnikov, I.A. (2018) 'The methodology for developing a robot control program for the Delta robot manipulator', *Sci. Bull. of NSTU*, 70(1), pp. 103–116. doi: 10.17212/1814-1196-2018-1-103-116 (in Russ.).
6. Zhang, G., Li, Z., Ni, F., Liu, H. (2019) 'A real-time robot control framework using ROS Control for 7-DoF Lightweight robot', *Proc. IEEE/ASME Int. Conf. AIM*, pp. 1754–1579. doi: 10.1109/AIM.2019.8868488.
7. Joseph, L. (2018) *Mastering ROS for Robotics Programming*, Packt Publ., 689 p.

Авторы

Тарачков Михаил Владимирович¹,

ассистент, mishklgpmi@mail.ru

Толстель Олег Владимирович¹,

к.т.н., доцент, tolstel.oleg@mail.ru

Калабин Александр Леонидович²,

д.ф.-м.н., профессор,

akalabin@yandex.ru

Authors

Mikhail V. Tarachkov¹,

Assistant, mishklgpmi@mail.ru

Oleg V. Tolstel¹, Ph.D. (Engineering),

Associate Professor, tolstel.oleg@mail.ru

Alexandr L. Kalabin²,

Dr.Sc. (Physics and Mathematics),

Professor, akalabin@yandex.ru

¹ Балтийский федеральный университет им. Иммануила Канта, образовательно-научный кластер «Институт высоких технологий», г. Калининград, 236041, Россия

² Тверской государственный технический университет, г. Тверь, 170026, Россия

¹ Immanuel Kant Baltic Federal University, Education and Science Cluster "Institute of High Technology", Kaliningrad, 236041, Russian Federation

² Tver State Technical University, Tver, 170026, Russian Federation